STATISTICS: textbooks and monographs volume 172

Computational Methods in Statistics and Econometrics

Hisashi Tanizaki
Graduate School of Economics
Kobe University, Kobe 657-8501, Japan
(tanizaki@kobe-u.ac.jp)

To My Family,
Miyuki, Toshifumi and Yasuyuki

Preface

As the personal computer progresses day by day, computer-intensive procedures have been developed in the field of statistics and econometrics. The computational procedures in statistics and econometrics include both Monte Carlo methods and nonparametric methods (or distribution-free methods). In the first half of this book, the Monte Carlo methods are discussed. That is, some representative random number generation methods and their applications are shown. The second half of this book is related to computer-intensive statistical techniques other than Monte Carlo methods and simulations, where the nonparametric methods are introduced.

Chapter 1 is an introduction to statistics and econometrics, which corresponds to my lecture notes in mathematical statistics course (about 15 lectures, each 90 minutes long) for first-year graduate students. Based on Chapter 1, the Monte Carlo and nonparametric methods are discussed in Chapters 2 – 8.

In the Monte Carlo methods, we discuss how to generate various random draws. Almost all the random draws are based on the uniform random draws. Therefore, it is one of the most important tasks to investigate uniform random number generation. Transforming the variable from the uniform random variable, various random draws are generated in Chapter 2, e.g., Bernoulli random draws, binomial random draws, normal random draws, χ^2 random draws, t random draws, F random draws, exponential random draws, gamma random draws, beta random draws, Cauchy random draws, logistic random draws and others.

Importance resampling, rejection sampling and the Metropolis-Hastings algorithm are the methods to generate random draws from any distribution, which are useful tools for random number generation even when it is not easy to generate the random draws. Three sampling methods are discussed in Chapter 3. Thus, in the Monte Carlo methods, random number generation is very important. Once we have the random draws, simply the arithmetic average of the random draws indicates the estimate of mean. The arithmetic average is approximately distributed with a normal random variable by the central limit theorem. Therefore, the statistical inference also becomes quite easy, using the random draws.

As some applications in the Monte Carlo methods, Baysian inference (Chapter 4), bias correction of the ordinary least squares (OLS) estimator in the autoregressive models (Chapter 5) and nonlinear non-Gaussian state space modeling (Chapter 6) are shown.

In the nonparametric methods, nonparametric statistical tests are discussed in Chapters 7 and 8. Nonparametric tests of difference between two sample means include score tests (e.g., Wilcoxon rank sum test, normal score test, logistic score test, Cauchy score test and so on) and Fisher's randomization test (or Fisher's permutation test). The nonparametric tests of difference between two sample means are discussed in Chapter 7. One of the features of nonparametric tests is that we do not have to impose any assumption on the underlying distribution. From no restriction on the distribution, it could be expected that nonparametric tests would be less powerful than the conventional parametric tests such as the *t* test. However, it is shown that the Wilcoxon rank sum test is as powerful as the *t* test under the location-shift alternatives and moreover that the Wilcoxon test is sometimes much more powerful than the *t* test. Especially, the remarkable fact about the Wilcoxon test is that it is about 95 per cent as powerful as the *t* test for normal data. It is known that Pitman's asymptotic relative efficiency of the normal score test relative to the *t* test is greater than one under the location-shift alternatives. This implies that the power of the normal score test is always larger than that of the *t* test. It is known that the normal score test is less powerful than the Wilcoxon test if the tails of the underlying distributions are diffuse. Since in general the nonparametric tests require a large computational burden, however, there are few studies on small sample properties although asymptotic properties from various aspects were studied in the past.

In addition to testing difference between two sample means, in Chapter 8 we also consider testing independence between two samples, which corresponds to testing correlation coefficient and regression coefficient. Small sample properties are discussed in the nonparametric statistical tests part of the book.

Thus, some selected representative computer-intensive methods are treated in this book, where the source codes are shown by Fortran 77 and sometimes C languages for the purpose of practical understanding. For this book, I used seven personal computers, i.e.,

- Athlon 1.4 GHz CPU, Windows 2000 Operating System
- Pentium III 1.4 GHz Dual CPU, and Windows 2000 Operating System
- Pentium III 1.0 GHz Dual CPU, and Windows 2000 Operating System
- Athlon 2000+ Dual CPU, and Windows 2000 Operating System
- Athlon 2000+ Dual CPU, and Linux (Slackware 8.0) Operating System
  (see `http://www.slackware.com` for Slackware)
- Pentium III 1.4 GHz Dual CPU, and Linux (Plamo Linux 2.2.5, which is equivalent to Slackware+Japanese) Operating System
  (see `http://www.linet.jp/Plamo` or `http://plamo-linux.jp` for Plamo Linux, which is a Japanese site)
- Pentium III 1.0 GHz Dual CPU, and Linux (Plamo Linux 2.2.5) Operating System

For almost two years, my personal computers have been running all the time to prepare this book. Under the Windows 2000 Operating System, the following Fortran and C compilers are used for computation.

- Open WATCOM C/C++ and Fortran 1.0 (`http://www.openwatcom.org`)
- Cygwin (`http://www.cygwin.com`)
- DJGPP (`http://www.delorie.com/djgpp`)

In addition to the free compilers shown above, in Section 7.5, Tables 7.4 and 7.5, the IMSL library (`http://www.vni.com/products/imsl`) with Microsoft Fortran PowerStation Version 4.00 is used to obtain the percent points of the normal, t and F distributions.

I am indebted to many people for assistance with this book. All cannot be mentioned in this short space, but I would like to acknowledge the Acquisitions Editor Taisuke Soda (Marcel Dekker, Inc.), who suggested that I write a book on statistical computing. I presented some chapters at the Econometrics workshop, Graduate School of Economics, Kobe University. I would like to thank the participants at the workshop for valuable comments. Furthermore, as mentioned above, Chapter 1 is based on my lecture notes in mathematical statistics course. Graduate students found some errors in the lecture notes. I am grateful to them for that. This research was partially supported by Japan Society for the Promotion of Science, Grants-in-Aid for Scientific Research (C)(2) #14530033 and Grants-in-Aid for the 21st Century COE Program #15COE500.

I used LaTeX to write this book. I did all of the typing, programming and so on. Therefore, all mistakes are mine. In programming, I use Fortran 77 in the first half of the book (Chapters 2 – 6) and C in the last half (Chapters 7 and 8). The *pointer*, which is often used in C language, is not familiar to Fortran users. Therefore, in Chapters 7 and 8, I did not use the pointer. Instead, I declared the external variables in the source codes, which are similar to the `common` sentence in Fortran 77. Finally, all the source codes used in this book are available in the CD-ROM attached with this book (therefore, readers do not have to type the source codes from the beginning).

Hisashi Tanizaki
Kobe, Japan

Contents

List of Tables

List of Figures

Chapter 1

Elements of Statistics

In this chapter, the statistical methods used in the proceeding chapters are summarized. Mood, Graybill and Bose (1974), Hogg and Craig (1995) and Stuart and Ord (1991, 1994) are good references in Sections 1.1 – 1.8, while Judge, Hill, Griffiths and Lee (1980) and Greene (1993, 1997, 2000) are representative textbooks in Section 1.9.

1.1 Event and Probability

1.1.1 Event

We consider an **experiment** whose outcome is not known in advance but an event occurs with probability, which is sometimes called a **random experiment**. The **sample space** of an experiment is the set of all possible outcomes. Each element of a sample space is called an **element** of the sample space or a **sample point**, which represents each outcome obtained by the experiment. An **event** is any collection of outcomes contained in the sample space, or equivalently a subset of the sample space. A **simple event** consists of exactly one element and a **compound event** consists of more than one element. Sample space is denoted by Ω and sample point is given by ω.

Suppose that event A is a subset of sample space Ω. Let ω be a sample point in event A. Then, we say that a sample point ω is contained in a sample space A, which is denoted by $\omega \in A$.

A set of the sample points which does not belong to event A is called the **complementary event** of A, which is denoted by A^c. An event which do not have any sample point is called the **empty event**, denoted by $\emptyset$. Conversely, an event which includes all possible sample points is called the **whole event**, represented by Ω.

Next, consider two events A and B. A set consisting of the whole sample points which belong to either event A or event B is called the **sum event**, which is denoted by $A \cap B$. A set consisting of the whole sample points which belong to both event A and event B is called the **product event**, denoted by $A \cap B$. When $A \cap B = \emptyset$, we say that events A and B are **mutually exclusive**.

Example 1.1: Consider an experiment of casting a die. We have six sample points, which are denoted by $\omega_1 = \{1\}$, $\omega_2 = \{2\}$, $\omega_3 = \{3\}$, $\omega_4 = \{4\}$, $\omega_5 = \{5\}$ and $\omega_6 = \{6\}$, where ω_i represents the sample point that we have i. In this experiment, the sample space is given by $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4, \omega_5, \omega_6\}$. Let A be the event that we have even numbers and B be the event that we have multiples of three. Then, we can write as $A = \{\omega_2, \omega_4, \omega_6\}$ and $B = \{\omega_3, \omega_6\}$. The complementary event of A is given by $A^c = \{\omega_1, \omega_3, \omega_5\}$, which is the event that we have odd numbers. The sum event of A and B is written as $A \cup B = \{\omega_2, \omega_3, \omega_4, \omega_6\}$, while the product event is $A \cap B = \{\omega_6\}$. Since $A \cap A^c = \emptyset$, we have the fact that A and A^c are mutually exclusive.

Example 1.2: Cast a coin three times. In this case, we have the following eight sample points:

$$\begin{array}{llll} \omega_1 = \text{(H,H,H)}, & \omega_2 = \text{(H,H,T)}, & \omega_3 = \text{(H,T,H)}, & \omega_4 = \text{(H,T,T)}, \\ \omega_5 = \text{(T,H,H)}, & \omega_6 = \text{(T,H,T)}, & \omega_7 = \text{(T,T,H)}, & \omega_8 = \text{(T,T,T)}, \end{array}$$

where H represents head while T indicates tail. For example, (H,T,H) means that the first flip lands head, the second flip is tail and the third one is head. Therefore, the sample space of this experiment can be written as:

$$\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4, \omega_5, \omega_6, \omega_7, \omega_8\}.$$

Let A be an event that we have two heads, B be an event that we obtain at least one tail, C be an event that we have head in the second flip, and D be an event that we obtain tail in the third flip. Then, the events A, B, C and D are give by:

$$\begin{array}{ll} A = \{\omega_2, \omega_3, \omega_5\}, & B = \{\omega_2, \omega_3, \omega_4, \omega_5, \omega_6, \omega_7, \omega_8\}, \\ C = \{\omega_1, \omega_2, \omega_5, \omega_6\}, & D = \{\omega_2, \omega_4, \omega_6, \omega_8\}. \end{array}$$

Since A is a subset of B, denoted by $A \subset B$, a sum event is $A \cup B = B$, while a product event is $A \cap B = A$. Moreover, we obtain $C \cap D = \{\omega_2, \omega_6\}$ and $C \cup D = \{\omega_1, \omega_2, \omega_4, \omega_5, \omega_6, \omega_8\}$.

1.1.2 Probability

Let $n(A)$ be the number of sample points in A. We have $n(A) \leq n(B)$ when $A \subseteq B$. Each sample point is equally likely to occur. In the case of Example 1.1 (Section 1.1.1), each of the six possible outcomes has probability 1/6 and in Example 1.2 (Section 1.1.1), each of the eight possible outcomes has probability 1/8. Thus, the probability which the event A occurs is defined as:

$$P(A) = \frac{n(A)}{n(\Omega)}.$$

In Example 1.1, $P(A) = 3/6$ and $P(A \cap B) = 1/6$ are obtained, because $n(\Omega) = 6$, $n(A) = 3$ and $n(A \cap B) = 1$. Similarly, in Example 1.2, we have $P(C) = 4/8$,

$P(A \cap B) = P(A) = 3/8$ and so on. Note that we obtain $P(A) \leq P(B)$ because of $A \subseteq B$.

It is known that we have the following three properties on probability:

$$0 \leq P(A) \leq 1, \tag{1.1}$$

$$P(\Omega) = 1, \tag{1.2}$$

$$P(\emptyset) = 0. \tag{1.3}$$

$\emptyset \subseteq A \subseteq \Omega$ implies $n(\emptyset) \leq n(A) \leq n(\Omega)$. Therefore, we have:

$$\frac{n(\emptyset)}{n(\Omega)} \leq \frac{n(A)}{n(\Omega)} \leq \frac{n(\Omega)}{n(\Omega)} = 1.$$

Dividing by $n(\Omega)$, we obtain:

$$P(\emptyset) \leq P(A) \leq P(\Omega) = 1.$$

Because $\emptyset$ has no sample point, the number of the sample point is given by $n(\emptyset) = 0$ and accordingly we have $P(\emptyset) = 0$. Therefore, $0 \leq P(A) \leq 1$ is obtained as in (1.1). Thus, (1.1) – (1.3) are obtained.

When events A and B are mutually exclusive, i.e., when $A \cap B = \emptyset$, then $P(A \cup B) = P(A) + P(B)$ holds. Moreover, since A and A^c are mutually exclusive, $P(A^c) = 1 - P(A)$ is obtained. Note that $P(A \cup A^c) = P(\Omega) = 1$ holds. Generally, unless A and B are not exclusive, we have the following formula:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B),$$

which is known as the **addition rule**. In Example 1.1, each probability is given by $P(A \cup B) = 2/3$, $P(A) = 1/2$, $P(B) = 1/3$ and $P(A \cap B) = 1/6$. Thus, in the example we can verify that the above addition rule holds.

The probability which event A occurs, given that event B has occurred, is called the **conditional probability**, i.e.,

$$P(A|B) = \frac{n(A \cap B)}{n(B)} = \frac{P(A \cap B)}{P(B)},$$

or equivalently,

$$P(A \cap B) = P(A|B)P(B),$$

which is called the **multiplication rule**. When event A is **independent** of event B, we have $P(A \cap B) = P(A)P(B)$, which implies that $P(A|B) = P(A)$. Conversely, $P(A \cap B) = P(A)P(B)$ implies that A is independent of B. In Example 1.2, because of $P(A \cap C) = 1/4$ and $P(C) = 1/2$, the conditional probability $P(A|C) = 1/2$ is obtained. From $P(A) = 3/8$, we have $P(A \cap C) \neq P(A)P(C)$. Therefore, A is not independent of C. As for C and D, since we have $P(C) = 1/2$, $P(D) = 1/2$ and $P(C \cap D) = 1/4$, we can show that C is independent of D.

1.2 Random Variable and Distribution

1.2.1 Univariate Random Variable and Distribution

The **random variable** X is defined as the real value function on sample space Ω. Since X is a function of a sample point ω, it is written as $X = X(\omega)$. Suppose that $X(\omega)$ takes a real value on the interval I. That is, X depends on a set of the sample point ω, i.e., $\{\omega; X(\omega) \in I\}$, which is simply written as $\{X \in I\}$.

In Example 1.1 (Section 1.1.1), suppose that X is a random variable which takes the number of spots up on the die. Then, X is a function of ω and takes the following values:

$$X(\omega_1) = 1, \quad X(\omega_2) = 2, \quad X(\omega_3) = 3, \quad X(\omega_4) = 4,$$
$$X(\omega_5) = 5, \quad X(\omega_6) = 6.$$

In Example 1.2 (Section 1.1.1), suppose that X is a random variable which takes the number of heads. Depending on the sample point ω_i, X takes the following values:

$$X(\omega_1) = 3, \quad X(\omega_2) = 2, \quad X(\omega_3) = 2, \quad X(\omega_4) = 1,$$
$$X(\omega_5) = 2, \quad X(\omega_6) = 1, \quad X(\omega_7) = 1, \quad X(\omega_8) = 0.$$

Thus, the random variable depends on a sample point.

There are two kinds of random variables. One is a **discrete random variable**, while another is a **continuous random variable**.

Discrete Random Variable and Probability Function: Suppose that the discrete random variable X takes $x_1, x_2, \cdots$, where $x_1 < x_2 < \cdots$ is assumed. Consider the probability that X takes x_i, i.e., $P(X = x_i) = p_i$, which is a function of x_i. That is, a function of x_i, say $f(x_i)$, is associated with $P(X = x_i) = p_i$. The function $f(x_i)$ represents the probability in the case where X takes x_i. Therefore, we have the following relation:

$$P(X = x_i) = p_i = f(x_i), \quad i = 1, 2, \cdots,$$

where $f(x_i)$ is called the **probability function** of X.

More formally, the function $f(x_i)$ which has the following properties is defined as the probability function.

$$f(x_i) \geq 0, \quad i = 1, 2, \cdots,$$
$$\sum_i f(x_i) = 1.$$

Furthermore, for an event A, we can write a probability as the following equation:

$$P(X \in A) = \sum_{x_i \in A} f(x_i).$$

Several functional forms of $f(x_i)$ are shown in Section 2.4.

In Example 1.2 (Section 1.1.1), all the possible values of X are 0, 1, 2 and 3. (note that X denotes the number of heads when a die is cast three times). That is, $x_1 = 0$, $x_2 = 1$, $x_3 = 2$ and $x_4 = 3$ are assigned in this case. The probability that X takes x_1, x_2, x_3 or x_4 is given by:

$$\begin{aligned}
P(X = 0) &= f(0) = P(\{\omega_8\}) = \frac{1}{8}, \\
P(X = 1) &= f(1) = P(\{\omega_4, \omega_6, \omega_7\}) = P(\{\omega_4\}) + P(\{\omega_6\}) + P(\{\omega_7\}) = \frac{3}{8}, \\
P(X = 2) &= f(2) = P(\{\omega_2, \omega_3, \omega_5\}) = P(\{\omega_2\}) + P(\{\omega_3\}) + P(\{\omega_5\}) = \frac{3}{8}, \\
P(X = 3) &= f(3) = P(\{\omega_1\}) = \frac{1}{8},
\end{aligned}$$

which can be written as:

$$P(X = x) = f(x) = \frac{3!}{x!(3-x)!}\left(\frac{1}{2}\right)^3, \quad x = 0, 1, 2, 3.$$

For $P(X = 1)$ and $P(X = 2)$, note that each sample point is mutually exclusive. The above probability function is called the **binomial distribution** discussed in Section 2.4.5. Thus, it is easy to check $f(x) \geq 0$ and $\sum_x f(x) = 1$ in Example 1.2.

Continuous Random Variable and Probability Density Function: Whereas a discrete random variable assumes at most a countable set of possible values, a continuous random variable X takes any real number within an interval I. For the interval I, the probability which X is contained in A is defined as:

$$P(X \in I) = \int_I f(x)\, \mathrm{d}x.$$

For example, let I be the interval between a and b for $a < b$. Then, we can rewrite $P(X \in I)$ as follows:

$$P(a < X < b) = \int_a^b f(x)\, \mathrm{d}x,$$

where $f(x)$ is called the **probability density function** of X, or simply the **density function** of X.

In order for $f(x)$ to be a probability density function, $f(x)$ has to satisfy the following properties:

$$\begin{aligned}
&f(x) \geq 0, \\
&\int_{-\infty}^{\infty} f(x)\, \mathrm{d}x = 1.
\end{aligned}$$

Some functional forms of $f(x)$ are discussed in Sections 2.1 – 2.3.

For a continuous random variable, note as follows:

$$P(X = x) = \int_x^x f(t)\,\mathrm{d}t = 0.$$

In the case of discrete random variables, $P(X = x_i)$ represents the probability which X takes x_i, i.e., $p_i = f(x_i)$. Thus, the probability function $f(x_i)$ itself implies probability. However, in the case of continuous random variables, $P(a < X < b)$ indicates the probability which X lies on the interval (a, b).

Example 1.3: As an example, consider the following function:

$$f(x) = \begin{cases} 1, & \text{for } 0 < x < 1, \\ 0, & \text{otherwise.} \end{cases}$$

Clearly, since $f(x) \geq 0$ for $-\infty < x < \infty$ and $\int_{-\infty}^{\infty} f(x)\,\mathrm{d}x = \int_0^1 f(x)\,\mathrm{d}x = [x]_0^1 = 1$, the above function can be a probability density function. In fact, it is called a **uniform distribution**. See Section 2.1 for the uniform distribution.

Example 1.4: As another example, consider the following function:

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2},$$

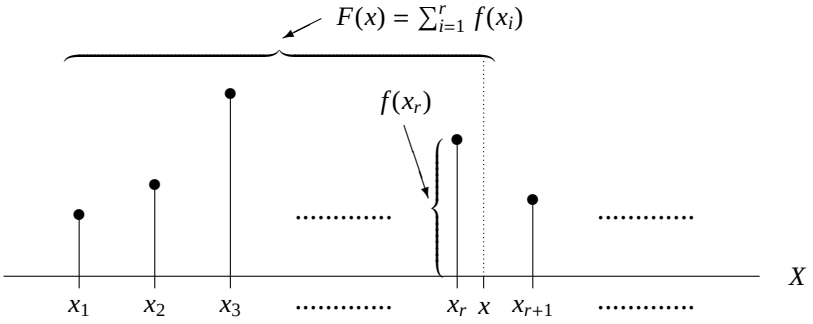
for $-\infty < x < \infty$. Clearly, we have $f(x) \geq 0$ for all x. We check whether $\int_{-\infty}^{\infty} f(x)\,\mathrm{d}x = 1$. First of all, we define I as $I = \int_{-\infty}^{\infty} f(x)\,\mathrm{d}x$.

To show $I = 1$, we may prove $I^2 = 1$ because of $f(x) > 0$ for all x, which is shown as follows:

$$\begin{aligned} I^2 &= \Big(\int_{-\infty}^{\infty} f(x)\,\mathrm{d}x\Big)^2 = \Big(\int_{-\infty}^{\infty} f(x)\,\mathrm{d}x\Big)\Big(\int_{-\infty}^{\infty} f(y)\,\mathrm{d}y\Big) \\ &= \Big(\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}x^2)\,\mathrm{d}x\Big)\Big(\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}y^2)\,\mathrm{d}y\Big) \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \exp\Big(-\frac{1}{2}(x^2 + y^2)\Big)\,\mathrm{d}x\,\mathrm{d}y \\ &= \frac{1}{2\pi} \int_0^{2\pi}\int_0^{\infty} \exp(-\frac{1}{2}r^2) r\,\mathrm{d}r\,\mathrm{d}\theta \\ &= \frac{1}{2\pi} \int_0^{2\pi}\int_0^{\infty} \exp(-s)\,\mathrm{d}s\,\mathrm{d}\theta = \frac{1}{2\pi} 2\pi[-\exp(-s)]_0^{\infty} = 1. \end{aligned}$$

In the fifth equality, integration by substitution is used. See Appendix 1.1 for the integration by substitution. $x = r\cos\theta$ and $y = r\sin\theta$ are taken for transformation,

Figure 1.1: Probability Function $f(x)$ and Distribution Function $F(x)$
— Discrete Random Variable —



Note that r is the integer which satisfies $x_r \leq x < x_{r+1}$.

which is a one-to-one transformation from (x, y) to (r, θ). Note that $0 < r < +\infty$ and $0 < \theta < 2\pi$. The Jacobian is given by:

$$J = \begin{vmatrix} \dfrac{\partial x}{\partial r} & \dfrac{\partial x}{\partial \theta} \\ \dfrac{\partial y}{\partial r} & \dfrac{\partial y}{\partial \theta} \end{vmatrix} = \begin{vmatrix} \cos\theta & -r\sin\theta \\ \sin\theta & r\cos\theta \end{vmatrix} = r.$$

In the inner integration of the sixth equality, again, integration by substitution is utilized, where transformation is $s = \dfrac{1}{2}r^2$.

Thus, we obtain the result $I^2 = 1$ and accordingly we have $I = 1$ because of $f(x) \geq 0$. Therefore, $f(x) = e^{-\frac{1}{2}x^2}/\sqrt{2\pi}$ is also taken as a probability density function. Actually, this density function is called the **standard normal probability density function**, discussed in Section 2.2.1.

Distribution Function: The **distribution function** (or the **cumulative distribution function**), denoted by $F(x)$, is defined as:

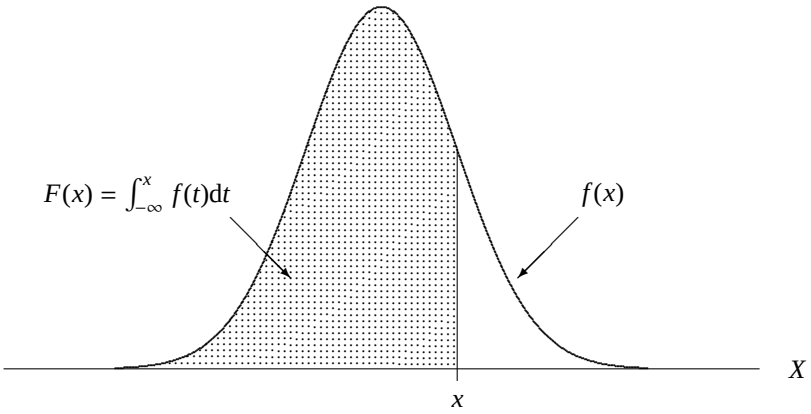
$$P(X \leq x) = F(x),$$

which represents the probability less than x. The properties of the distribution function $F(x)$ are given by:

$$\begin{aligned} &F(x_1) \leq F(x_2), \quad \text{for } x_1 < x_2, \\ &P(a < X \leq b) = F(b) - F(a), \quad \text{for } a < b, \\ &F(-\infty) = 0, \quad F(+\infty) = 1. \end{aligned}$$

The difference between the discrete and continuous random variables is given by:

Figure 1.2: Density Function $f(x)$ and Distribution Function $F(x)$
— Continuous Random Variable —



1. Discrete random variable (Figure 1.1):

   • $F(x) = \sum_{i=1}^{r} f(x_i) = \sum_{i=1}^{r} p_i$,
   where r denotes the integer which satisfies $x_r \leq x < x_{r+1}$.

   • $F(x_i) - F(x_i - \epsilon) = f(x_i) = p_i$,
   where ϵ is a small positive number less than $x_i - x_{i-1}$.

2. Continuous random variable (Figure 1.2):

   • $F(x) = \int_{-\infty}^{x} f(t)\, \mathrm{d}t$,

   • $F'(x) = f(x)$.

$f(x)$ and $F(x)$ are displayed in Figure 1.1 for a discrete random variable and Figure 1.2 for a continuous random variable.

1.2.2 Multivariate Random Variable and Distribution

We consider two random variables X and Y in this section. It is easy to extend to more than two random variables.

Discrete Random Variables: Suppose that discrete random variables X and Y take $x_1, x_2, \cdots$ and $y_1, y_2, \cdots$, respectively. The probability which event $\{\omega; X(\omega) = x_i$ and $Y(\omega) = y_j\}$ occurs is given by:

$$P(X = x_i, Y = y_j) = f_{xy}(x_i, y_j),$$

where $f_{xy}(x_i, y_j)$ represents the **joint probability function** of X and Y. In order for $f_{xy}(x_i, y_j)$ to be a joint probability function, $f_{xy}(x_i, y_j)$ has to satisfies the following properties:

$$f_{xy}(x_i, y_j) \geq 0, \quad i, j = 1, 2, \cdots$$
$$\sum_i \sum_j f_{xy}(x_i, y_j) = 1.$$

Define $f_x(x_i)$ and $f_y(y_j)$ as:

$$f_x(x_i) = \sum_j f_{xy}(x_i, y_j), \quad i = 1, 2, \cdots,$$
$$f_y(y_j) = \sum_i f_{xy}(x_i, y_j), \quad j = 1, 2, \cdots.$$

Then, $f_x(x_i)$ and $f_y(y_j)$ are called the **marginal probability functions** of X and Y. $f_x(x_i)$ and $f_y(y_j)$ also have the properties of the probability functions, i.e., $f_x(x_i) \geq 0$ and $\sum_i f_x(x_i) = 1$, and $f_y(y_j) \geq 0$ and $\sum_i f_y(y_j) = 1$.

Continuous Random Variables: Consider two continuous random variables X and Y. For a domain D, the probability which event $\{\omega; (X(\omega), Y(\omega)) \in D\}$ occurs is given by:

$$P((X, Y) \in D) = \iint_D f_{xy}(x, y)\, \mathrm{d}x\, \mathrm{d}y,$$

where $f_{xy}(x, y)$ is called the **joint probability density function** of X and Y or the **joint density function** of X and Y. $f_{xy}(x, y)$ has to satisfy the following properties:

$$f_{xy}(x, y) \geq 0,$$
$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_{xy}(x, y)\, \mathrm{d}x\, \mathrm{d}y = 1.$$

Define $f_x(x)$ and $f_y(y)$ as:

$$f_x(x) = \int_{-\infty}^{\infty} f_{xy}(x, y)\, \mathrm{d}y, \qquad \text{for all } x \text{ and } y,$$
$$f_y(y) = \int_{-\infty}^{\infty} f_{xy}(x, y)\, \mathrm{d}x,$$

where $f_x(x)$ and $f_y(y)$ are called the **marginal probability density functions** of X and Y or the **marginal density functions** of X and Y.

For example, consider the event $\{\omega; a < X(\omega) < b,\ c < Y(\omega) < d\}$, which is a specific case of the domain D. Then, the probability that we have the event $\{\omega; a < X(\omega) < b,\ c < Y(\omega) < d\}$ is written as:

$$P(a < X < b,\ c < Y < d) = \int_a^b \int_c^d f_{xy}(x, y)\, \mathrm{d}x\, \mathrm{d}y.$$

The mixture of discrete and continuous random variables is also possible. For example, let X be a discrete random variable and Y be a continuous random variable. X takes $x_1, x_2, \cdots$. The probability which both X takes x_i and Y takes real numbers within the interval I is given by:

$$P(X = x_i, Y \in I) = \int_I f_{xy}(x_i, y)\, \mathrm{d}y.$$

Then, we have the following properties:

$$f_{xy}(x_i, y) \geq 0, \quad \text{for all } y \text{ and } i = 1, 2, \cdots,$$
$$\sum_i \int_{-\infty}^{\infty} f_{xy}(x_i, y)\, \mathrm{d}y = 1.$$

The marginal probability function of X is given by:

$$f_x(x_i) = \int_{-\infty}^{\infty} f_{xy}(x_i, y)\, \mathrm{d}y,$$

for $i = 1, 2, \cdots$. The marginal probability density function of Y is:

$$f_y(y) = \sum_i f_{xy}(x_i, y).$$

1.2.3 Conditional Distribution

Discrete Random Variable: The **conditional probability function** of X given $Y = y_j$ is represented as:

$$P(X = x_i | Y = y_j) = f_{x|y}(x_i|y_j) = \frac{f_{xy}(x_i, y_j)}{f_y(y_j)} = \frac{f_{xy}(x_i, y_j)}{\sum_i f_{xy}(x_i, y_j)}.$$

The second equality indicates the definition of the conditional probability, which is shown in Section 1.1.2. The features of the conditional probability function $f_{x|y}(x_i|y_j)$ are:

$$f_{x|y}(x_i|y_j) \geq 0, \quad i = 1, 2, \cdots,$$
$$\sum_i f_{x|y}(x_i|y_j) = 1, \quad \text{for any } j.$$

Continuous Random Variable: The **conditional probability density function** of X given $Y = y$ (or the **conditional density function** of X given $Y = y$) is:

$$f_{x|y}(x|y) = \frac{f_{xy}(x, y)}{f_y(y)} = \frac{f_{xy}(x, y)}{\int_{-\infty}^{\infty} f_{xy}(x, y)\, \mathrm{d}x}.$$

The properties of the conditional probability density function $f_{x|y}(x|y)$ are given by:

$$f_{x|y}(x|y) \geq 0,$$
$$\int_{-\infty}^{\infty} f_{x|y}(x|y)\, \mathrm{d}x = 1, \quad \text{for any } Y = y.$$

Independence of Random Variables: For discrete random variables X and Y, we say that X is **independent** (or **stochastically independent**) of Y if and only if $f_{xy}(x_i, y_j) = f_x(x_i) f_y(y_j)$. Similarly, for continuous random variables X and Y, we say that X is independent of Y if and only if $f_{xy}(x, y) = f_x(x) f_y(y)$.

When X and Y are stochastically independent, $g(X)$ and $h(Y)$ are also stochastically independent, where $g(X)$ and $h(Y)$ are functions of X and Y.

1.3 Mathematical Expectation

1.3.1 Univariate Random Variable

Definition of Mathematical Expectation: Let $g(X)$ be a function of random variable X. The mathematical expectation of $g(X)$, denoted by $\mathrm{E}(g(X))$, is defined as follows:

$$\mathrm{E}(g(X)) = \begin{cases} \displaystyle\sum_i g(x_i) p_i = \sum_i g(x_i) f(x_i), & \text{(Discrete Random Variable),} \\ \displaystyle\int_{-\infty}^{\infty} g(x) f(x)\, \mathrm{d}x, & \text{(Continuous Random Variable).} \end{cases}$$

The following three functional forms of $g(X)$ are important.

1. $\boldsymbol{g(X) = X}$.

   The expectation of X, $\mathrm{E}(X)$, is known as **mean** of random variable X.

   $$\begin{aligned} \mathrm{E}(X) &= \begin{cases} \displaystyle\sum_i x_i f(x_i), & \text{(Discrete Random Variable),} \\ \displaystyle\int_{-\infty}^{\infty} x f(x)\, \mathrm{d}x, & \text{(Continuous Random Variable),} \end{cases} \\ &= \mu, \quad (\text{or } \mu_x). \end{aligned}$$

   When a distribution of X is symmetric, mean indicates the center of the distribution.

2. $\boldsymbol{g(X) = (X - \mu)^2}$.

   The expectation of $(X - \mu)^2$ is known as **variance** of random variable X, which is denoted by $\mathrm{V}(X)$.

   $$\begin{aligned} \mathrm{V}(X) &= \mathrm{E}((X - \mu)^2) \\ &= \begin{cases} \displaystyle\sum_i (x_i - \mu)^2 f(x_i), & \text{(Discrete Random Variable),} \\ \displaystyle\int_{-\infty}^{\infty} (x - \mu)^2 f(x)\, \mathrm{d}x, & \text{(Continuous Random Variable),} \end{cases} \\ &= \sigma^2, \quad (\text{or } \sigma_x^2). \end{aligned}$$

If X is broadly distributed, $\sigma^2 = \mathrm{V}(X)$ becomes large. Conversely, if the distribution is concentrated on the center, σ^2 becomes small. Note that $\sigma = \sqrt{V(X)}$ is called the **standard deviation**.

3. $\boldsymbol{g(X) = e^{\theta X}}$.

   The expectation of $e^{\theta X}$ is called the **moment-generating function**, which is denoted by $\phi(\theta)$.

$$\begin{aligned}\phi(\theta) &= \mathrm{E}(e^{\theta X}) \\ &= \begin{cases} \displaystyle\sum_i e^{\theta x_i} f(x_i), & \text{(Discrete Random Variable)}, \\ \displaystyle\int_{-\infty}^{\infty} e^{\theta x} f(x)\,\mathrm{d}x, & \text{(Continuous Random Variable)}. \end{cases}\end{aligned}$$

   Note that the definition of e is given by:

$$\begin{aligned} e &= \lim_{x\to 0}(1+x)^{\frac{1}{x}} = \lim_{h\to\infty}\Big(1+\frac{1}{h}\Big)^h \\ &= 2.71828182845905. \end{aligned}$$

   The moment-generating function plays an important roll in statistics, which is discussed in Section 1.5.

In Examples 1.5 – 1.8, mean, variance and the moment-generating function are computed.

Example 1.5: In Example 1.2 of flipping a coin three times (Section 1.1.1), we see in Section 1.2.1 that the probability function is written as the following binomial distribution:

$$P(X = x) = f(x) = \frac{n!}{x!(n-x)!} p^x (1-p)^{n-x}, \quad \text{for } x = 0, 1, 2, \cdots, n,$$

where $n = 3$ and $p = 1/2$. When X has the binomial distribution above, we obtain $\mathrm{E}(X)$, $\mathrm{V}(X)$ and $\phi(\theta)$ as follows.

First, $\mu = \mathrm{E}(X)$ is computed as:

$$\begin{aligned}\mu = \mathrm{E}(X) &= \sum_x x f(x) = \sum_x x \frac{n!}{x!(n-x)!} p^x (1-p)^{n-x} \\ &= \sum_x \frac{n!}{(x-1)!(n-x)!} p^x (1-p)^{n-x} = np \sum_x \frac{(n-1)!}{(x-1)!(n-x)!} p^{x-1} (1-p)^{n-x} \\ &= np \sum_{x'} \frac{n'!}{x'!(n'-x')!} p^{x'} (1-p)^{n'-x'} = np,\end{aligned}$$

where $n' = n - 1$ and $x' = x - 1$ are set.

Second, in order to obtain $\sigma^2 = \mathrm{V}(X)$, we rewrite $\mathrm{V}(X)$ as:

$$\sigma^2 = \mathrm{V}(X) = \mathrm{E}(X^2) - \mu^2 = \mathrm{E}(X(X-1)) + \mu - \mu^2.$$

$\mathrm{E}(X(X-1))$ is given by:

$$\begin{aligned}
\mathrm{E}(X(X-1)) &= \sum_x x(x-1)f(x) = \sum_x x(x-1)\frac{n!}{x!(n-x)!}p^x(1-p)^{n-x} \\
&= \sum_x \frac{n!}{(x-2)!(n-x)!}p^x(1-p)^{n-x} \\
&= n(n-1)p^2 \sum_x \frac{(n-2)!}{(x-2)!(n-x)!}p^{x-2}(1-p)^{n-x} \\
&= n(n-1)p^2 \sum_{x'} \frac{n'!}{x'!(n'-x')!}p^{x'}(1-p)^{n'-x'} = n(n-1)p^2,
\end{aligned}$$

where $n' = n-2$ and $x' = x-2$ are re-defined. Therefore, $\sigma^2 = \mathrm{V}(X)$ is obtained as:

$$\begin{aligned}
\sigma^2 = \mathrm{V}(X) &= \mathrm{E}(X(X-1)) + \mu - \mu^2 \\
&= n(n-1)p^2 + np - n^2p^2 = -np^2 + np = np(1-p).
\end{aligned}$$

Finally, the moment-generating function $\phi(\theta)$ is represented as:

$$\begin{aligned}
\phi(\theta) = \mathrm{E}(e^{\theta X}) &= \sum_x e^{\theta x}\frac{n!}{x!(n-x)!}p^x(1-p)^{n-p} \\
&= \sum_x \frac{n!}{x!(n-x)!}(pe^\theta)^x(1-p)^{n-p} = (pe^\theta + 1 - p)^n.
\end{aligned}$$

In the last equality, we utilize the following formula:

$$(a+b)^n = \sum_{x=0}^{n} \frac{n!}{x!(n-x)!}a^x b^{n-x},$$

which is called the **binomial theorem**.

Example 1.6: As an example of continuous random variables, in Section 1.2.1 the uniform distribution is introduced, which is given by:

$$f(x) = \begin{cases} 1, & \text{for } 0 < x < 1, \\ 0, & \text{otherwise.} \end{cases}$$

When X has the uniform distribution above, $\mathrm{E}(X)$, $\mathrm{V}(X)$ and $\phi(\theta)$ are computed as follows:

$$\mu = \mathrm{E}(X) = \int_{-\infty}^{\infty} x f(x)\,\mathrm{d}x = \int_0^1 x\,\mathrm{d}x = \left[\frac{1}{2}x^2\right]_0^1 = \frac{1}{2},$$

$$\sigma^2 = \mathrm{V}(X) = \mathrm{E}(X^2) - \mu^2$$
$$= \int_{-\infty}^{\infty} x^2 f(x)\,\mathrm{d}x - \mu^2 = \int_0^1 x^2\,\mathrm{d}x - \mu^2 = \Big[\frac{1}{3}x^3\Big]_0^1 - \Big(\frac{1}{2}\Big)^2 = \frac{1}{12},$$
$$\phi(\theta) = \mathrm{E}(e^{\theta X}) = \int_{-\infty}^{\infty} e^{\theta x} f(x)\,\mathrm{d}x = \int_0^1 e^{\theta x}\,\mathrm{d}x = [\frac{1}{\theta}e^{\theta x}]_0^1 = \frac{1}{\theta}(e^{\theta} - 1).$$

Example 1.7: As another example of continuous random variables, we take the standard normal distribution:

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}, \quad \text{for } -\infty < x < \infty,$$

which is discussed in Section 2.2.1. When X has a standard normal distribution, i.e., when $X \sim N(0,1)$, $\mathrm{E}(X)$, $\mathrm{V}(X)$ and $\phi(\theta)$ are as follows.

$\mathrm{E}(X)$ is obtained as:

$$\mathrm{E}(X) = \int_{-\infty}^{\infty} x f(x)\,\mathrm{d}x = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} x e^{-\frac{1}{2}x^2}\,\mathrm{d}x = \frac{1}{\sqrt{2\pi}} \Big[-e^{-\frac{1}{2}x^2}\Big]_{-\infty}^{\infty} = 0,$$

because $\lim_{x\to\pm\infty} -e^{-\frac{1}{2}x^2} = 0$.

$\mathrm{V}(X)$ is computed as follows:

$$\mathrm{V}(X) = \mathrm{E}(X^2) = \int_{-\infty}^{\infty} x^2 f(x)\,\mathrm{d}x = \int_{-\infty}^{\infty} x^2 \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}\,\mathrm{d}x = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} x \frac{\mathrm{d}(-e^{-\frac{1}{2}x^2})}{\mathrm{d}x}\,\mathrm{d}x$$
$$= \frac{1}{\sqrt{2\pi}} \Big[x(-e^{-\frac{1}{2}x^2})\Big]_{-\infty}^{\infty} + \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-\frac{1}{2}x^2}\,\mathrm{d}x = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}\,\mathrm{d}x = 1.$$

The first equality holds because of $\mathrm{E}(X) = 0$. In the fifth equality, use the following integration formula, called the **integration by parts**:

$$\int_a^b h(x)g'(x)\,\mathrm{d}x = \Big[h(x)g(x)\Big]_a^b - \int_a^b h'(x)g(x)\,\mathrm{d}x,$$

where we take $h(x) = x$ and $g(x) = -e^{-\frac{1}{2}x^2}$ in this case. See Appendix 1.2 for the integration by parts. In the sixth equality, $\lim_{x\to\pm\infty} -xe^{-\frac{1}{2}x^2} = 0$ is utilized. The last equality is because the integration of the standard normal probability density function is equal to one (see p.6 in Section 1.2.1 for the integration of the standard normal probability density function).

$\phi(\theta)$ is derived as follows:

$$\phi(\theta) = \int_{-\infty}^{\infty} e^{\theta x} f(x)\,\mathrm{d}x = \int_{-\infty}^{\infty} e^{\theta x} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}\,\mathrm{d}x = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2 + \theta x}\,\mathrm{d}x$$
$$= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\left((x-\theta)^2 - \theta^2\right)}\,\mathrm{d}x = e^{\frac{1}{2}\theta^2} \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(x-\theta)^2}\,\mathrm{d}x = e^{\frac{1}{2}\theta^2}.$$

The last equality holds because the integration indicates the normal density with mean θ and variance one. See Section 2.2.2 for the normal density.

Example 1.8: When the moment-generating function of X is given by $\phi_x(\theta) = e^{\frac{1}{2}\theta^2}$ (i.e., X has a standard normal distribution), we want to obtain the moment-generating function of $Y = \mu + \sigma X$.

Let $\phi_x(\theta)$ and $\phi_y(\theta)$ be the moment-generating functions of X and Y, respectively. Then, the moment-generating function of Y is obtained as follows:

$$\begin{aligned}\phi_y(\theta) &= \mathrm{E}(e^{\theta Y}) = \mathrm{E}(e^{\theta(\mu+\sigma X)}) = e^{\theta\mu}\mathrm{E}(e^{\theta\sigma X}) = e^{\theta\mu}\phi_x(\theta\sigma) = e^{\theta\mu}e^{\frac{1}{2}\sigma^2\theta^2} \\ &= \exp(\mu\theta + \frac{1}{2}\sigma^2\theta^2).\end{aligned}$$

Some Formulas of Mean and Variance:

1. **Theorem:** $\mathrm{E}(aX + b) = a\mathrm{E}(X) + b$, where a and b are constant.

   Proof:

   When X is a discrete random variable,

   $$\begin{aligned}\mathrm{E}(aX + b) &= \sum_i (ax_i + b)f(x_i) = a\sum_i x_i f(x_i) + b\sum_i f(x_i) \\ &= a\mathrm{E}(X) + b.\end{aligned}$$

   Note that we have $\sum_i x_i f(x_i) = \mathrm{E}(X)$ from the definition of mean and $\sum_i f(x_i) = 1$ because $f(x_i)$ is a probability function.

   If X is a continuous random variable,

   $$\begin{aligned}\mathrm{E}(aX + b) &= \int_{-\infty}^{\infty} (ax + b)f(x)\,\mathrm{d}x = a\int_{-\infty}^{\infty} xf(x)\,\mathrm{d}x + b\int_{-\infty}^{\infty} f(x)\,\mathrm{d}x \\ &= a\mathrm{E}(X) + b.\end{aligned}$$

   Similarly, note that we have $\int_{-\infty}^{\infty} xf(x)\,\mathrm{d}x = \mathrm{E}(X)$ from the definition of mean and $\int_{-\infty}^{\infty} f(x)\,\mathrm{d}x = 1$ because $f(x)$ is a probability density function.

2. **Theorem:** $\mathrm{V}(X) = \mathrm{E}(X^2) - \mu^2$, where $\mu = \mathrm{E}(X)$.

   Proof:

   $\mathrm{V}(X)$ is rewritten as follows:

   $$\begin{aligned}\mathrm{V}(X) &= \mathrm{E}((X - \mu)^2) = \mathrm{E}(X^2 - 2\mu X - \mu^2) \\ &= \mathrm{E}(X^2) - 2\mu\mathrm{E}(X) + \mu^2 = \mathrm{E}(X^2) - \mu^2.\end{aligned}$$

   The first equality is due to the definition of variance.

3. **Theorem:** $\mathrm{V}(aX + b) = a^2\mathrm{V}(X)$, where a and b are constant.

Proof:

From the definition of the mathematical expectation, $\mathrm{V}(aX+b)$ is represented as:

$$\begin{aligned}\mathrm{V}(aX+b) &= \mathrm{E}\Big(((aX+b)-\mathrm{E}(aX+b))^2\Big) = \mathrm{E}\Big((aX-a\mu)^2\Big)\\ &= \mathrm{E}(a^2(X-\mu)^2) = a^2\mathrm{E}((X-\mu)^2) = a^2\mathrm{V}(X)\end{aligned}$$

The first and the fifth equalities are from the definition of variance. We use $\mathrm{E}(aX+b) = a\mu + b$ in the second equality.

4. **Theorem:** The random variable X is assumed to be distributed with mean $\mathrm{E}(X) = \mu$ and variance $\mathrm{V}(X) = \sigma^2$. Define $Z = (X-\mu)/\sigma$. Then, we have $\mathrm{E}(Z) = 0$ and $\mathrm{V}(Z) = 1$.

   Proof:

   $\mathrm{E}(X)$ and $\mathrm{V}(X)$ are obtained as:

$$\mathrm{E}(Z) = \mathrm{E}\Big(\frac{X-\mu}{\sigma}\Big) = \frac{\mathrm{E}(X)-\mu}{\sigma} = 0,$$

$$\mathrm{V}(Z) = \mathrm{V}\Big(\frac{1}{\sigma}X - \frac{\mu}{\sigma}\Big) = \frac{1}{\sigma^2}\mathrm{V}(X) = 1.$$

   The transformation from X to Z is known as normalization or standardization.

1.3.2 Bivariate Random Variable

Definition: Let $g(X,Y)$ be a function of random variables X and Y. The mathematical expectation of $g(X,Y)$, denoted by $\mathrm{E}(g(X,Y))$, is defined as:

$$\mathrm{E}(g(X,Y)) = \begin{cases} \displaystyle\sum_i \sum_j g(x_i,y_j) f(x_i,y_j), & \text{(Discrete Random Variables)},\\ \displaystyle\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y) f(x,y)\,\mathrm{d}x\,\mathrm{d}y, & \text{(Continuous Random Variables)}.\end{cases}$$

The following four functional forms are important, i.e., mean, variance, covariance and the moment-generating function.

1. $\boldsymbol{g(X,Y) = X}$:

   The expectation of random variable X, i.e., $\mathrm{E}(X)$, is given by:

$$\begin{aligned}\mathrm{E}(X) &= \begin{cases} \displaystyle\sum_i \sum_j x_i f(x_i,y_j), & \text{(Discrete Random Variables)},\\ \displaystyle\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x f(x,y)\,\mathrm{d}x\,\mathrm{d}y, & \text{(Continuous Random Variables)},\end{cases}\\ &= \mu_x.\end{aligned}$$

   The case of $g(X,Y) = Y$ is exactly the same formulation as above, i.e., $\mathrm{E}(Y) = \mu_y$.

2. $\boldsymbol{g(X,Y) = (X-\mu_x)^2}$:

   The expectation of $(X-\mu_x)^2$ is known as variance of random variable X, which is denoted by $\mathrm{V}(X)$ and represented as follows:

$$\begin{aligned}\mathrm{V}(X) &= \mathrm{E}((X-\mu_x)^2)\\ &= \begin{cases}\displaystyle\sum_i\sum_j (x_i-\mu_x)^2 f(x_i,y_j), & \text{(Discrete Case)},\\ \displaystyle\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x-\mu_x)^2 f(x,y)\,\mathrm{d}x\,\mathrm{d}y, & \text{(Continuous Case)},\end{cases}\\ &= \sigma_x^2.\end{aligned}$$

   The variance of Y is also obtained in the same fashion, i.e., $\mathrm{V}(Y) = \sigma_y^2$.

3. $\boldsymbol{g(X,Y) = (X-\mu_x)(Y-\mu_y)}$:

   The expectation of $(X-\mu_x)(Y-\mu_y)$ is known as **covariance** of X and Y, which is denoted by $\mathrm{Cov}(X,Y)$ and written as:

$$\begin{aligned}\mathrm{Cov}(X,Y) &= \mathrm{E}((X-\mu_x)(Y-\mu_y))\\ &= \begin{cases}\displaystyle\sum_i\sum_j (x_i-\mu_x)(y_j-\mu_y) f(x_i,y_j), & \text{(Discrete Case)},\\ \displaystyle\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x-\mu_x)(y-\mu_y) f(x,y)\,\mathrm{d}x\,\mathrm{d}y, & \text{(Continuous Case)}.\end{cases}\end{aligned}$$

   Thus, covariance is defined in the case of bivariate random variables.

4. $\boldsymbol{g(X,Y) = e^{\theta_1 X+\theta_2 Y}}$:

   The mathematical expectation of $e^{\theta_1 X+\theta_2 Y}$ is called the moment-generating function, which is denoted by $\phi(\theta_1,\theta_2)$ and written as:

$$\begin{aligned}\phi(\theta_1,\theta_2) &= \mathrm{E}(e^{\theta_1 X+\theta_2 Y})\\ &= \begin{cases}\displaystyle\sum_i\sum_j e^{\theta_1 x_i+\theta_2 y_j} f(x_i,y_j), & \text{(Discrete Case)},\\ \displaystyle\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{\theta_1 x+\theta_2 y} f(x,y)\,\mathrm{d}x\,\mathrm{d}y, & \text{(Continuous Case)}.\end{cases}\end{aligned}$$

   In Section 1.5, the moment-generating function in the multivariate cases is discussed in more detail.

Some Formulas of Mean and Variance: We consider two random variables X and Y. Some formulas are shown as follows.

1. **Theorem:** $\mathrm{E}(X+Y) = \mathrm{E}(X) + \mathrm{E}(Y)$.

Proof:

For discrete random variables X and Y, it is given by:

$$\begin{aligned}\mathrm{E}(X+Y) &= \sum_i \sum_j (x_i + y_j) f_{xy}(x_i, y_j) \\ &= \sum_i \sum_j x_i f_{xy}(x_i, y_j) + \sum_i \sum_j y_j f_{xy}(x_i, y_j) \\ &= \mathrm{E}(X) + \mathrm{E}(Y).\end{aligned}$$

For continuous random variables X and Y, we can show:

$$\begin{aligned}\mathrm{E}(X+Y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (x+y) f_{xy}(x,y)\,\mathrm{d}x\,\mathrm{d}y \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x f_{xy}(x,y)\,\mathrm{d}x\,\mathrm{d}y + \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} y f_{xy}(x,y)\,\mathrm{d}x\,\mathrm{d}y \\ &= \mathrm{E}(X) + \mathrm{E}(Y).\end{aligned}$$

2. **Theorem:** $\mathrm{E}(XY) = \mathrm{E}(X)\mathrm{E}(Y)$, when X is independent of Y.

   Proof:

   For discrete random variables X and Y,

$$\begin{aligned}\mathrm{E}(XY) &= \sum_i \sum_j x_i y_j f_{xy}(x_i, y_j) = \sum_i \sum_j x_i y_j f_x(x_i) f_y(y_j) \\ &= \Big(\sum_i x_i f_x(x_i)\Big)\Big(\sum_j y_j f_y(y_j)\Big) = \mathrm{E}(X)\mathrm{E}(Y).\end{aligned}$$

   If X is independent of Y, the second equality holds, i.e., $f_{xy}(x_i, y_j) = f_x(x_i) f_y(y_j)$.

   For continuous random variables X and Y,

$$\begin{aligned}\mathrm{E}(XY) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} xy f_{xy}(x,y)\,\mathrm{d}x\,\mathrm{d}y \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} xy f_x(x) f_y(y)\,\mathrm{d}x\,\mathrm{d}y \\ &= \Big(\int_{-\infty}^{\infty} x f_x(x)\,\mathrm{d}x\Big)\Big(\int_{-\infty}^{\infty} y f_y(y)\,\mathrm{d}y\Big) = \mathrm{E}(X)\mathrm{E}(Y).\end{aligned}$$

   When X is independent of Y, we have $f_{xy}(x,y) = f_x(x) f_y(y)$ in the second equality.

3. **Theorem:** $\mathrm{Cov}(X, Y) = \mathrm{E}(XY) - \mathrm{E}(X)\mathrm{E}(Y)$.

Proof:

For both discrete and continuous random variables, we can rewrite as follows:

$$\begin{aligned}\mathrm{Cov}(X,Y) &= \mathrm{E}((X-\mu_x)(Y-\mu_y)) = \mathrm{E}(XY-\mu_xY-\mu_yX+\mu_x\mu_y)\\ &= \mathrm{E}(XY)-\mathrm{E}(\mu_xY)-\mathrm{E}(\mu_yX)+\mu_x\mu_y\\ &= \mathrm{E}(XY)-\mu_x\mathrm{E}(Y)-\mu_y\mathrm{E}(X)+\mu_x\mu_y\\ &= \mathrm{E}(XY)-\mu_x\mu_y-\mu_y\mu_x+\mu_x\mu_y = \mathrm{E}(XY)-\mu_x\mu_y\\ &= \mathrm{E}(XY)-\mathrm{E}(X)\mathrm{E}(Y).\end{aligned}$$

In the fourth equality, the theorem in Section 1.3.1 is used, i.e., $\mathrm{E}(\mu_xY) = \mu_x\mathrm{E}(Y)$ and $\mathrm{E}(\mu_yX) = \mu_y\mathrm{E}(X)$.

4. **Theorem:** $\mathrm{Cov}(X,Y) = 0$, when X is independent of Y.

   Proof:

   From the above two theorems, we have $\mathrm{E}(XY) = \mathrm{E}(X)\mathrm{E}(Y)$ when X is independent of Y and $\mathrm{Cov}(X,Y) = \mathrm{E}(XY) - \mathrm{E}(X)\mathrm{E}(Y)$. Therefore, $\mathrm{Cov}(X,Y) = 0$ is obtained when X is independent of Y.

5. **Definition:** The **correlation coefficient** between X and Y, denoted by ρ_{xy}, is defined as:

$$\rho_{xy} = \frac{\mathrm{Cov}(X,Y)}{\sqrt{\mathrm{V}(X)}\sqrt{\mathrm{V}(Y)}} = \frac{\mathrm{Cov}(X,Y)}{\sigma_x\sigma_y}.$$

   When $\rho_{xy} > 0$, we say that there is a **positive correlation** between X and Y. As ρ_{xy} approaches 1, we say that there is a **strong positive correlation** between X and Y. When $\rho_{xy} < 0$, we say that there is a **negative correlation** between X and Y. As ρ_{xy} approaches -1, we say that there is a **strong negative correlation** between X and Y.

6. **Theorem:** $\rho_{xy} = 0$, when X is independent of Y.

   Proof:

   When X is independent of Y, we have $\mathrm{Cov}(X,Y) = 0$. Therefore, we can obtain the result $\rho_{xy} = \dfrac{\mathrm{Cov}(X,Y)}{\sqrt{\mathrm{V}(X)}\sqrt{\mathrm{V}(Y)}} = 0$. However, note that $\rho_{xy} = 0$ does not mean the independence between X and Y.

7. **Theorem:** $\mathrm{V}(X \pm Y) = \mathrm{V}(X) \pm 2\mathrm{Cov}(X,Y) + \mathrm{V}(Y)$.

   Proof:

   For both discrete and continuous random variables, $\mathrm{V}(X \pm Y)$ is rewritten as follows:

$$\mathrm{V}(X \pm Y) = \mathrm{E}\Big(((X \pm Y) - \mathrm{E}(X \pm Y))^2\Big) = \mathrm{E}\Big(((X-\mu_x) \pm (Y-\mu_y))^2\Big)$$

$$
\begin{aligned}
&= \mathrm{E}((X-\mu_x)^2 \pm 2(X-\mu_x)(Y-\mu_y) + (Y-\mu_y)^2) \\
&= \mathrm{E}((X-\mu_x)^2) \pm 2\mathrm{E}((X-\mu_x)(Y-\mu_y)) + \mathrm{E}((Y-\mu_y)^2) \\
&= \mathrm{V}(X) \pm 2\mathrm{Cov}(X,Y) + \mathrm{V}(Y).
\end{aligned}
$$

8. **Theorem:** $-1 \le \rho_{xy} \le 1$.

   Proof:

   Consider the following function of t: $f(t) = \mathrm{V}(Xt - Y)$, which is always greater than or equal to zero because of the definition of variance. Therefore, for all t, we have $f(t) \ge 0$. $f(t)$ is rewritten as follows:

$$
\begin{aligned}
f(t) &= \mathrm{V}(Xt - Y) = \mathrm{V}(Xt) - 2\mathrm{Cov}(Xt, Y) + \mathrm{V}(Y) \\
&= t^2\mathrm{V}(X) - 2t\mathrm{Cov}(X,Y) + \mathrm{V}(Y) \\
&= \mathrm{V}(X)\Big(t - \frac{\mathrm{Cov}(X,Y)}{\mathrm{V}(X)}\Big)^2 + \mathrm{V}(Y) - \frac{(\mathrm{Cov}(X,Y))^2}{\mathrm{V}(X)}.
\end{aligned}
$$

   In order to have $f(t) \ge 0$ for all t, we need the following condition:

$$
\mathrm{V}(Y) - \frac{(\mathrm{Cov}(X,Y))^2}{\mathrm{V}(X)} \ge 0,
$$

   because the first term in the last equality is nonnegative, which implies:

$$
\frac{(\mathrm{Cov}(X,Y))^2}{\mathrm{V}(X)\mathrm{V}(Y)} \le 1.
$$

   Therefore, we have:

$$
-1 \le \frac{\mathrm{Cov}(X,Y)}{\sqrt{\mathrm{V}(X)}\sqrt{\mathrm{V}(Y)}} \le 1.
$$

   From the definition of correlation coefficient, i.e., $\rho_{xy} = \dfrac{\mathrm{Cov}(X,Y)}{\sqrt{\mathrm{V}(X)}\sqrt{\mathrm{V}(Y)}}$, we obtain the result: $-1 \le \rho_{xy} \le 1$.

9. **Theorem:** $\mathrm{V}(X \pm Y) = \mathrm{V}(X) + \mathrm{V}(Y)$, when X is independent of Y.

   Proof:

   From the theorem above, $\mathrm{V}(X \pm Y) = \mathrm{V}(X) \pm 2\mathrm{Cov}(X,Y) + \mathrm{V}(Y)$ generally holds. When random variables X and Y are independent, we have $\mathrm{Cov}(X,Y) = 0$. Therefore, $\mathrm{V}(X+Y) = \mathrm{V}(X) + \mathrm{V}(Y)$ holds, when X is independent of Y.

10. **Theorem:** For n random variables $X_1, X_2, \cdots, X_n$,

$$
\begin{aligned}
\mathrm{E}(\sum_i a_i X_i) &= \sum_i a_i \mu_i, \\
\mathrm{V}(\sum_i a_i X_i) &= \sum_i \sum_j a_i a_j \mathrm{Cov}(X_i, X_j),
\end{aligned}
$$

where $\mathrm{E}(X_i) = \mu_i$ and a_i is a constant value. Especially, when $X_1, X_2, \cdots, X_n$ are mutually independent, we have the following:

$$\mathrm{V}(\sum_i a_i X_i) = \sum_i a_i^2 \mathrm{V}(X_i).$$

Proof:

For mean of $\sum_i a_i X_i$, the following representation is obtained.

$$\mathrm{E}(\sum_i a_i X_i) = \sum_i \mathrm{E}(a_i X_i) = \sum_i a_i \mathrm{E}(X_i) = \sum_i a_i \mu_i.$$

The first and second equalities come from the previous theorems on mean.

For variance of $\sum_i a_i X_i$, we can rewrite as follows:

$$\begin{aligned}\mathrm{V}(\sum_i a_i X_i) &= \mathrm{E}\Big(\sum_i a_i (X_i - \mu_i)\Big)^2 = \mathrm{E}\Big(\sum_i a_i (X_i - \mu_i)\Big)\Big(\sum_j a_j (X_j - \mu_j)\Big) \\ &= \mathrm{E}\Big(\sum_i \sum_j a_i a_j (X_i - \mu_i)(X_j - \mu_j)\Big) \\ &= \sum_i \sum_j a_i a_j \mathrm{E}\Big((X_i - \mu_i)(X_j - \mu_j)\Big) = \sum_i \sum_j a_i a_j \mathrm{Cov}(X_i, X_j).\end{aligned}$$

When $X_1, X_2, \cdots, X_n$ are mutually independent, we obtain $\mathrm{Cov}(X_i, X_j) = 0$ for all $i \neq j$ from the previous theorem. Therefore, we obtain:

$$\mathrm{V}(\sum_i a_i X_i) = \sum_i a_i^2 \mathrm{V}(X_i).$$

Note that $\mathrm{Cov}(X_i, X_i) = \mathrm{E}((X_i - \mu)^2) = \mathrm{V}(X_i)$.

11. **Theorem:** n random variables $X_1, X_2, \cdots, X_n$ are mutually independently and identically distributed with mean μ and variance σ^2. That is, for all $i = 1, 2, \cdots, n$, $\mathrm{E}(X_i) = \mu$ and $\mathrm{V}(X_i) = \sigma^2$ are assumed. Consider arithmetic average $\overline{X} = (1/n)\sum_{i=1}^n X_i$. Then, mean and variance of $\overline{X}$ are given by:

$$\mathrm{E}(\overline{X}) = \mu, \qquad \mathrm{V}(\overline{X}) = \frac{\sigma^2}{n}.$$

Proof:

The mathematical expectation of $\overline{X}$ is given by:

$$\mathrm{E}(\overline{X}) = \mathrm{E}(\frac{1}{n}\sum_{i=1}^n X_i) = \frac{1}{n}\mathrm{E}(\sum_{i=1}^n X_i) = \frac{1}{n}\sum_{i=1}^n \mathrm{E}(X_i) = \frac{1}{n}\sum_{i=1}^n \mu = \frac{1}{n} n\mu = \mu.$$

$\mathrm{E}(aX) = a\mathrm{E}(X)$ in the second equality and $\mathrm{E}(X + Y) = \mathrm{E}(X) + \mathrm{E}(Y)$ in the third equality are utilized, where X and Y are random variables and a is a constant value. For these formulas, see p.15 in Section 1.3.1 and p.17 in this section.

The variance of $\overline{X}$ is computed as follows:

$$\mathrm{V}(\overline{X}) = \mathrm{V}(\frac{1}{n}\sum_{i=1}^{n} X_i) = \frac{1}{n^2}\mathrm{V}(\sum_{i=1}^{n} X_i) = \frac{1}{n^2}\sum_{i=1}^{n}\mathrm{V}(X_i) = \frac{1}{n^2}\sum_{i=1}^{n}\sigma^2 = \frac{1}{n^2}n\sigma^2 = \frac{\sigma^2}{n}.$$

We use $\mathrm{V}(aX) = a^2\mathrm{V}(X)$ in the second equality and $\mathrm{V}(X+Y) = \mathrm{V}(X) + \mathrm{V}(Y)$ for X independent of Y in the third equality, where X and Y denote random variables and a is a constant value. For these formulas, see p.15 in Section 1.3.1 and p.20 in this section.

1.4 Transformation of Variables

Transformation of variables is used in the case of continuous random variables. Based on a distribution of a random variable, a distribution of the transformed random variable is derived. In other words, when a distribution of X is known, we can find a distribution of Y using the transformation of variables, where Y is a function of X.

1.4.1 Univariate Case

Distribution of $Y = \psi^{-1}(X)$: Let $f_x(x)$ be the probability density function of continuous random variable X and $X = \psi(Y)$ be a one-to-one transformation. Then, the probability density function of Y, i.e., $f_y(y)$, is given by:

$$f_y(y) = |\psi'(y)|\, f_x\Big(\psi(y)\Big).$$

We can derive the above transformation of variables from X to Y as follows. Let $f_x(x)$ and $F_x(x)$ be the probability density function and the distribution function of X, respectively. Note that $F_x(x) = P(X \le x)$ and $f_x(x) = F_x'(x)$.

When $X = \psi(Y)$, we want to obtain the probability density function of Y. Let $f_y(y)$ and $F_y(y)$ be the probability density function and the distribution function of Y, respectively.

In the case of $\psi'(X) > 0$, the distribution function of Y, $F_y(y)$, is rewritten as follows:

$$F_y(y) = P(Y \le y) = P\Big(\psi(Y) \le \psi(y)\Big) = P\Big(X \le \psi(y)\Big) = F_x\Big(\psi(y)\Big).$$

The first equality is the definition of the cumulative distribution function. The second equality holds because of $\psi'(Y) > 0$. Therefore, differentiating $F_y(y)$ with respect to y, we can obtain the following expression:

$$f_y(y) = F_y'(y) = \psi'(y)F_x'\Big(\psi(y)\Big) = \psi'(y)\, f_x\Big(\psi(y)\Big). \tag{1.4}$$

Next, in the case of $\psi'(X) < 0$, the distribution function of Y, $F_y(y)$, is rewritten as follows:

$$\begin{aligned} F_y(y) &= P(Y \le y) = P\Big(\psi(Y) \ge \psi(y)\Big) = P\Big(X \ge \psi(y)\Big) \\ &= 1 - P\Big(X < \psi(y)\Big) = 1 - F_x\Big(\psi(y)\Big). \end{aligned}$$

Thus, in the case of $\psi'(X) < 0$, pay attention to the second equality, where the inequality sign is reversed. Differentiating $F_y(y)$ with respect to y, we obtain the following result:

$$f_y(y) = F'_y(y) = -\psi'(y)F'_x\Big(\psi(y)\Big) = -\psi'(y)\, f_x\Big(\psi(y)\Big). \tag{1.5}$$

Note that $-\psi'(y) > 0$.

Thus, summarizing the above two cases, i.e., $\psi'(X) > 0$ and $\psi'(X) < 0$, equations (1.4) and (1.5) indicate the following result:

$$f_y(y) = |\psi'(y)|\, f_x\Big(\psi(y)\Big),$$

which is called the **transformation of variables**.

Example 1.9: When X has a standard normal density function, i.e., when $X \sim N(0, 1)$, we derive the probability density function of Y, where $Y = \mu + \sigma X$.

Since we have:

$$X = \psi(Y) = \frac{Y - \mu}{\sigma},$$

$\psi'(y) = 1/\sigma$ is obtained. Therefore, the density function of Y, $f_y(y)$, is given by:

$$f_y(y) = |\psi'(y)|\, f_x\Big(\psi(y)\Big) = \frac{1}{\sigma\sqrt{2\pi}} \exp\Big(-\frac{1}{2\sigma^2}(y - \mu)^2\Big),$$

which indicates the normal distribution with mean μ and variance σ^2, denoted by $N(\mu, \sigma^2)$.

On Distribution of $Y = X^2$: As an example, when we know the distribution function of X as $F_x(x)$, we want to obtain the distribution function of Y, $F_y(y)$, where $Y = X^2$. Using $F_x(x)$, $F_y(y)$ is rewritten as follows:

$$\begin{aligned} F_y(y) &= P(Y \le y) = P(X^2 \le y) = P(-\sqrt{y} \le X \le \sqrt{y}) \\ &= F_x(\sqrt{y}) - F_x(-\sqrt{y}). \end{aligned}$$

Therefore, when we have $f_x(x)$ and $Y = X^2$, the probability density function of Y is obtained as follows:

$$f_y(y) = F'_y(y) = \frac{1}{2\sqrt{y}}\Big(f_x(\sqrt{y}) + f_x(-\sqrt{y})\Big).$$

1.4.2 Multivariate Cases

Bivariate Case: Let $f_{xy}(x, y)$ be a joint probability density function of X and Y. Let $X = \psi_1(U, V)$ and $Y = \psi_2(U, V)$ be a one-to-one transformation from (X, Y) to (U, V). Then, we obtain a joint probability density function of U and V, denoted by $f_{uv}(u, v)$, as follows:

$$f_{uv}(u, v) = |J| f_{xy}\Big(\psi_1(u, v), \psi_2(u, v)\Big),$$

where J is called the **Jacobian** of the transformation, which is defined as:

$$J = \begin{vmatrix} \dfrac{\partial x}{\partial u} & \dfrac{\partial x}{\partial v} \\ \dfrac{\partial y}{\partial u} & \dfrac{\partial y}{\partial v} \end{vmatrix}.$$

Multivariate Case: Let $f_x(x_1, x_2, \cdots, x_n)$ be a joint probability density function of X_1, X_2, $\cdots$ X_n. Suppose that a one-to-one transformation from $(X_1, X_2, \cdots, X_n)$ to $(Y_1, Y_2, \cdots, Y_n)$ is given by:

$$\begin{aligned} X_1 &= \psi_1(Y_1, Y_2, \cdots, Y_n), \\ X_2 &= \psi_2(Y_1, Y_2, \cdots, Y_n), \\ &\vdots \\ X_n &= \psi_n(Y_1, Y_2, \cdots, Y_n). \end{aligned}$$

Then, we obtain a joint probability density function of Y_1, Y_2, $\cdots$, Y_n, denoted by $f_y(y_1, y_2, \cdots, y_n)$, as follows:

$$f_y(y_1, y_2, \cdots, y_n) = |J| f_x\Big(\psi_1(y_1, \cdots, y_n), \psi_2(y_1, \cdots, y_n), \cdots, \psi_n(y_1, \cdots, y_n)\Big),$$

where J is called the Jacobian of the transformation, which is defined as:

$$J = \begin{vmatrix} \dfrac{\partial x_1}{\partial y_1} & \dfrac{\partial x_1}{\partial y_2} & \cdots & \dfrac{\partial x_1}{\partial y_n} \\ \dfrac{\partial x_2}{\partial y_1} & \dfrac{\partial x_2}{\partial y_2} & \cdots & \dfrac{\partial x_2}{\partial y_n} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial x_n}{\partial y_1} & \dfrac{\partial x_n}{\partial y_2} & \cdots & \dfrac{\partial x_n}{\partial y_n} \end{vmatrix}.$$

1.5 Moment-Generating Function

1.5.1 Univariate Case

As discussed in Section 1.3.1, the moment-generating function is defined as $\phi(\theta) = \mathrm{E}(e^{\theta X})$. In this section, several important theorems and remarks of the moment-generating function are summarized.

For a random variable X, $\mu_n' \equiv \mathrm{E}(X^n)$ is called the ***n*th moment** of X. Then, we have the following first theorem.

1. **Theorem:** $\phi^{(n)}(0) = \mu_n' \equiv \mathrm{E}(X^n)$.

   Proof:

   First, from the definition of the moment-generating function, $\phi(\theta)$ is written as:

   $$\phi(\theta) = \mathrm{E}(e^{\theta X}) = \int_{-\infty}^{\infty} e^{\theta x} f(x)\, \mathrm{d}x.$$

   The nth derivative of $\phi(\theta)$, denoted by $\phi^{(n)}(\theta)$, is:

   $$\phi^{(n)}(\theta) = \int_{-\infty}^{\infty} x^n e^{\theta x} f(x)\, \mathrm{d}x.$$

   Evaluating $\phi^{(n)}(\theta)$ at $\theta = 0$, we obtain:

   $$\phi^{(n)}(0) = \int_{-\infty}^{\infty} x^n f(x)\, \mathrm{d}x = \mathrm{E}(X^n) \equiv \mu_n',$$

   where the second equality comes from the definition of the mathematical expectation.

2. **Remark:** Let X and Y be two random variables. When the moment-generating function of X is equivalent to that of Y, we have the fact that X has the same distribution as Y.

3. **Theorem:** Let $\phi(\theta)$ be the moment-generating function of X. Then, the moment-generating function of Y, where $Y = aX + b$, is given by $e^{b\theta}\phi(a\theta)$.

   Proof:

   Let $\phi_y(\theta)$ be the moment-generating function of Y. Then, $\phi_y(\theta)$ is rewritten as follows:

   $$\phi_y(\theta) = \mathrm{E}(e^{\theta Y}) = \mathrm{E}(e^{\theta(aX+b)}) = e^{b\theta}\mathrm{E}(e^{a\theta X}) = e^{b\theta}\phi(a\theta).$$

   Note that $\phi(\theta)$ represents the moment-generating function of X.

4. **Theorem:** Let $\phi_1(\theta)$, $\phi_2(\theta)$, $\cdots$, $\phi_n(\theta)$ be the moment-generating functions of X_1, X_2, $\cdots$, X_n, which are mutually independently distributed random variables. Define $Y = X_1 + X_2 + \cdots + X_n$. Then, the moment-generating function of Y is given by $\phi_1(\theta)\phi_2(\theta)\cdots\phi_n(\theta)$, i.e.,

   $$\phi_y(\theta) = \mathrm{E}(e^{\theta Y}) = \phi_1(\theta)\phi_2(\theta)\cdots\phi_n(\theta),$$

   where $\phi_y(\theta)$ represents the moment-generating function of Y.

Proof:

The moment-generating function of Y, i.e., $\phi_y(\theta)$, is rewritten as:

$$\begin{aligned}\phi_y(\theta) &= \mathrm{E}(e^{\theta Y}) = \mathrm{E}(e^{\theta(X_1+X_2+\cdots+X_n)}) = \mathrm{E}(e^{\theta X_1})\mathrm{E}(e^{\theta X_2})\cdots\mathrm{E}(e^{\theta X_n}) \\ &= \phi_1(\theta)\phi_2(\theta)\cdots\phi_n(\theta).\end{aligned}$$

The third equality holds because X_1, X_2, $\cdots$, X_n are mutually independently distributed random variables.

5. **Theorem:** When X_1, X_2, $\cdots$, X_n are mutually independently and identically distributed and the moment-generating function of X_i is given by $\phi(\theta)$ for all i, the moment-generating function of Y is represented by $\Big(\phi(\theta)\Big)^n$, where $Y = X_1 + X_2 + \cdots + X_n$.

   Proof:

   Using the above theorem, we have the following:

$$\phi_y(\theta) = \phi_1(\theta)\phi_2(\theta)\cdots\phi_n(\theta) = \phi(\theta)\phi(\theta)\cdots\phi(\theta) = \Big(\phi(\theta)\Big)^n.$$

   Note that $\phi_i(\theta) = \phi(\theta)$ for all i.

6. **Theorem:** When X_1, X_2, $\cdots$, X_n are mutually independently and identically distributed and the moment-generating function of X_i is given by $\phi(\theta)$ for all i, the moment-generating function of $\overline{X}$ is represented by $\Big(\phi(\frac{\theta}{n})\Big)^n$, where $\overline{X} = (1/n)\sum_{i=1}^n X_i$.

   Proof:

   Let $\phi_{\overline{x}}(\theta)$ be the moment-generating function of $\overline{X}$.

$$\phi_{\overline{x}}(\theta) = \mathrm{E}(e^{\theta\overline{X}}) = \mathrm{E}(e^{\frac{\theta}{n}\sum_{i=1}^n X_i}) = \prod_{i=1}^n \mathrm{E}(e^{\frac{\theta}{n}X_i}) = \prod_{i=1}^n \phi(\frac{\theta}{n}) = \Big(\phi(\frac{\theta}{n})\Big)^n.$$

Example 1.10: For the binomial random variable, the moment-generating function $\phi(\theta)$ is known as:

$$\phi(\theta) = (pe^\theta + 1 - p)^n,$$

which is discussed in Example 1.5 (Section 1.3.1). Using the moment-generating function, we check whether $\mathrm{E}(X) = np$ and $\mathrm{V}(X) = np(1-p)$ are obtained when X is a binomial random variable.

The first- and the second-derivatives with respect to θ are given by:

$$\begin{aligned}\phi'(\theta) &= npe^\theta(pe^\theta + 1 - p)^{n-1}, \\ \phi''(\theta) &= npe^\theta(pe^\theta + 1 - p)^{n-1} + n(n-1)p^2e^{2\theta}(pe^\theta + 1 - p)^{n-2}.\end{aligned}$$

Evaluating at $\theta = 0$, we have:

$$\mathrm{E}(X) = \phi'(0) = np, \quad \mathrm{E}(X^2) = \phi''(0) = np + n(n-1)p^2.$$

Therefore, $\mathrm{V}(X) = \mathrm{E}(X^2) - (\mathrm{E}(X))^2 = np(1-p)$ can be derived. Thus, we can make sure that $\mathrm{E}(X)$ and $\mathrm{V}(X)$ are obtained from $\phi(\theta)$.

1.5.2 Multivariate Cases

Bivariate Case: As discussed in Section 1.3.2, for two random variables X and Y, the moment-generating function is defined as $\phi(\theta_1, \theta_2) = \mathrm{E}(e^{\theta_1 X + \theta_2 Y})$. Some useful and important theorems and remarks are shown as follows.

1. **Theorem:** Consider two random variables X and Y. Let $\phi(\theta_1, \theta_2)$ be the moment-generating function of X and Y. Then, we have the following result:

$$\frac{\partial^{j+k}\phi(0,0)}{\partial\theta_1^j \partial\theta_2^k} = \mathrm{E}(X^j Y^k).$$

Proof:

Let $f_{xy}(x, y)$ be the probability density function of X and Y. From the definition, $\phi(\theta_1, \theta_2)$ is written as:

$$\phi(\theta_1, \theta_2) = \mathrm{E}(e^{\theta_1 X + \theta_2 Y}) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{\theta_1 x + \theta_2 y} f_{xy}(x, y)\,\mathrm{d}x\,\mathrm{d}y.$$

Taking the jth derivative of $\phi(\theta_1, \theta_2)$ with respect to θ_1 and at the same time the kth derivative with respect to θ_2, we have the following expression:

$$\frac{\partial^{j+k}\phi(\theta_1,\theta_2)}{\partial\theta_1^j \partial\theta_2^k} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^j y^k e^{\theta_1 x + \theta_2 y} f_{xy}(x, y)\,\mathrm{d}x\,\mathrm{d}y.$$

Evaluating the above equation at $(\theta_1, \theta_2) = (0, 0)$, we can easily obtain:

$$\frac{\partial^{j+k}\phi(0,0)}{\partial\theta_1^j \partial\theta_2^k} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^j y^k f_{xy}(x, y)\,\mathrm{d}x\,\mathrm{d}y \equiv \mathrm{E}(X^j Y^k).$$

2. **Remark:** Let (X_i, Y_i) be a pair of random variables. Suppose that the moment-generating function of (X_1, Y_1) is equivalent to that of (X_2, Y_2). Then, (X_1, Y_1) has the same distribution function as (X_2, Y_2).

3. **Theorem:** Let $\phi(\theta_1, \theta_2)$ be the moment-generating function of (X, Y). The moment-generating function of X is given by $\phi_1(\theta_1)$ and that of Y is $\phi_2(\theta_2)$. Then, we have the following facts:

$$\phi_1(\theta_1) = \phi(\theta_1, 0), \quad \phi_2(\theta_2) = \phi(0, \theta_2).$$

Proof:

Again, the definition of the moment-generating function of X and Y is represented as:

$$\phi(\theta_1, \theta_2) = \mathrm{E}(e^{\theta_1 X + \theta_2 Y}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{\theta_1 x + \theta_2 y} f_{xy}(x, y)\, \mathrm{d}x\, \mathrm{d}y.$$

When $\phi(\theta_1, \theta_2)$ is evaluated at $\theta_2 = 0$, $\phi(\theta_1, 0)$ is rewritten as follows:

$$\begin{aligned}
\phi(\theta_1, 0) = \mathrm{E}(e^{\theta_1 X}) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{\theta_1 x} f_{xy}(x, y)\, \mathrm{d}x\, \mathrm{d}y \\
&= \int_{-\infty}^{\infty} e^{\theta_1 x} \Big(\int_{-\infty}^{\infty} f_{xy}(x, y)\, \mathrm{d}y \Big)\, \mathrm{d}x \\
&= \int_{-\infty}^{\infty} e^{\theta_1 x} f_x(x)\, \mathrm{d}x = \mathrm{E}(e^{\theta_1 X}) = \phi_1(\theta_1).
\end{aligned}$$

Thus, we obtain the result: $\phi(\theta_1, 0) = \phi_1(\theta_1)$. Similarly, $\phi(0, \theta_2) = \phi_2(\theta_2)$ can be derived.

4. **Theorem:** The moment-generating function of (X, Y) is given by $\phi(\theta_1, \theta_2)$. Let $\phi_1(\theta_1)$ and $\phi_2(\theta_2)$ be the moment-generating functions of X and Y, respectively. If X is independent of Y, we have:

$$\phi(\theta_1, \theta_2) = \phi_1(\theta_1)\phi_2(\theta_2).$$

Proof:

From the definition of $\phi(\theta_1, \theta_2)$, the moment-generating function of X and Y is rewritten as follows:

$$\phi(\theta_1, \theta_2) = \mathrm{E}(e^{\theta_1 X + \theta_2 Y}) = \mathrm{E}(e^{\theta_1 X})\mathrm{E}(e^{\theta_2 Y}) = \phi_1(\theta_1)\phi_2(\theta_2).$$

The second equality holds because X is independent of Y.

Multivariate Case: For multivariate random variables $X_1, X_2, \cdots, X_n$, the moment-generating function is defined as:

$$\phi(\theta_1, \theta_2, \cdots, \theta_n) = \mathrm{E}(e^{\theta_1 X_1 + \theta_2 X_2 + \cdots + \theta_n X_n}).$$

1. **Theorem:** If the multivariate random variables $X_1, X_2, \cdots, X_n$ are mutually independent, the moment-generating function of $X_1, X_2, \cdots, X_n$, denoted by $\phi(\theta_1, \theta_2, \cdots, \theta_n)$, is given by:

$$\phi(\theta_1, \theta_2, \cdots, \theta_n) = \phi_1(\theta_1)\phi_2(\theta_2) \cdots \phi_n(\theta_n),$$

where $\phi_i(\theta) = \mathrm{E}(e^{\theta X_i})$.

Proof:

From the definition of the moment-generating function in the multivariate cases, we obtain the following:

$$\begin{aligned}\phi(\theta_1, \theta_2, \cdots, \theta_n) &= \mathrm{E}(e^{\theta_1 X_1 + \theta_2 X_2 + \cdots + \theta_n X_n}) = \mathrm{E}(e^{\theta_1 X_1})\mathrm{E}(e^{\theta_2 X_2}) \cdots \mathrm{E}(e^{\theta_n X_n}) \\ &= \phi_1(\theta_1)\phi_2(\theta_2) \cdots \phi_n(\theta_n).\end{aligned}$$

2. **Theorem:** Suppose that the multivariate random variables X_1, X_2, $\cdots$, X_n are mutually independently and identically distributed. X_i has a normal distribution with mean μ and variance σ^2, i.e., $X_i \sim N(\mu, \sigma^2)$. Let us define $\hat{\mu} = \sum_{i=1}^n a_i X_i$, where a_i, $i = 1, 2, \cdots, n$, are assumed to be known. Then, $\hat{\mu}$ has a normal distribution with mean $\mu \sum_{i=1}^n a_i$ and variance $\sigma^2 \sum_{i=1}^n a_i^2$, i.e., $\hat{\mu} \sim N(\mu \sum_{i=1}^n a_i, \sigma^2 \sum_{i=1}^n a_i^2)$.

   Proof:

   From Example 1.8 (p.15) and Example 1.9 (p.23), it is shown that the moment-generating function of X is given by: $\phi_x(\theta) = \exp(\mu\theta + \frac{1}{2}\sigma^2\theta^2)$, when X is normally distributed as $X \sim N(\mu, \sigma^2)$.

   Let $\phi_{\hat{\mu}}$ be the moment-generating function of $\hat{\mu}$.

$$\begin{aligned}\phi_{\hat{\mu}}(\theta) = \mathrm{E}(e^{\theta\hat{\mu}}) = \mathrm{E}(e^{\theta \sum_{i=1}^n a_i X_i}) = \prod_{i=1}^n \mathrm{E}(e^{\theta a_i X_i}) = \prod_{i=1}^n \phi_x(a_i\theta) \\ = \prod_{i=1}^n \exp(\mu a_i\theta + \frac{1}{2}\sigma^2 a_i^2\theta^2) = \exp(\mu \sum_{i=1}^n a_i\theta + \frac{1}{2}\sigma^2 \sum_{i=1}^n a_i^2\theta^2)\end{aligned}$$

   which is equivalent to the moment-generating function of the normal distribution with mean $\mu \sum_{i=1}^n a_i$ and variance $\sigma^2 \sum_{i=1}^n a_i^2$, where μ and σ^2 in $\phi_x(\theta)$ is simply replaced by $\mu \sum_{i=1}^n a_i$ and $\sigma^2 \sum_{i=1}^n a_i^2$ in $\phi_{\hat{\mu}}(\theta)$, respectively.

   Moreover, note as follows. When $a_i = 1/n$ is taken for all $i = 1, 2, \cdots, n$, i.e., when $\hat{\mu} = \overline{X}$ is taken, $\hat{\mu} = \overline{X}$ is normally distributed as: $\overline{X} \sim N(\mu, \sigma^2/n)$. The readers should check difference between Theorem 11 on p.21 and this theorem.

1.6 Law of Large Numbers and Central Limit Theorem

1.6.1 Chebyshev's Inequality

In this section, we introduce Chebyshev's inequality, which enables us to find upper and lower bounds given a certain probability.

Theorem: Let $g(X)$ be a nonnegative function of the random variable X, i.e., $g(X) \geq 0$. If $\mathrm{E}(g(X))$ exists, then we have:

$$P(g(X) \geq k) \leq \frac{\mathrm{E}(g(X))}{k}, \tag{1.6}$$

for a positive constant value k.

Proof:
We define the discrete random variable U as follows:

$$U = \begin{cases} 1, & \text{if } g(X) \geq k, \\ 0, & \text{if } g(X) < k. \end{cases}$$

Thus, the discrete random variable U takes 0 or 1. Suppose that the probability function of U is given by:

$$f(u) = P(U = u),$$

where $P(U = u)$ is represented as:

$$P(U = 1) = P(g(X) \geq k),$$
$$P(U = 0) = P(g(X) < k).$$

Then, in spite of the value which U takes, the following equation always holds:

$$g(X) \geq kU,$$

which implies that we have $g(X) \geq k$ when $U = 1$ and $g(X) \geq 0$ when $U = 0$, where k is a positive constant value. Therefore, taking the expectation on both sides, we obtain:

$$\mathrm{E}(g(X)) \geq k\mathrm{E}(U), \tag{1.7}$$

where $\mathrm{E}(U)$ is given by:

$$\begin{aligned} \mathrm{E}(U) &= \sum_{u=0}^{1} uP(U = u) = 1 \times P(U = 1) + 0 \times P(U = 0) = P(U = 1) \\ &= P(g(X) \geq k). \end{aligned} \tag{1.8}$$

Accordingly, substituting equation (1.8) into equation (1.7), we have the following inequality:

$$P(g(X) \geq k) \leq \frac{\mathrm{E}(g(X))}{k}.$$

Chebyshev's Inequality: Assume that $\mathrm{E}(X) = \mu$, $\mathrm{V}(X) = \sigma^2$, and λ is a positive constant value. Then, we have the following inequality:

$$P(|X - \mu| \geq \lambda\sigma) \leq \frac{1}{\lambda^2},$$

or equivalently,

$$P(|X - \mu| < \lambda\sigma) \geq 1 - \frac{1}{\lambda^2},$$

which is called **Chebyshev's inequality**.

Proof:
Take $g(X) = (X - \mu)^2$ and $k = \lambda^2\sigma^2$. Then, we have:

$$P((X - \mu)^2 \geq \lambda^2\sigma^2) \leq \frac{\mathrm{E}(X - \mu)^2}{\lambda^2\sigma^2},$$

which implies

$$P(|X - \mu| \geq \lambda\sigma) \leq \frac{1}{\lambda^2}.$$

Note that $\mathrm{E}(X - \mu)^2 = \mathrm{V}(X) = \sigma^2$.

Since we have $P(|X - \mu| \geq \lambda\sigma) + P(|X - \mu| < \lambda\sigma) = 1$, we can derive the following inequality:

$$P(|X - \mu| < \lambda\sigma) \geq 1 - \frac{1}{\lambda^2}. \tag{1.9}$$

An Interpretation of Chebyshev's inequality: $1/\lambda^2$ is an upper bound for the probability $P(|X - \mu| \geq \lambda\sigma)$. Equation (1.9) is rewritten as:

$$P(\mu - \lambda\sigma < X < \mu + \lambda\sigma) \geq 1 - \frac{1}{\lambda^2}.$$

That is, the probability that X falls within $\lambda\sigma$ units of μ is greater than or equal to $1 - 1/\lambda^2$. Taking an example of $\lambda = 2$, the probability that X falls within two standard deviations of its mean is at least 0.75.

Furthermore, note as follows. Taking $\epsilon = \lambda\sigma$, we obtain as follows:

$$P(|X - \mu| \geq \epsilon) \leq \frac{\sigma^2}{\epsilon^2},$$

i.e.,

$$P(|X - \mathrm{E}(X)| \geq \epsilon) \leq \frac{\mathrm{V}(X)}{\epsilon^2}, \tag{1.10}$$

which inequality is used in the next section.

Remark: Equation (1.10) can be derived when we take $g(X) = (X - \mu)^2$, $\mu = \mathrm{E}(X)$ and $k = \epsilon^2$ in equation (1.6). Even when we have $\mu \neq \mathrm{E}(X)$, the following inequality still hold:

$$P(|X - \mu| \geq \epsilon) \leq \frac{\mathrm{E}((X - \mu)^2)}{\epsilon^2}.$$

Note that $\mathrm{E}((X - \mu)^2)$ represents the mean square error (MSE). When $\mu = \mathrm{E}(X)$, the mean square error reduces to the variance.

1.6.2 Law of Large Numbers (Convergence in probability)

Law of Large Numbers: Assume that $X_1, X_2, \cdots, X_n$ are mutually independently and identically distributed with mean $\mathrm{E}(X_i) = \mu$ and variance $\mathrm{V}(X_i) = \sigma^2 < \infty$ for all i. Then, for any positive value ϵ, as $n \longrightarrow \infty$, we have the following result:

$$P(|\overline{X}_n - \mu| > \epsilon) \longrightarrow 0,$$

where $\overline{X}_n = (1/n) \sum_{i=1}^{n} X_i$. We say that $\overline{X}_n$ converges to μ in probability.

Proof:
Using (1.10), Chebyshev's inequality is represented as follows:

$$P(|\overline{X}_n - \mathrm{E}(\overline{X}_n)| > \epsilon) \leq \frac{\mathrm{V}(\overline{X}_n)}{\epsilon^2},$$

where X in (1.10) is replaced by $\overline{X}_n$. As in Section 1.3.2 (p.21), we have $\mathrm{E}(\overline{X}_n) = \mu$ and $\mathrm{V}(\overline{X}_n) = \sigma^2/n$, which are substituted into the above inequality. Then, we obtain:

$$P(|\overline{X}_n - \mu| > \epsilon) \leq \frac{\sigma^2}{n\epsilon^2}.$$

Accordingly, when $n \longrightarrow \infty$, the following equation holds:

$$P(|\overline{X}_n - \mu| > \epsilon) \leq \frac{\sigma^2}{n\epsilon^2} \longrightarrow 0.$$

That is, $\overline{X}_n \longrightarrow \mu$ is obtained as $n \longrightarrow \infty$, which is written as: $\mathrm{plim}\, \overline{X}_n = \mu$. This theorem is called the **law of large numbers**.

The condition $P(|\overline{X}_n - \mu| > \epsilon) \longrightarrow 0$ or equivalently $P(|\overline{X}_n - \mu| < \epsilon) \longrightarrow 1$ is used as the definition of **convergence in probability**. In this case, we say that $\overline{X}_n$ converges to μ in probability.

Theorem: In the case where $X_1, X_2, \cdots, X_n$ are not identically distributed and they are not mutually independently distributed, we assume that

$$m_n = \mathrm{E}(\sum_{i=1}^{n} X_i) < \infty,$$
$$V_n = \mathrm{V}(\sum_{i=1}^{n} X_i) < \infty,$$
$$\frac{V_n}{n^2} \longrightarrow 0, \quad \text{as } n \longrightarrow \infty.$$

Then, we obtain the following result:

$$\frac{\sum_{i=1}^{n} X_i - m_n}{n} \longrightarrow 0.$$

That is, $\overline{X}_n$ converges to $\lim_{n\to\infty} \frac{m_n}{n}$ in probability. This theorem is also called the law of large numbers.

1.6.3 Central Limit Theorem

Central Limit Theorem: $X_1, X_2, \cdots, X_n$ are mutually independently and identically distributed with $\mathrm{E}(X_i) = \mu$ and $\mathrm{V}(X_i) = \sigma^2$ for all i. Both μ and σ^2 are finite. Under the above assumptions, when $n \longrightarrow \infty$, we have:

$$P\Big(\frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}} < x\Big) \longrightarrow \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2}\, \mathrm{d}u,$$

which is called the **central limit theorem**.

Proof:

Define $Y_i = \dfrac{X_i - \mu}{\sigma}$. We can rewrite as follows:

$$\frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}} = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \frac{X_i - \mu}{\sigma} = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} Y_i.$$

Since $Y_1, Y_2, \cdots, Y_n$ are mutually independently and identically distributed, the moment-generating function of Y_i is identical for all i, which is denoted by $\phi(\theta)$. Using $\mathrm{E}(Y_i) = 0$ and $\mathrm{V}(Y_i) = 1$, the moment-generating function of Y_i, $\phi(\theta)$, is rewritten as:

$$\begin{aligned}\phi(\theta) = \mathrm{E}(e^{Y_i\theta}) &= \mathrm{E}\Big(1 + Y_i\theta + \frac{1}{2}Y_i^2\theta^2 + \frac{1}{3!}Y_i^3\theta^3 \cdots\Big) \\ &= 1 + \frac{1}{2}\theta^2 + O(\theta^3).\end{aligned}$$

In the second equality, $e^{Y_i\theta}$ is approximated by the Taylor series expansion around $\theta = 0$. See Appendix 1.3 for the Taylor series expansion. $O(x)$ implies that it is a polynomial function of x and the higher-order terms but it is dominated by x. In this case, $O(\theta^3)$ is a function of $\theta^3, \theta^4, \cdots$. Since the moment-generating function is conventionally evaluated at $\theta = 0$, θ^3 is the largest value of $\theta^3, \theta^4, \cdots$ and accordingly $O(\theta^3)$ is dominated by θ^3 (in other words, $\theta^4, \theta^5, \cdots$ are small enough, compared with θ^3).

Define Z as:

$$Z = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} Y_i.$$

Then, the moment-generating function of Z, i.e., $\phi_z(\theta)$, is given by:

$$\phi_z(\theta) = \mathrm{E}(e^{Z\theta}) = \mathrm{E}\Big(e^{\frac{\theta}{\sqrt{n}}\sum_{i=1}^{n} Y_i}\Big) = \prod_{i=1}^{n} \mathrm{E}\Big(e^{\frac{\theta}{\sqrt{n}}Y_i}\Big) = \Big(\phi(\frac{\theta}{\sqrt{n}})\Big)^n$$
$$= \Big(1 + \frac{1}{2}\frac{\theta^2}{n} + O(\frac{\theta^3}{n^{\frac{3}{2}}})\Big)^n = \Big(1 + \frac{1}{2}\frac{\theta^2}{n} + O(n^{-\frac{3}{2}})\Big)^n.$$

We consider that n goes to infinity. Therefore, $O(\frac{\theta^3}{n^{\frac{3}{2}}})$ indicates a function of $n^{-\frac{3}{2}}$.

Moreover, consider $x = \frac{1}{2}\frac{\theta^2}{n} + O(n^{-\frac{3}{2}})$. Multiply n/x on both sides of $x = \frac{1}{2}\frac{\theta^2}{n} + O(n^{-\frac{3}{2}})$. Then, we obtain $n = \frac{1}{x}\Big(\frac{1}{2}\theta^2 + O(n^{-\frac{1}{2}})\Big)$. Substitute $n = \frac{1}{x}\Big(\frac{1}{2}\theta^2 + O(n^{-\frac{1}{2}})\Big)$ into the moment-generating function of Z, i.e., $\phi_z(\theta)$. Then, we obtain:

$$\phi_z(\theta) = \Big(1 + \frac{1}{2}\frac{\theta^2}{n} + O(n^{-\frac{3}{2}})\Big)^n = (1+x)^{\frac{1}{x}(\frac{\theta^2}{2} + O(n^{-\frac{1}{2}}))}$$
$$= \Big((1+x)^{\frac{1}{x}}\Big)^{\frac{\theta^2}{2} + O(n^{-\frac{1}{2}})} \longrightarrow e^{\frac{\theta^2}{2}}.$$

Note that $x \longrightarrow 0$ when $n \longrightarrow \infty$ and that $\lim_{x\to 0}(1+x)^{1/x} = e$ as in Section 1.2.3 (p.12). Furthermore, we have $O(n^{-\frac{1}{2}}) \longrightarrow 0$ as $n \longrightarrow \infty$.

Since $\phi_z(\theta) = e^{\frac{\theta^2}{2}}$ is the moment-generating function of the standard normal distribution (see p.14 in Section 1.3.1 for the moment-generating function of the standard normal probability density), we have:

$$P\Big(\frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}} < x\Big) \longrightarrow \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2}\, \mathrm{d}u,$$

or equivalently,

$$\frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}} \longrightarrow N(0,1).$$

The following expression is also possible:

$$\sqrt{n}(\overline{X}_n - \mu) \longrightarrow N(0, \sigma^2). \tag{1.11}$$

Corollary 1: When $\mathrm{E}(X_i) = \mu$, $\mathrm{V}(X_i) = \sigma^2$ and $\overline{X}_n = (1/n)\sum_{i=1}^{n} X_i$, note that

$$\frac{\overline{X}_n - \mathrm{E}(\overline{X}_n)}{\sqrt{\mathrm{V}(\overline{X}_n)}} = \frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}}.$$

Therefore, we can rewrite the above theorem as:

$$P\Big(\frac{\overline{X}_n - \mathrm{E}(\overline{X}_n)}{\sqrt{\mathrm{V}(\overline{X}_n)}} < x\Big) \longrightarrow \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2}\, \mathrm{d}u.$$

Corollary 2: Consider the case where $X_1, X_2, \cdots, X_n$ are not identically distributed and they are not mutually independently distributed. Assume that

$$\lim_{n\to\infty} n\mathrm{V}(\overline{X_n}) = \sigma^2 < \infty,$$

where $\overline{X_n} = (1/n)\sum_{i=1}^{n} X_i$. Then, when $n \longrightarrow \infty$, we have:

$$P\Big(\frac{\overline{X_n} - \mathrm{E}(\overline{X_n})}{\sqrt{\mathrm{V}(\overline{X_n})}} < x\Big) = P\Big(\frac{\sum_{i=1}^{n} X_i - \mathrm{E}(\sum_{i=1}^{n} X_i)}{\sqrt{\mathrm{V}(\sum_{i=1}^{n} X_i)}} < x\Big) \longrightarrow \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2}\, \mathrm{d}u.$$

1.7 Statistical Inference

1.7.1 Point Estimation

Suppose that the functional form of the underlying distribution on population is known but the parameter θ included in the distribution is not known. The distribution function of population is given by $f(x;\theta)$. Let $x_1, x_2, \cdots, x_n$ be the n observed data drawn from the population distribution. Consider estimating the parameter θ using the n observed data. Let $\hat{\theta}_n(x_1, x_2, \cdots, x_n)$ be a function of the observed data $x_1, x_2, \cdots, x_n$. Suppose that $\hat{\theta}_n(x_1, x_2, \cdots, x_n)$ is constructed from the purpose of estimating the parameter θ. $\hat{\theta}_n(x_1, x_2, \cdots, x_n)$ takes a certain value given the n observed data. Then, $\hat{\theta}_n(x_1, x_2, \cdots, x_n)$ is called the **point estimate** of θ, or simply the **estimate** of θ.

Example 1.11: Consider the case of $\theta = (\mu, \sigma^2)$, where the unknown parameters contained in population is given by mean and variance. A point estimate of population mean μ is given by:

$$\hat{\mu}_n(x_1, x_2, \cdots, x_n) \equiv \overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i.$$

A point estimate of population variance σ^2 is:

$$\hat{\sigma}_n^2(x_1, x_2, \cdots, x_n) \equiv s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2.$$

An alternative point estimate of population variance σ^2 is:

$$\widetilde{\sigma}_n^2(x_1, x_2, \cdots, x_n) \equiv s^{**2} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2.$$

1.7.2 Statistic, Estimate and Estimator

The underlying distribution of population is assumed to be known, but the parameter θ, which characterizes the underlying distribution, is unknown. The probability density function of population is given by $f(x;\theta)$. Let $X_1, X_2, \cdots, X_n$ be a subset of population, which are regarded as the random variables and are assumed to be mutually independent. $x_1, x_2, \cdots, x_n$ are taken as the experimental values of the random variables X_1, $X_2, \cdots, X_n$. In statistics, we consider that n-variate random variables $X_1, X_2, \cdots, X_n$ takes the experimental values $x_1, x_2, \cdots, x_n$ by chance. There, the experimental values and the actually observed data series are used in the same meaning.

As discussed in Section 1.7.1, $\hat{\theta}_n(x_1, x_2, \cdots, x_n)$ denotes the point estimate of θ. In the case where the observed data $x_1, x_2, \cdots, x_n$ are replaced by the corresponding random variables $X_1, X_2, \cdots, X_n$, a function of $X_1, X_2, \cdots, X_n$, i.e., $\hat{\theta}(X_1, X_2, \cdots, X_n)$, is called the **estimator** of θ, which should be distinguished from the **estimate** of θ, i.e., $\hat{\theta}(x_1, x_2, \cdots, x_n)$.

Example 1.12: Let $X_1, X_2, \cdots, X_n$ denote a random sample of n from a given distribution $f(x;\theta)$. Consider the case of $\theta = (\mu, \sigma^2)$.

The estimator of μ is given by $\overline{X} = (1/n)\sum_{i=1}^{n} X_i$, while the estimate of μ is $\overline{x} = (1/n)\sum_{i=1}^{n} x_i$. The estimator of σ^2 is $S^2 = \sum_{i=1}^{n}(X_i - \overline{X})^2/(n-1)$ and the estimate of σ^2 is $s^2 = \sum_{i=1}^{n}(x_i - \overline{x})^2/(n-1)$.

There are numerous estimators and estimates of θ. All of $(1/n)\sum_{i=1}^{n} X_i$, $(X_1+X_n)/2$, median of $(X_1, X_2, \cdots, X_n)$ and so on are taken as the estimators of μ. Of course, they are called the estimates of θ when X_i is replaced by x_i for all i. Similarly, both $S^2 = \sum_{i=1}^{n}(X_i - \overline{X})^2/(n-1)$ and $S^{*2} = \sum_{i=1}^{2}(X_i - \overline{X})^2/n$ are the estimators of σ^2. We need to choose one out of the numerous estimators of θ. The problem of choosing an optimal estimator out of the numerous estimators is discussed in Sections 1.7.4 and 1.7.5.

In addition, note as follows. A function of random variables is called a **statistic**. The statistic for estimation of the parameter is called an estimator. Therefore, an estimator is a family of a statistic.

1.7.3 Estimation of Mean and Variance

Suppose that the population distribution is given by $f(x;\theta)$. The random sample X_1, $X_2, \cdots, X_n$ are assumed to be drawn from the population distribution $f(x;\theta)$, where $\theta = (\mu, \sigma^2)$. Therefore, we can assume that $X_1, X_2, \cdots, X_n$ are mutually independently

and identically distributed, where "identically" implies $\mathrm{E}(X_i) = \mu$ and $\mathrm{V}(X_i) = \sigma^2$ for all i.

Consider the estimators of $\theta = (\mu, \sigma^2)$ as follows.

1. The estimator of population mean μ is:
   - $\overline{X} = \dfrac{1}{n}\sum_{i=1}^{n} X_i$.
2. The estimators of population variance σ^2 are:
   - $S^{*2} = \dfrac{1}{n}\sum_{i=1}^{n}(X_i - \mu)^2$, when μ is known,
   - $S^2 = \dfrac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2$,
   - $S^{**2} = \dfrac{1}{n}\sum_{i=1}^{n}(X_i - \overline{X})^2$,

Properties of $\overline{X}$: From Theorem on p.21, mean and variance of $\overline{X}$ are obtained as follows:

$$\mathrm{E}(\overline{X}) = \mu, \quad \mathrm{V}(\overline{X}) = \frac{\sigma^2}{n}.$$

Properties of S^{*2}, S^2 and S^{**2}: The expectation of S^{*2} is:

$$\begin{aligned}\mathrm{E}(S^{*2}) &= \mathrm{E}\Big(\frac{1}{n}\sum_{i=1}^{n}(X_i - \mu)^2\Big) = \frac{1}{n}\mathrm{E}\Big(\sum_{i=1}^{n}(X_i - \mu)^2\Big) = \frac{1}{n}\sum_{i=1}^{n}\mathrm{E}\Big((X_i - \mu)^2\Big) \\ &= \frac{1}{n}\sum_{i=1}^{n}\mathrm{V}(X_i) = \frac{1}{n}\sum_{i=1}^{n}\sigma^2 = \frac{1}{n}n\sigma^2 = \sigma^2,\end{aligned}$$

where $\mathrm{E}((X_i - \mu)^2) = \mathrm{V}(X_i) = \sigma^2$ is used in the fourth and fifth equalities.

Next, the expectation of S^2 is given by:

$$\begin{aligned}\mathrm{E}(S^2) &= \mathrm{E}\Big(\frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2\Big) = \frac{1}{n-1}\mathrm{E}\Big(\sum_{i=1}^{n}(X_i - \overline{X})^2\Big) \\ &= \frac{1}{n-1}\mathrm{E}\Big(\sum_{i=1}^{n}((X_i - \mu) - (\overline{X} - \mu))^2\Big) \\ &= \frac{1}{n-1}\mathrm{E}\Big(\sum_{i=1}^{n}((X_i - \mu)^2 - 2(X_i - \mu)(\overline{X} - \mu) + (\overline{X} - \mu)^2)\Big) \\ &= \frac{1}{n-1}\mathrm{E}\Big(\sum_{i=1}^{n}(X_i - \mu)^2 - 2(\overline{X} - \mu)\sum_{i=1}^{n}(X_i - \mu) + n(\overline{X} - \mu)^2\Big)\end{aligned}$$

$$= \frac{1}{n-1}\mathrm{E}\Big(\sum_{i=1}^{n}(X_i-\mu)^2 - n(\overline{X}-\mu)^2\Big)$$
$$= \frac{n}{n-1}\mathrm{E}\Big(\frac{1}{n}\sum_{i=1}^{n}(X_i-\mu)^2\Big) - \frac{n}{n-1}\mathrm{E}((\overline{X}-\mu)^2)$$
$$= \frac{n}{n-1}\sigma^2 - \frac{n}{n-1}\frac{\sigma^2}{n} = \sigma^2.$$

$\sum_{i=1}^{n}(X_i-\mu) = n(\overline{X}-\mu)$ is used in the sixth equality. $\mathrm{E}\Big((1/n)\sum_{i=1}^{n}(X_i-\mu)^2\Big) = \mathrm{E}(S^{*2}) = \sigma^2$ and $\mathrm{E}((\overline{X}-\mu)^2) = \mathrm{V}(\overline{X}) = \sigma^2/n$ are required in the eighth equality.

Finally, the mathematical expectation of S^{**2} is represented by:

$$\mathrm{E}(S^{**2}) = \mathrm{E}\Big(\frac{1}{n}\sum_{i=1}^{n}(X_i-\overline{X})^2\Big) = \mathrm{E}\Big(\frac{n-1}{n}\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\overline{X})^2\Big)$$
$$= \mathrm{E}\Big(\frac{n-1}{n}S^2\Big) = \frac{n-1}{n}\mathrm{E}(S^2) = \frac{n-1}{n}\sigma^2 \neq \sigma^2.$$

Summarizing the above results, we obtain as follows:

$$\mathrm{E}(S^{*2}) = \sigma^2, \quad \mathrm{E}(S^2) = \sigma^2, \quad \mathrm{E}(S^{**2}) = \frac{n-1}{n}\sigma^2 \neq \sigma^2.$$

1.7.4 Point Estimation: Optimality

As mentioned in the previous sections, θ denotes the parameter to be estimated. $\hat{\theta}_n(X_1, X_2, \cdots, X_n)$ represents the estimator of θ, while $\hat{\theta}_n(x_1, x_2, \cdots, x_n)$ indicates the estimate of θ. Hereafter, in the case of no confusion, $\hat{\theta}_n(X_1, X_2, \cdots, X_n)$ is simply written as $\hat{\theta}_n$.

As discussed above, there are numerous candidates of the estimator $\hat{\theta}_n$. The desired properties which $\hat{\theta}_n$ have to satisfy include unbiasedness, efficiency and consistency.

Unbiasedness: One of the desirable features that the estimator of the parameter should have is given by:

$$\mathrm{E}(\hat{\theta}_n) = \theta, \tag{1.12}$$

which implies that $\hat{\theta}_n$ is distributed around θ. When the condition (1.12) holds, $\hat{\theta}_n$ is called the **unbiased estimator** of θ. $\mathrm{E}(\hat{\theta}_n) - \theta$ is defined as **bias**.

As an example of unbiasedness, consider the case of $\theta = (\mu, \sigma^2)$. Suppose that $X_1, X_2, \cdots, X_n$ are mutually independently and identically distributed with mean μ and variance σ^2. Consider the following estimators of μ and σ^2.

1. The estimator of μ is:

   - $\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$.

2. The estimators of σ^2 are:
   - $S^2 = \dfrac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2$,
   - $S^{**2} = \dfrac{1}{n}\sum_{i=1}^{n}(X_i - \overline{X})^2$.

Since we have obtained $\mathrm{E}(\overline{X}) = \mu$ and $\mathrm{E}(S^2) = \sigma^2$ in Section 1.7.3, $\overline{X}$ and S^2 are unbiased estimators of μ and σ^2. However, we have obtained the result $\mathrm{E}(S^{**2}) \neq \sigma^2$ in Section 1.7.3 and therefore S^{**2} is not an unbiased estimator of σ^2. Thus, according to the criterion of unbiasedness, S^2 is preferred to S^{**2} for estimation of σ^2.

Efficiency: Consider two estimators, i.e., $\hat{\theta}_n$ and $\widetilde{\theta}_n$. Both are assumed to be unbiased. That is, we have the following condition: $\mathrm{E}(\hat{\theta}_n) = \theta$ and $\mathrm{E}(\widetilde{\theta}_n) = \theta$. When $\mathrm{V}(\hat{\theta}_n) < \mathrm{V}(\widetilde{\theta}_n)$, we say that $\hat{\theta}_n$ is more efficient than $\widetilde{\theta}_n$. The estimator which is widely distributed is not preferred.

Consider as many unbiased estimators as possible. The unbiased estimator with the least variance is known as the **efficient estimator**. We have the case where an efficient estimator does not exist.

In order to obtain the efficient estimator, we utilize Cramer-Rao inequality. Suppose that X_i has the probability density function $f(x_i;\theta)$ for all i, i.e., $X_1, X_2, \cdots, X_n$ are mutually independently and identically distributed. For any unbiased estimator of θ, denoted by $\hat{\theta}_n$, it is known that we have the following inequality:

$$\mathrm{V}(\hat{\theta}_n) \geq \frac{\sigma^2(\theta)}{n}, \tag{1.13}$$

where

$$\sigma^2(\theta) = \frac{1}{\mathrm{E}\left(\left(\dfrac{\partial \log f(X;\theta)}{\partial\theta}\right)^2\right)} = \frac{1}{\mathrm{V}\left(\left(\dfrac{\partial \log f(X;\theta)}{\partial\theta}\right)\right)} = -\frac{1}{\mathrm{E}\left(\dfrac{\partial^2 \log f(X;\theta)}{\partial\theta^2}\right)}, \tag{1.14}$$

which is known as the **Cramer-Rao inequality**. See Appendix 1.4 for proof of the Cramer-Rao inequality.

When there exists the unbiased estimator $\hat{\theta}_n$ such that the equality in (1.13) holds, $\hat{\theta}_n$ becomes the unbiased estimator with minimum variance, which is the efficient estimator. $\sigma^2(\theta)/n$ is called the **Cramer-Rao lower bound**.

Example 1.13 (Efficient Estimator): Suppose that $X_1, X_2, \cdots, X_n$ are mutually independently, identically and normally distributed with mean μ and variance σ^2. Then, we show that $\overline{X}$ is an efficient estimator of μ.

When $\sigma^2 < \infty$, from Theorem on p.21, $\mathrm{V}(\overline{X})$ is given by σ^2/n in spite of the distribution of X_i, $i = 1, 2, \cdots, n$. .....(A)

On the other hand, because we assume that X_i is normally distributed with mean μ and variance σ^2, the probability density function of X_i is given by:

$$f(x;\mu) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{1}{2\sigma^2}(x-\mu)^2\Big).$$

The Cramer-Rao inequality is represented as:

$$\mathrm{V}(\overline{X}) \geq \frac{1}{n\mathrm{E}\left(\left(\frac{\partial \log f(X;\mu)}{\partial \mu}\right)^2\right)},$$

where the logarithm of $f(X;\mu)$ is written as:

$$\log f(X;\mu) = -\frac{1}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}(X-\mu)^2.$$

Therefore, the partial derivative of $f(X;\mu)$ with respect to μ is:

$$\frac{\partial \log f(X;\mu)}{\partial \mu} = \frac{1}{\sigma^2}(X-\mu).$$

Accordingly, the Cramer-Rao inequality in this case is written as:

$$\mathrm{V}(\overline{X}) \geq \frac{1}{n\mathrm{E}\left(\left(\frac{1}{\sigma^2}(X-\mu)\right)^2\right)} = \frac{1}{n\frac{1}{\sigma^4}\mathrm{E}((X-\mu)^2)} = \frac{\sigma^2}{n}. \quad \ldots\ldots\ldots\ldots\ldots\ldots (\mathrm{B})$$

From (A) and (B), variance of $\overline{X}$ is equal to the lower bound of Cramer-Rao inequality, i.e., $\mathrm{V}(\overline{X}) = \frac{\sigma^2}{n}$, which implies that the equality included in the Cramer-Rao inequality holds. Therefore, we can conclude that the sample mean $\overline{X}$ is an efficient estimator of μ.

Example 1.14 (Linear Unbiased Minimum Variance Estimator): Suppose that $X_1, X_2, \cdots, X_n$ are mutually independently and identically distributed with mean μ and variance σ^2 (note that the normality assumption is excluded from Example 1.13). Consider the following linear estimator: $\hat{\mu} = \sum_{i=1}^n a_i X_i$. Then, we want to show $\hat{\mu}$ (i.e., $\overline{X}$) is a **linear unbiased minimum variance estimator** if $a_i = 1/n$ for all i, i.e., if $\hat{\mu} = \overline{X}$.

Utilizing Theorem on p.20, when $\mathrm{E}(X_i) = \mu$ and $\mathrm{V}(X_i) = \sigma^2$ for all i, we have: $\mathrm{E}(\hat{\mu}) = \mu \sum_{i=1}^n a_i$ and $\mathrm{V}(\hat{\mu}) = \sigma^2 \sum_{i=1}^n a_i^2$.

Since $\hat{\mu}$ is linear in X_i, $\hat{\mu}$ is called a **linear estimator** of μ. In order for $\hat{\mu}$ to be unbiased, we need to have the condition: $\mathrm{E}(\hat{\mu}) = \mu \sum_{i=1}^n a_i = \mu$. That is, if $\sum_{i=1}^n a_i = 1$ is satisfied, $\hat{\mu}$ gives us a **linear unbiased estimator**. Thus, as mentioned in Example 1.12 of Section 1.7.2, there are numerous unbiased estimators.

The variance of $\hat{\mu}$ is given by $\sigma^2 \sum_{i=1}^n a_i^2$. We obtain the value of a_i which minimizes $\sum_{i=1}^n a_i^2$ with the constraint $\sum_{i=1}^n a_i = 1$. Construct the Lagrange function as follows:

$$L = \frac{1}{2} \sum_{i=1}^{n} a_i^2 + \lambda(1 - \sum_{i=1}^{n} a_i),$$

where λ denotes the Lagrange multiplier. The $\frac{1}{2}$ in front of the first term appears to make life easier later on and does not affect the outcome. To determine the optimum values, we set the partial derivatives of L with respect to a_i and λ equal to zero, i.e.,

$$\frac{\partial L}{\partial a_i} = a_i - \lambda = 0, \qquad i = 1, 2, \cdots, n,$$
$$\frac{\partial L}{\partial \lambda} = 1 - \sum_{i=1}^{n} a_i = 0.$$

Solving the above equations, $a_i = \lambda = 1/n$ is obtained. Therefore, when $a_i = 1/n$ for all i, $\hat{\mu}$ has minimum variance in a class of linear unbiased estimators. That is, $\overline{X}$ is a **linear unbiased minimum variance estimator**.

The linear unbiased minimum variance estimator should be distinguished from the efficient estimator discussed in Example 1.13. The former does not requires the assumption on the underlying distribution. The latter gives us the unbiased estimator which variance is equal to the Cramer-Rao lower bound, which is not restricted to a class of the linear unbiased estimators. Under the assumption of normal population, the linear unbiased minimum variance estimator leads to the efficient estimator. However, both are different in general. In addition, note that the efficient estimator does not necessarily exist.

Consistency: Let $\hat{\theta}_n$ be an estimator of θ. Suppose that for any $\epsilon > 0$ we have the following:

$$P(|\hat{\theta}_n - \theta| > \epsilon) \longrightarrow 0, \qquad \text{as} \quad n \longrightarrow \infty,$$

which implies that $\hat{\theta} \longrightarrow \theta$ as $n \longrightarrow \infty$. Then, we say that $\hat{\theta}_n$ is a **consistent estimator** of θ. That is, the estimator which approaches the true parameter value as the sample size is large is called the consistent estimator of the parameter.

Example 1.15: Suppose that $X_1, X_2, \cdots, X_n$ are mutually independently and identically distributed with mean μ and variance σ^2. Assume that σ^2 is known. Then, it is shown that $\overline{X}$ is a consistent estimator of μ.

From (1.10), Chebyshev's inequality is given by:

$$P(|X - \mathrm{E}(X)| > \epsilon) \leq \frac{\mathrm{V}(X)}{\epsilon^2},$$

for a random variable X. Here, replacing X by $\overline{X}$, we obtain $\mathrm{E}(\overline{X})$ and $\mathrm{V}(\overline{X})$ as follows:

$$\mathrm{E}(\overline{X}) = \mu, \quad \mathrm{V}(\overline{X}) = \frac{\sigma^2}{n},$$

because $\mathrm{E}(X_i) = \mu$ and $\mathrm{V}(X_i) = \sigma^2 < \infty$ are assumed for all i.

Then, when $n \longrightarrow \infty$, we obtain the following result:

$$P(|\overline{X} - \mu| > \epsilon) \leq \frac{\sigma^2}{n\epsilon^2} \longrightarrow 0,$$

which implies that $\overline{X} \longrightarrow \mu$ as $n \longrightarrow \infty$. Therefore, we can conclude that $\overline{X}$ is a consistent estimator of μ.

Summarizing the results up to now, $\overline{X}$ is an unbiased, minimum variance and consistent estimator of population mean μ. When the distribution of X_i is assumed to be normal for all i, $\overline{X}$ leads to an unbiased, efficient and consistent estimator of μ.

Example 1.16: Suppose that $X_1, X_2, \cdots, X_n$ are mutually independently, identically and normally distributed with mean μ and variance σ^2. Consider $S^{**2} = \dfrac{1}{n}\displaystyle\sum_{i=1}^{n}(X_i - \overline{X})^2$, which is an estimate of σ^2.

In Remark on p.32, X and μ are replaced by S^{**2} and σ^2. Then, we obtain the following inequality:

$$P(|S^{**2} - \sigma^2| < \epsilon) \geq 1 - \frac{\mathrm{E}((S^{**2} - \sigma^2)^2)}{\epsilon^2}.$$

We compute $\mathrm{E}((S^{**2} - \sigma^2)^2)$. Since $(n-1)S^2/\sigma^2 \sim \chi^2(n-1)$, we obtain $\mathrm{E}((n-1)S^2/\sigma^2) = n-1$ and $\mathrm{V}((n-1)S^2/\sigma^2) = 2(n-1)$, where $S^2 = \dfrac{1}{n-1}\displaystyle\sum_{i=1}^{n}(X_i - \overline{X})^2$. See Section 2.2.8 (p.106) for the chi-square distribution $\chi^2(n-1)$. Therefore, $\mathrm{E}(S^2) = \sigma^2$ and $\mathrm{V}(S^2) = 2\sigma^4/(n-1)$ can be derived. Using $S^{**2} = S^2(n-1)/n$, we have the following:

$$\begin{aligned}\mathrm{E}((S^{**2} - \sigma^2)^2) &= \mathrm{E}\Big(\Big(\frac{n-1}{n}S^2 - \sigma^2\Big)^2\Big) = \mathrm{E}\Big(\Big(\frac{n-1}{n}(S^2 - \sigma^2) - \frac{\sigma^2}{n}\Big)^2\Big) \\ &= \frac{(n-1)^2}{n^2}\mathrm{E}((S^2 - \sigma^2)^2) + \frac{\sigma^4}{n^2} = \frac{(n-1)^2}{n^2}\mathrm{V}(S^2) + \frac{\sigma^4}{n^2} = \frac{(2n-1)}{n^2}\sigma^4.\end{aligned}$$

Therefore, as $n \longrightarrow \infty$, we obtain:

$$P(|S^{**2} - \sigma^2| < \epsilon) \geq 1 - \frac{1}{\epsilon^2}\frac{(2n-1)}{n^2}\sigma^4 \longrightarrow 1.$$

Because $S^{**2} \longrightarrow \sigma^2$, S^{**2} is a consistent estimator of σ^2. Thus, S^{**2} is not unbiased (see Section 1.7.3, p.38), but is is consistent.

1.7.5 Maximum Likelihood Estimator

In Section 1.7.4, the properties of the estimators $\overline{X}$ and S^2 are discussed. It is shown that $\overline{X}$ is an unbiased, efficient and consistent estimator of μ under normality assumption and that S^2 is an unbiased estimator of σ^2. Note that S^2 is not efficient but consistent (we do not check these features of S^2 in this book).

The population parameter θ depends on a functional form of the population distribution $f(x;\theta)$. It corresponds to (μ, σ^2) in the case of the normal distribution and β in the case of the exponential distribution (Section 2.2.4). Now, in more general cases, we want to consider how to estimate θ. The maximum likelihood estimator gives us one of the solutions.

Let $X_1, X_2, \cdots, X_n$ be mutually independently and identically distributed random samples. X_i has the probability density function $f(x;\theta)$. Under these assumptions, the joint density function of $X_1, X_2, \cdots, X_n$ is given by:

$$f(x_1, x_2, \cdots, x_n; \theta) = \prod_{i=1}^{n} f(x_i; \theta),$$

where θ denotes the unknown parameter.

Given the actually observed data $x_1, x_2, \cdots, x_n$, the joint density $f(x_1, x_2, \cdots, x_n; \theta)$ is regarded as a function of θ, i.e.,

$$l(\theta) = l(\theta; x) = l(\theta; x_1, x_2, \cdots, x_n) = \prod_{i=1}^{n} f(x_i; \theta).$$

$l(\theta)$ is called the **likelihood function**.

Let $\hat{\theta}_n$ be the θ which maximizes the likelihood function. Replacing $x_1, x_2, \cdots, x_n$ by $X_1, X_2, \cdots, X_n$, $\hat{\theta}_n = \hat{\theta}_n(X_1, X_2, \cdots, X_n)$ is called the **maximum likelihood estimator**, while $\hat{\theta}_n(x_1, x_2, \cdots, x_n)$ is called the **maximum likelihood estimate**.

That is, solving the following equation:

$$\frac{\partial l(\theta)}{\partial \theta} = 0,$$

the maximum likelihood estimator $\hat{\theta}_n \equiv \hat{\theta}_n(X_1, X_2, \cdots, X_n)$ is obtained.

Example 1.17: Suppose that $X_1, X_2, \cdots, X_n$ are mutually independently, identically and normally distributed with mean μ and variance σ^2. We derive the maximum likelihood estimators of μ and σ^2. The joint density (or the likelihood function) of $X_1, X_2, \cdots, X_n$ is written as:

$$\begin{aligned} f(x_1, x_2, \cdots, x_n; \mu, \sigma^2) &= \prod_{i=1}^{n} f(x_i; \mu, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{1}{2\sigma^2}(x_i - \mu)^2\Big) \\ &= (2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{1}{2\sigma^2} \sum_{i=1}^{n} (x_i - \mu)^2\Big) = l(\mu, \sigma^2). \end{aligned}$$

The logarithm of the likelihood function is given by:

$$\log l(\mu, \sigma^2) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \mu)^2,$$

which is called the **log-likelihood function**. For maximization of the likelihood function, differentiating the log-likelihood function $\log l(\mu, \sigma^2)$ with respect to μ and σ^2, the first derivatives should be equal to zero, i.e.,

$$\frac{\partial \log l(\mu, \sigma^2)}{\partial \mu} = \frac{1}{\sigma^2}\sum_{i=1}^{n}(x_i - \mu) = 0,$$
$$\frac{\partial \log l(\mu, \sigma^2)}{\partial \sigma^2} = -\frac{n}{2}\frac{1}{\sigma^2} + \frac{1}{2\sigma^4}\sum_{i=1}^{n}(x_i - \mu)^2 = 0.$$

Let $\hat{\mu}$ and $\hat{\sigma}^2$ be the solution which satisfies the above two equations. Solving the two equations, we obtain the maximum likelihood estimates as follows:

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i = \overline{x},$$
$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{\mu})^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2 = s^{**2}.$$

Replacing x_i by X_i for $i = 1, 2, \cdots, n$, the maximum likelihood estimators of μ and σ^2 are given by $\overline{X}$ and S^{**2}, respectively. Since $\mathrm{E}(\overline{X}) = \mu$, the maximum likelihood estimator of μ, $\overline{X}$, is an unbiased estimator. However, because of $\mathrm{E}(S^{**2}) = \dfrac{n-1}{n}\sigma^2 \neq \sigma^2$ as shown in Section 1.7.3, the maximum likelihood estimator of σ^2, S^{**2}, is not an unbiased estimator.

Properties of Maximum Likelihood Estimator: For small sample, the maximum likelihood estimator has the following properties.

- The maximum likelihood estimator is not necessarily unbiased in general, but we often have the case where we can construct the unbiased estimator by an appropriate transformation.

  For instance, in Example 1.17, we find that the maximum likelihood estimator of σ^2, S^{**2}, is not unbiased. However, $\dfrac{n}{n-1}S^{**2}$ is an unbiased estimator of σ^2.

- If the efficient estimator exists, i.e., if there exists the estimator which satisfies the equality in the Cramer-Rao inequality, the maximum likelihood estimator is efficient.

For large sample, as $n \longrightarrow \infty$, the maximum likelihood estimator of θ, $\hat{\theta}_n$, has the following property:

$$\sqrt{n}(\hat{\theta}_n - \theta) \longrightarrow N(0, \sigma^2(\theta)), \tag{1.15}$$

where

$$\sigma^2(\theta) = \frac{1}{\mathrm{E}\left(\left(\frac{\partial \log f(X;\theta)}{\partial \theta}\right)^2\right)}.$$

(1.15) indicates that the maximum likelihood estimator has consistency, asymptotic unbiasedness, asymptotic efficiency and asymptotic normality. Asymptotic normality of the maximum likelihood estimator comes from the central limit theorem discussed in Section 1.6.3. Even though the underlying distribution is not normal, i.e., even though $f(x;\theta)$ is not normal, the maximum likelihood estimator is asymptotically normally distributed. Note that the properties of $n \longrightarrow \infty$ are called the asymptotic properties, which include consistency, asymptotic normality and so on.

By normalizing, as $n \longrightarrow \infty$, we obtain as follows:

$$\frac{\sqrt{n}(\hat{\theta}_n - \theta)}{\sigma(\theta)} = \frac{\hat{\theta}_n - \theta}{\sigma(\theta)/\sqrt{n}} \longrightarrow N(0,1).$$

As another representation, when n is large, we can approximate the distribution of $\hat{\theta}_n$ as follows:

$$\hat{\theta}_n \sim N\Big(\theta, \frac{\sigma^2(\theta)}{n}\Big).$$

This implies that when $n \longrightarrow \infty$, $\hat{\theta}_n$ approaches the lower bound of Cramer-Rao inequality: $\sigma^2(\theta)/n$, which property is called an asymptotic efficiency.

Moreover, replacing θ in variance $\sigma^2(\theta)$ by $\hat{\theta}_n$, when $n \longrightarrow \infty$, we have the following property:

$$\frac{\hat{\theta}_n - \theta}{\sigma(\hat{\theta}_n)/\sqrt{n}} \longrightarrow N(0,1), \tag{1.16}$$

which also comes from the central limit theorem.

Practically, when n is large, we approximately use as follows:

$$\hat{\theta}_n \sim N\Big(\theta, \frac{\sigma^2(\hat{\theta}_n)}{n}\Big). \tag{1.17}$$

Proof of (1.15): By the central limit theorem (1.11) on p.34,

$$\frac{1}{\sqrt{n}} \sum_{i=1}^{n} \frac{\partial \log f(X_i;\theta)}{\partial \theta} \longrightarrow N\Big(0, \frac{1}{\sigma^2(\theta)}\Big), \tag{1.18}$$

where $\sigma^2(\theta)$ is defined in (1.14), i.e., $\mathrm{V}(\partial \log f(X_i;\theta)/\partial\theta) = 1/\sigma^2(\theta)$. As shown in (1.46) of Appendix 1.4, note that $\mathrm{E}(\partial \log f(X_i;\theta)/\partial\theta) = 0$. We can apply the central limit theorem, taking $\partial \log f(X_i;\theta)/\partial\theta$ as the ith random variable.

By performing the first order Taylor series expansion around $\hat{\theta}_n = \theta$, we have the following approximation:

$$\begin{aligned} 0 &= \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \frac{\partial \log f(X_i; \hat{\theta}_n)}{\partial \theta} \\ &= \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \frac{\partial \log f(X_i; \theta)}{\partial \theta} + \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \frac{\partial^2 \log f(X_i; \theta)}{\partial \theta^2} (\hat{\theta}_n - \theta) + \cdots. \end{aligned}$$

Therefore, the following approximation also holds:

$$\frac{1}{\sqrt{n}} \sum_{i=1}^{n} \frac{\partial \log f(X_i; \theta)}{\partial \theta} \approx -\frac{1}{\sqrt{n}} \sum_{i=1}^{n} \frac{\partial^2 \log f(X_i; \theta)}{\partial \theta^2} (\hat{\theta}_n - \theta).$$

From (1.18) and the above equation, we obtain:

$$-\frac{1}{n} \sum_{i=1}^{n} \frac{\partial^2 \log f(X_i; \theta)}{\partial \theta^2} \sqrt{n}(\hat{\theta}_n - \theta) \longrightarrow N\Big(0, \frac{1}{\sigma^2(\theta)}\Big).$$

The law of large numbers indicates as follows:

$$-\frac{1}{n} \sum_{i=1}^{n} \frac{\partial^2 \log f(X_i; \theta)}{\partial \theta^2} \longrightarrow -\mathrm{E}\Big(\frac{\partial^2 \log f(X_i; \theta)}{\partial \theta^2}\Big) = \frac{1}{\sigma^2(\theta)},$$

where the last equality is from (1.14). Thus, we have the following relation:

$$-\frac{1}{n} \sum_{i=1}^{n} \frac{\partial^2 \log f(X_i; \theta)}{\partial \theta^2} \sqrt{n}(\hat{\theta}_n - \theta) \longrightarrow \frac{1}{\sigma^2(\theta)} \sqrt{n}(\hat{\theta}_n - \theta) \longrightarrow N\Big(0, \frac{1}{\sigma^2(\theta)}\Big)$$

Therefore, the asymptotic normality of the maximum likelihood estimator is obtained as follows:

$$\sqrt{n}(\hat{\theta}_n - \theta) \longrightarrow N(0, \sigma^2(\theta)).$$

Thus, (1.15) is obtained.

1.7.6 Interval Estimation

In Sections 1.7.1 – 1.7.5, the point estimation is discussed. It is important to know where the true parameter value of θ is likely to lie.

Suppose that the population distribution is given by $f(x; \theta)$. Using the random sample $X_1, X_2, \cdots, X_n$ drawn from the population distribution, we construct the two statistics, say, $\hat{\theta}_U(X_1, X_2, \cdots, X_n; \theta^*)$ and $\hat{\theta}_L(X_1, X_2, \cdots, X_n; \theta^{**})$, where θ^* and θ^{**} denote the constant values which satisfy:

$$P(\theta^* < \hat{\theta}_n < \theta^{**}) = 1 - \alpha, \tag{1.19}$$

for $\theta^{**} > \theta^{*}$. Note that $\hat{\theta}_n$ depends on X_1, X_2, $\cdots$, X_n as well as θ, i.e., $\hat{\theta}_n \equiv \hat{\theta}_n(X_1, X_2, \cdots, X_n; \theta)$. Now we assume that we can solve (1.19) with respect to θ, which is rewritten as follows:

$$P\Big(\hat{\theta}_L(X_1, X_2, \cdots, X_n; \theta^{*}) < \theta < \hat{\theta}_U(X_1, X_2, \cdots, X_n; \theta^{**})\Big) = 1 - \alpha. \tag{1.20}$$

(1.20) implies that θ lies on the interval $\Big(\hat{\theta}_L(X_1, X_2, \cdots, X_n; \theta^{*}), \hat{\theta}_U(X_1, X_2, \cdots, X_n; \theta^{**})\Big)$ with probability $1 - \alpha$. Depending on a functional form of $\hat{\theta}_n(X_1, X_2, \cdots, X_n; \theta)$, we possibly have the situation that θ^{*} and θ^{**} are switched with each other.

Now, we replace the random variables X_1, X_2, $\cdots$, X_n by the experimental values x_1, x_2, $\cdots$, x_n. Then, we say that the interval:

$$\Big(\hat{\theta}_L(x_1, x_2, \cdots, x_n; \theta^{*}),\ \hat{\theta}_U(x_1, x_2, \cdots, x_n; \theta^{**})\Big)$$

is called the $100 \times (1 - \alpha)\%$ **confidence interval** of θ. Thus, estimating the interval is known as the **interval estimation**, which is distinguished from the point estimation. In the interval, $\hat{\theta}_L(x_1, x_2, \cdots, x_n; \theta^{*})$ is known as the **lower bound** of the confidence interval, while $\hat{\theta}_U(x_1, x_2, \cdots, x_n; \theta^{**})$ is the **upper bound** of the confidence interval.

Given probability α, the $\hat{\theta}_L(X_1, X_2, \cdots, X_n; \theta^{*})$ and $\hat{\theta}_U(X_1, X_2, \cdots, X_n; \theta^{**})$ which satisfies equation (1.20) are not unique. For estimation of the unknown parameter θ, it is more optimal to minimize the width of the confidence interval. Therefore, we should choose θ^{*} and θ^{**} which minimizes the width $\hat{\theta}_U(X_1, X_2, \cdots, X_n; \theta^{**}) - \hat{\theta}_L(X_1, X_2, \cdots, X_n; \theta^{*})$.

Interval Estimation of $\overline{X}$: Let X_1, X_2, $\cdots$, X_n be mutually independently and identically distributed random variables. X_i has a distribution with mean μ and variance σ^2. From the central limit theorem,

$$\frac{\overline{X} - \mu}{\sigma / \sqrt{n}} \longrightarrow N(0, 1).$$

Replacing σ^2 by its estimator S^2 (or S^{**2}),

$$\frac{\overline{X} - \mu}{S / \sqrt{n}} \longrightarrow N(0, 1).$$

Therefore, when n is large enough,

$$P(z^{*} < \frac{\overline{X} - \mu}{S / \sqrt{n}} < z^{**}) = 1 - \alpha,$$

where z^{*} and z^{**} ($z^{*} < z^{**}$) are percent points from the standard normal density function. Solving the inequality above with respect to μ, the following expression is obtained.

$$P\Big(\overline{X} - z^{**}\frac{S}{\sqrt{n}} < \mu < \overline{X} - z^{*}\frac{S}{\sqrt{n}}\Big) = 1 - \alpha,$$

where $\hat{\theta}_L$ and $\hat{\theta}_U$ correspond to $\overline{X} - z^{**}S/\sqrt{n}$ and $\overline{X} - z^{*}S/\sqrt{n}$, respectively.

The length of the confidence interval is given by:

$$\hat{\theta}_U - \hat{\theta}_L = \frac{S}{\sqrt{n}}(z^{**} - z^{*}),$$

which should be minimized subject to:

$$\int_{z^{*}}^{z^{**}} f(x)\,\mathrm{d}x = 1 - \alpha,$$

i.e.,

$$F(z^{**}) - F(z^{*}) = 1 - \alpha,$$

where $F(\cdot)$ denotes the standard normal cumulative distribution function.

Solving the minimization problem above, we can obtain the conditions that $f(z^{*}) = f(z^{**})$ for $z^{*} < z^{**}$ and that $f(x)$ is symmetric. Therefore, we have:

$$-z^{*} = z^{**} = z_{\alpha/2},$$

where $z_{\alpha/2}$ denotes the $100 \times \alpha/2$ percent point from the standard normal density function.

Accordingly, replacing the estimators $\overline{X}$ and S^2 by their estimates $\overline{x}$ and s^2, the $100 \times (1-\alpha)\%$ confidence interval of μ is approximately represented as:

$$\Big(\overline{x} - z_{\alpha/2}\frac{s}{\sqrt{n}},\ \overline{x} + z_{\alpha/2}\frac{s}{\sqrt{n}}\Big),$$

for large n.

For now, we do not impose any assumptions on the distribution of X_i. If we assume that X_i is normal, $\sqrt{n}(\overline{X} - \mu)/S$ has a t distribution with $n-1$ degrees of freedom for any n. Therefore, $100 \times (1-\alpha)\%$ confidence interval of μ is given by:

$$\Big(\overline{x} - t_{\alpha/2}(n-1)\frac{s}{\sqrt{n}},\ \overline{x} + t_{\alpha/2}(n-1)\frac{s}{\sqrt{n}}\Big),$$

where $t_{\alpha/2}(n-1)$ denotes the $100 \times \alpha/2$ percent point of the t distribution with $n-1$ degrees of freedom. See Section 2.2.10, p.115 for the t distribution.

Interval Estimation of $\hat{\theta}_n$: Let $X_1, X_2, \cdots, X_n$ be mutually independently and identically distributed random variables. X_i has the probability density function $f(x_i;\theta)$. Suppose that $\hat{\theta}_n$ represents the maximum likelihood estimator of θ.

From (1.17), we can approximate the $100 \times (1-\alpha)\%$ confidence interval of θ as follows:

$$\Big(\hat{\theta}_n - z_{\alpha/2}\frac{\sigma(\hat{\theta}_n)}{\sqrt{n}},\ \hat{\theta}_n + z_{\alpha/2}\frac{\sigma(\hat{\theta}_n)}{\sqrt{n}}\Big).$$

Table 1.1: Type I and Type II Errors

	H_0 is true.	H_0 is false.
Acceptance of H_0	Correct judgment	Type II Error (Probability β)
Rejection of H_0	Type I Error (Probability α = Significance Level)	Correct judgment ($1-\beta$ = Power)

1.8 Testing Hypothesis

1.8.1 Basic Concepts in Testing Hypothesis

Given the population distribution $f(x;\theta)$, we want to judge from the observed values $x_1, x_2, \cdots, x_n$ whether the hypothesis on the parameter θ, e.g. $\theta = \theta_0$, is correct or not. The hypothesis that we want to test is called the **null hypothesis**, which is denoted by $H_0 : \theta = \theta_0$. The hypothesis against the null hypothesis, e.g. $\theta \neq \theta_0$, is called the **alternative hypothesis**, which is denoted by $H_1 : \theta \neq \theta_0$.

Type I and Type II Errors: When we test the null hypothesis H_0, as shown in Table 1.1 we have four cases, i.e., (i) we accept H_0 when H_0 is true, (ii) we reject H_0 when H_0 is true, (iii) we accept H_0 when H_0 is false, and (iv) we reject H_0 when H_0 is false. (i) and (iv) are correct judgments, while (ii) and (iii) are not correct. (ii) is called a **type I error** and (iii) is called a **type II error**. The probability which a type I error occurs is called the **significance level**, which is denoted by α, and the probability of committing a type II error is denoted by β. Probability of (iv) is called the **power** or the **power function**, because it is a function of the parameter θ.

Testing Procedures: The testing procedure is summarized as follows.

1. Construct the null hypothesis (H_0) on the parameter.
2. Consider an appropriate statistic, which is called a **test statistic**. Derive a distribution function of the test statistic when H_0 is true.
3. From the observed data, compute the observed value of the test statistic.
4. Compare the distribution and the observed value of the test statistic. When the observed value of the test statistic is in the tails of the distribution, we consider that H_0 is not likely to occur and we reject H_0.

The region that H_0 is unlikely to occur and accordingly H_0 is rejected is called the **rejection region** or the **critical region**, denoted by R. Conversely, the region that

H_0 is likely to occur and accordingly H_0 is accepted is called the **acceptance region**, denoted by A.

Using the rejection region R and the acceptance region A, the type I and II errors and the power are formulated as follows. Suppose that the test statistic is give by $T = T(X_1, X_2, \cdots, X_n)$. The probability of committing a type I error, i.e., the significance level α, is given by:

$$P(T(X_1, X_2, \cdots, X_n) \in R | H_0 \text{ is true}) = \alpha,$$

which is the probability that rejects H_0 when H_0 is true. Conventionally, the significance level $\alpha = 0.1, 0.05, 0.01$ is chosen in practice. The probability of committing a type II error, i.e., β, is represented as:

$$P(T(X_1, X_2, \cdots, X_n) \in A | H_0 \text{ is not true}) = \beta,$$

which corresponds to the probability that accepts H_0 when H_0 is not true. The power is defined as $1 - \beta$, i.e.,

$$P(T(X_1, X_2, \cdots, X_n) \in R | H_0 \text{ is not true}) = 1 - \beta,$$

which is the probability that rejects H_0 when H_0 is not true.

1.8.2 Power Function

Let X_1, X_2, $\cdots$, X_n be mutually independently, identically and normally distributed with mean μ and variance σ^2. Assume that σ^2 is known.

In Figure 1.3, we consider the hypothesis on the population mean μ, i.e., the null hypothesis $H_0 : \mu = \mu_0$ against the alternative hypothesis $H_1 : \mu = \mu_1$, where $\mu_1 > \mu_0$ is taken. The dark shadow area corresponds to the probability of committing a type I error, i.e., the significance level, while the light shadow area indicates the probability of committing a type II error. The probability of the right-hand side of f^* in the distribution under H_1 represents the power of the test, i.e., $1 - \beta$.

In the case of normal population, the distribution of sample mean $\overline{X}$ is given by:

$$\overline{X} \sim N\Big(\mu, \frac{\sigma^2}{n}\Big).$$

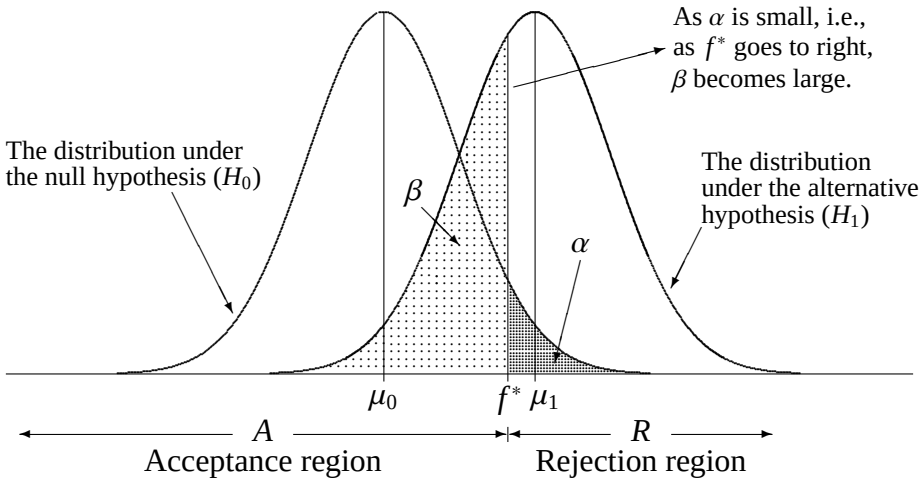
For the distribution of $\overline{X}$, see the moment-generating function of $\overline{X}$ in Theorem on p.29. By normalization, we have:

$$\frac{\overline{X} - \mu}{\sigma / \sqrt{n}} \sim N(0, 1).$$

Therefore, under the null hypothesis $H_0 : \mu = \mu_0$, we obtain:

$$\frac{\overline{X} - \mu_0}{\sigma / \sqrt{n}} \sim N(0, 1),$$

Figure 1.3: Type I Error (α) and Type II Error (β)



where μ is replaced by μ_0. Since the significance level α is the probability which rejects H_0 when H_0 is true, it is given by:

$$\alpha = P\Big(\overline{X} > \mu_0 + z_\alpha \frac{\sigma}{\sqrt{n}}\Big),$$

where z_α denotes $100\times\alpha$ percent point of the standard normal density function. Therefore, the rejection region is given by: $\overline{X} > \mu_0 + z_\alpha \sigma/\sqrt{n}$.

Since the power $1-\beta$ is the probability which rejects H_0 when H_1 is true, it is given by:

$$\begin{aligned} 1-\beta &= P\Big(\overline{X} > \mu_0 + z_\alpha \frac{\sigma}{\sqrt{n}}\Big) = P\Big(\frac{\overline{X}-\mu_1}{\sigma/\sqrt{n}} > \frac{\mu_0-\mu_1}{\sigma/\sqrt{n}} + z_\alpha\Big) \\ &= 1 - P\Big(\frac{\overline{X}-\mu_1}{\sigma/\sqrt{n}} < \frac{\mu_0-\mu_1}{\sigma/\sqrt{n}} + z_\alpha\Big) = 1 - F\Big(\frac{\mu_0-\mu_1}{\sigma/\sqrt{n}} + z_\alpha\Big), \end{aligned}$$

where $F(\cdot)$ represents the standard normal cumulative distribution function, which is given by $F(x) = \int_{-\infty}^{x} (2\pi)^{-1/2} \exp(-\frac{1}{2}t^2)\, \mathrm{d}t$. The power function is a function of μ_1, given μ_0 and α.

1.8.3 Testing Hypothesis on Population Mean

Let $X_1, X_2, \cdots, X_n$ be mutually independently, identically and normally distributed with mean μ and variance σ^2. Assume that σ^2 is known.

Consider testing the null hypothesis $H_0 : \mu = \mu_0$. When the null hypothesis H_0 is true, the distribution of $\overline{X}$ is given by:

$$\frac{\overline{X}-\mu_0}{\sigma/\sqrt{n}} \sim N(0,1).$$

Therefore, the test statistic is given by: $\sqrt{n}(\overline{X}-\mu_0)/\sigma$, while the test statistic value is: $\sqrt{n}(\overline{x}-\mu_0)/\sigma$, where the sample mean $\overline{X}$ is replaced by the observed value $\overline{x}$.

Depending on the alternative hypothesis, we have the following three cases.

1. **The alternative hypothesis $H_1: \mu < \mu_0$** (one-sided test):

   We have: $P\Big(\dfrac{\overline{X}-\mu_0}{\sigma/\sqrt{n}} < -z_\alpha\Big) = \alpha$. Therefore, when $\dfrac{\overline{x}-\mu_0}{\sigma/\sqrt{n}} < -z_\alpha$, we reject the null hypothesis $H_0: \mu = \mu_0$ at the significance level α.

2. **The alternative hypothesis $H_1: \mu > \mu_0$** (one-sided test):

   We have: $P\Big(\dfrac{\overline{X}-\mu_0}{\sigma/\sqrt{n}} > z_\alpha\Big) = \alpha$. Therefore, when $\dfrac{\overline{x}-\mu_0}{\sigma/\sqrt{n}} > z_\alpha$, we reject the null hypothesis $H_0: \mu = \mu_0$ at the significance level α.

3. **The alternative hypothesis $H_1: \mu \neq \mu_0$** (two-sided test):

   We have: $P\Big(\Big|\dfrac{\overline{X}-\mu_0}{\sigma/\sqrt{n}}\Big| > z_{\alpha/2}\Big) = \alpha$. Therefore, when $\Big|\dfrac{\overline{x}-\mu_0}{\sigma/\sqrt{n}}\Big| > z_{\alpha/2}$, we reject the null hypothesis $H_0: \mu = \mu_0$ at the significance level α.

When the sample size n is large enough, the testing procedure above can be applied to the cases: (i) the distribution of X_i is not known and (ii) σ^2 is replaced by its estimator S^2 (in the case where σ^2 is not known).

1.8.4 Wald Test

From (1.16), under the null hypothesis $H_0: \theta = \theta_0$, as $n \longrightarrow \infty$, the maximum likelihood estimator $\hat{\theta}_n$ is distributed as follows:

$$\frac{\hat{\theta}_n - \theta_0}{\sigma(\hat{\theta}_n)/\sqrt{n}} \sim N(0,1).$$

For $H_0: \theta = \theta_0$ and $H_1: \theta \neq \theta_0$, replacing $X_1, X_2, \cdots, X_n$ in $\hat{\theta}_n$ by the observed values $x_1, x_2, \cdots, x_n$, we obtain the following testing procedure:

1. If we have:
$$\left|\frac{\hat{\theta}_n - \theta_0}{\sigma(\hat{\theta}_n)/\sqrt{n}}\right| > z_{\alpha/2},$$
   we reject the null hypothesis H_0 at the significance level α, because the probability which H_0 occurs is small enough.

2. As for $H_0: \theta = \theta_0$ and $H_1: \theta > \theta_0$, if we have:
$$\frac{\hat{\theta}_n - \theta_0}{\sigma(\hat{\theta}_n)/\sqrt{n}} > z_\alpha,$$
   we reject H_0 at the significance level α.

3. For $H_0 : \theta = \theta_0$ and $H_1 : \theta < \theta_0$, when we have the following:

$$\frac{\hat{\theta}_n - \theta_0}{\sigma(\hat{\theta}_n)/\sqrt{n}} < -z_\alpha,$$

we reject H_0 at the significance level α.

The testing procedure introduced here is called the **Wald test**.

Example 1.18: $X_1, X_2, \cdots, X_n$ are mutually independently, identically and exponentially distributed. Consider the following exponential probability density function:

$$f(x;\gamma) = \gamma e^{-\gamma x},$$

for $0 < x < \infty$.

Using the Wald test, we want to test the null hypothesis $H_0 : \gamma = \gamma_0$ against the alternative hypothesis $H_1 : \gamma \neq \gamma_0$.

Generally, as $n \longrightarrow \infty$, the distribution of the maximum likelihood estimator of the parameter γ, $\hat{\gamma}_n$, is asymptotically represented as:

$$\frac{\hat{\gamma}_n - \gamma}{\sigma(\hat{\gamma}_n)/\sqrt{n}} \sim N(0,1),$$

where

$$\sigma^2(\gamma) = \left(\mathrm{E}\left(\left(\frac{\mathrm{d}\log f(X;\gamma)}{\mathrm{d}\gamma}\right)^2\right)\right)^{-1} = -\left(\mathrm{E}\left(\frac{\mathrm{d}^2\log f(X;\gamma)}{\mathrm{d}\gamma^2}\right)\right)^{-1}.$$

See (1.14) and (1.16) for the above properties on the maximum likelihood estimator.

Therefore, under the null hypothesis $H_0 : \gamma = \gamma_0$, when n is large enough, we have the following distribution:

$$\frac{\hat{\gamma}_n - \gamma_0}{\sigma(\hat{\gamma}_n)/\sqrt{n}} \sim N(0,1).$$

As for the null hypothesis $H_0 : \gamma = \gamma_0$ against the alternative hypothesis $H_1 : \gamma \neq \gamma_0$, if we have:

$$\left|\frac{\hat{\gamma}_n - \gamma_0}{\sigma(\hat{\gamma}_n)/\sqrt{n}}\right| > z_{\alpha/2},$$

we can reject H_0 at the significance level α.

We need to derive $\sigma^2(\gamma)$ and $\hat{\gamma}_n$ to perform the testing procedure. First, $\sigma^2(\gamma)$ is given by:

$$\sigma^2(\gamma) = -\left(\mathrm{E}\left(\frac{\mathrm{d}^2\log f(X;\gamma)}{\mathrm{d}\gamma^2}\right)\right)^{-1} = \gamma^2.$$

Note that the first- and the second-derivatives of $\log f(X;\gamma)$ with respect to γ are given by:

$$\frac{\mathrm{d}\log f(X;\gamma)}{\mathrm{d}\gamma} = \frac{1}{\gamma} - X, \qquad \frac{\mathrm{d}^2\log f(X;\gamma)}{\mathrm{d}\gamma^2} = -\frac{1}{\gamma^2}.$$

Next, the maximum likelihood estimator of γ, i.e., $\hat{\gamma}_n$, is obtained as follows. Since X_1, X_2 $\cdots$, X_n are mutually independently and identically distributed, the likelihood function $l(\gamma)$ is given by:

$$l(\gamma) = \prod_{i=1}^{n} f(x_i;\gamma) = \prod_{i=1}^{n} \gamma e^{-\gamma x_i} = \gamma^n e^{-\gamma \sum x_i}.$$

Therefore, the log-likelihood function is written as:

$$\log l(\gamma) = n\log(\gamma) - \gamma\sum_{i=1}^{n} x_i.$$

We obtain the value of γ which maximizes $\log l(\gamma)$. Solving the following equation:

$$\frac{\mathrm{d}\log l(\gamma)}{\mathrm{d}\gamma} = \frac{n}{\gamma} - \sum_{i=1}^{n} x_i = 0,$$

the maximum likelihood estimator of γ, i.e., $\hat{\gamma}_n$ is represented as:

$$\hat{\gamma}_n = \frac{n}{\sum_{i=1}^{n} X_i} = \frac{1}{\overline{X}}.$$

Then, we have the following:

$$\frac{\hat{\gamma}_n - \gamma}{\sigma(\hat{\gamma}_n)/\sqrt{n}} = \frac{\hat{\gamma}_n - \gamma}{\hat{\gamma}_n/\sqrt{n}} \longrightarrow N(0,1),$$

where $\hat{\gamma}_n$ is given by $1/\overline{X}$.

For $H_0 : \gamma = \gamma_0$ and $H_1 : \gamma \neq \gamma_0$, if we have:

$$\left|\frac{\hat{\gamma}_n - \gamma_0}{\hat{\gamma}_n/\sqrt{n}}\right| > z_{\alpha/2},$$

we reject H_0 at the significance level α.

1.8.5 Likelihood Ratio Test

Suppose that the population distribution is given by $f(x;\theta)$, where $\theta = (\theta_1, \theta_2)$. Consider testing the null hypothesis $\theta_1 = \theta_1^*$ against the alternative hypothesis $H_1 : \theta_1 \neq \theta_1^*$,

using the observed values $(x_1, x_2, \cdots, x_n)$ corresponding to the random sample $(X_1, X_2, \cdots, X_n)$.

Let θ_1 and θ_2 be $1 \times k_1$ and $1 \times k_2$ vectors, respectively. Therefore, $\theta = (\theta_1, \theta_2)$ denotes a $1 \times (k_1 + k_2)$ vector. Since we take the null hypothesis as $H_0 : \theta_1 = \theta_1^*$, the number of restrictions is given by k_1, which is equal to the dimension of θ_1.

The likelihood function is written as:

$$l(\theta_1, \theta_2) = \prod_{i=1}^{n} f(x_i; \theta_1, \theta_2).$$

Let $(\widetilde{\theta}_1, \widetilde{\theta}_2)$ be the maximum likelihood estimator of (θ_1, θ_2). That is, $(\widetilde{\theta}_1, \widetilde{\theta}_2)$ indicates the solution of (θ_1, θ_2), obtained from the following equations:

$$\frac{\partial l(\theta_1, \theta_2)}{\partial \theta_1} = 0, \quad \frac{\partial l(\theta_1, \theta_2)}{\partial \theta_2} = 0.$$

The solution $(\widetilde{\theta}_1, \widetilde{\theta}_2)$ is called the **unconstrained maximum likelihood estimator**, because the null hypothesis $H_0 : \theta_1 = \theta_1^*$ is not taken into account.

Let $\hat{\theta}_2$ be the maximum likelihood estimator of θ_2 under the null hypothesis $H_0 : \theta_1 = \theta_1^*$. That is, $\hat{\theta}_2$ is a solution of the following equation:

$$\frac{\partial l(\theta_1^*, \theta_2)}{\partial \theta_2} = 0.$$

The solution $\hat{\theta}_2$ is called the **constrained maximum likelihood estimator** of θ_2, because the likelihood function is maximized with respect to θ_2 subject to the constraint $\theta_1 = \theta_1^*$.

Define λ as follows:

$$\lambda = \frac{l(\theta_1^*, \hat{\theta}_2)}{l(\widetilde{\theta}_1, \widetilde{\theta}_2)},$$

which is called the **likelihood ratio**.

As n goes to infinity, it is known that we have:

$$-2\log(\lambda) \sim \chi^2(k_1),$$

where k_1 denotes the number of the constraints.

Let $\chi_\alpha^2(k_1)$ be the $100 \times \alpha$ percent point from the chi-square distribution with k_1 degrees of freedom. When $-2\log(\lambda) > \chi_\alpha^2(k_1)$, we reject the null hypothesis $H_0 : \theta_1 = \theta_1^*$ at the significance level α. If $-2\log(\lambda)$ is close to zero, we accept the null hypothesis. When $(\theta_1^*, \hat{\theta}_2)$ is close to $(\widetilde{\theta}_1, \widetilde{\theta}_2)$, $-2\log(\lambda)$ approaches zero.

The likelihood ratio test is useful in the case where it is not easy to derive the distribution of $(\widetilde{\theta}_1, \widetilde{\theta}_2)$.

Example 1.19: $X_1, X_2, \cdots, X_n$ are mutually independently, identically and exponentially distributed. Consider the following exponential probability density function:

$$f(x;\gamma) = \gamma e^{-\gamma x},$$

for $0 < x < \infty$.

Using the likelihood ratio test, we want to test the null hypothesis $H_0 : \gamma = \gamma_0$ against the alternative hypothesis $H_1 : \gamma \neq \gamma_0$. Remember that in Example 1.18 we test the hypothesis with the Wald test.

In this case, the likelihood ratio is given by:

$$\lambda = \frac{l(\gamma_0)}{l(\hat{\gamma}_n)},$$

where $\hat{\gamma}_n$ is derived in Example 1.18, i.e.,

$$\hat{\gamma}_n = \frac{n}{\sum_{i=1}^{n} X_i} = \frac{1}{\overline{X}}.$$

Since the number of the constraint is equal to one, as the sample size n goes to infinity we have the following asymptotic distribution:

$$-2\log\lambda \longrightarrow \chi^2(1).$$

The likelihood ratio is computed as follows:

$$\lambda = \frac{l(\gamma_0)}{l(\hat{\gamma}_n)} = \frac{\gamma_0^n e^{-\gamma_0 \sum X_i}}{\hat{\gamma}_n^n e^{-n}}.$$

If $-2\log\lambda > \chi^2_\alpha(1)$, we reject the null hypothesis $H_0 : \mu = \mu_0$ at the significance level α. Note that $\chi^2_\alpha(1)$ denotes the $100 \times \alpha$ percent point from the chi-square distribution with one degree of freedom.

Example 1.20: Suppose that $X_1, X_2, \cdots, X_n$ are mutually independently, identically and normally distributed with mean zero and variance σ^2.

The normal probability density function with mean μ and variance σ^2 is given by:

$$f(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}.$$

By the likelihood ratio test, we want to test the null hypothesis $H_0 : \mu = \mu_0$ against the alternative hypothesis $H_1 : \mu \neq \mu_0$.

The likelihood ratio is given by:

$$\lambda = \frac{l(\mu_0, \widetilde{\sigma}^2)}{l(\hat{\mu}, \hat{\sigma}^2)},$$

where $\widetilde{\sigma}^2$ is the constrained maximum likelihood estimator with the constraint $\mu = \mu_0$, while $(\hat{\mu}, \hat{\sigma}^2)$ denotes the unconstrained maximum likelihood estimator. In this case, since the number of the constraint is one, the asymptotic distribution is as follows:

$$-2\log\lambda \longrightarrow \chi^2(1).$$

Now, we derive $l(\mu_0, \widetilde{\sigma}^2)$ and $l(\hat{\mu}, \hat{\sigma}^2)$. $l(\mu, \sigma^2)$ is written as:

$$\begin{aligned} l(\mu, \sigma^2) &= f(x_1, x_2, \cdots, x_n; \mu, \sigma^2) = \prod_{i=1}^{n} f(x_i; \mu, \sigma^2) \\ &= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{1}{2\sigma^2}(x_i - \mu)^2\Big) \\ &= (2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \mu)^2\Big). \end{aligned}$$

The log-likelihood function $\log l(\mu, \sigma^2)$ is represented as:

$$\log l(\mu, \sigma^2) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \mu)^2.$$

For the numerator of the likelihood ratio, under the constraint $\mu = \mu_0$, maximize $\log l(\mu_0, \sigma^2)$ with respect to σ^2. Since we obtain the first-derivative:

$$\frac{\partial \log l(\mu_0, \sigma^2)}{\partial \sigma^2} = -\frac{n}{2}\frac{1}{\sigma^2} + \frac{1}{2\sigma^4}\sum_{i=1}^{n}(x_i - \mu_0)^2 = 0,$$

the constrained maximum likelihood estimator $\widetilde{\sigma}^2$ is given by:

$$\widetilde{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu_0)^2.$$

Therefore, replacing σ^2 by $\widetilde{\sigma}^2$, $l(\mu_0, \widetilde{\sigma}^2)$ is written as:

$$l(\mu_0, \widetilde{\sigma}^2) = (2\pi\widetilde{\sigma}^2)^{-n/2} \exp\Big(-\frac{1}{2\widetilde{\sigma}^2}\sum_{i=1}^{n}(x_i - \mu_0)^2\Big) = (2\pi\widetilde{\sigma}^2)^{-n/2} \exp\Big(-\frac{n}{2}\Big).$$

For the denominator of the likelihood ratio, because the unconstrained maximum likelihood estimators are obtained as:

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i, \qquad \hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{\mu})^2,$$

$l(\hat{\mu}, \hat{\sigma}^2)$ is written as:

$$l(\hat{\mu}, \hat{\sigma}^2) = (2\pi\hat{\sigma}^2)^{-n/2} \exp\Big(-\frac{1}{2\hat{\sigma}^2}\sum_{i=1}^{n}(x_i - \hat{\mu})^2\Big) = (2\pi\hat{\sigma}^2)^{-n/2} \exp\Big(-\frac{n}{2}\Big).$$

Thus, the likelihood ratio is given by:

$$\lambda = \frac{l(\mu_0, \widetilde{\sigma}^2)}{l(\hat{\mu}, \hat{\sigma}^2)} = \frac{(2\pi\widetilde{\sigma}^2)^{-n/2} \exp\left(-\frac{n}{2}\right)}{(2\pi\hat{\sigma}^2)^{-n/2} \exp\left(-\frac{n}{2}\right)} = \left(\frac{\widetilde{\sigma}^2}{\hat{\sigma}^2}\right)^{-n/2}.$$

Asymptotically, we have:

$$-2\log\lambda = n(\log\widetilde{\sigma}^2 - \log\hat{\sigma}^2) \sim \chi^2(1).$$

When $-2\log\lambda > \chi^2_\alpha(1)$, we reject the null hypothesis $H_0 : \mu = \mu_0$ at the significance level α.

1.9 Regression Analysis

1.9.1 Setup of the Model

When (X_1, Y_1), (X_2, Y_2), $\cdots$, (X_n, Y_n) are available, suppose that there is a linear relationship between Y and X, i.e.,

$$Y_i = \beta_1 + \beta_2 X_i + u_i, \tag{1.21}$$

for $i = 1, 2, \cdots, n$.

X_i and Y_i denote the ith observations. Y_i is called the **dependent variable** or the **unexplanatory variable**, while X_i is known as the **independent variable** or the **explanatory variable**. β_1 and β_2 are unknown **parameter**s to be estimated. u_i is the unobserved **error term** assumed to be a random variable with mean zero and variance σ^2. β_1 and β_2 are called the **regression coefficient**s.

X_i is assumed to be nonstochastic, but Y_i is stochastic because Y_i depends on the error u_i. The error terms $u_1, u_2, \cdots, u_n$ are assumed to be mutually independently and identically distributed. It is assumed that u_i has a distribution with mean zero, i.e., $\mathrm{E}(u_i) = 0$ is assumed. Taking the expectation on both sides of equation (1.21), the expectation of Y_i is represented as:

$$\begin{aligned} \mathrm{E}(Y_i) &= \mathrm{E}(\beta_1 + \beta_2 X_i + u_i) = \beta_1 + \beta_2 X_i + \mathrm{E}(u_i) \\ &= \beta_1 + \beta_2 X_i, \end{aligned} \tag{1.22}$$

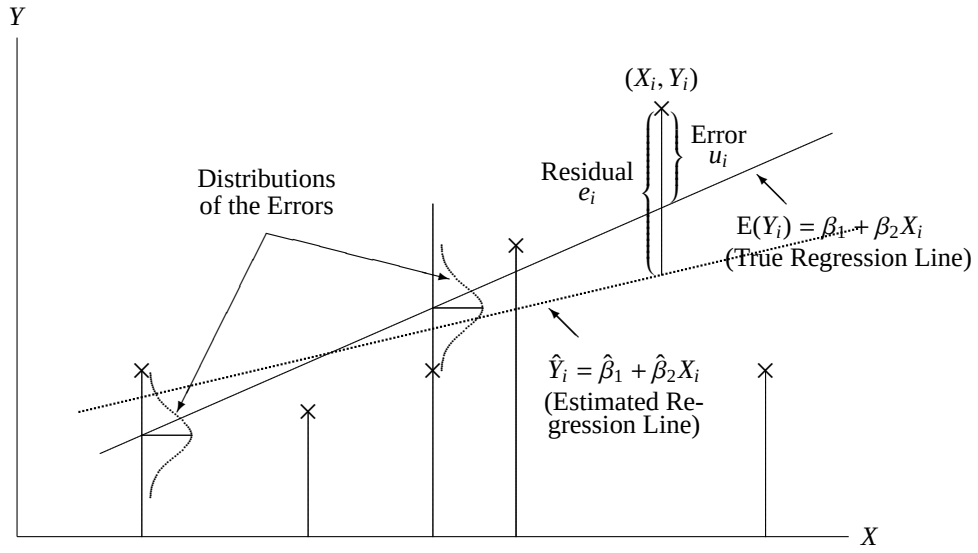
for $i = 1, 2, \cdots, n$. Using $\mathrm{E}(Y_i)$ we can rewrite (1.21) as $Y_i = \mathrm{E}(Y_i) + u_i$. Equation (1.22) represents the true regression line.

Let $\hat{\beta}_1$ and $\hat{\beta}_2$ be estimators of β_1 and β_2. Replacing (β_1, β_2) by $(\hat{\beta}_1, \hat{\beta}_2)$, equation (1.21) turns out to be:

$$Y_i = \hat{\beta}_1 + \hat{\beta}_2 X_i + e_i, \tag{1.23}$$

for $i = 1, 2, \cdots, n$, where e_i is called the **residual**. The residual e_i is taken as the experimental value of u_i.

Figure 1.4: True and Estimated Regression Lines



We define $\hat{Y}_i$ as follows:

$$\hat{Y}_i = \hat{\beta}_1 + \hat{\beta}_2 X_i, \tag{1.24}$$

for $i = 1, 2, \cdots, n$, which is interpreted as the predicted value of Y_i. Equation (1.24) indicates the estimated regression line, which is different from equation (1.22). Moreover, using $\hat{Y}_i$ we can rewrite (1.23) as $Y_i = \hat{Y}_i + e_i$.

Equations (1.22) and (1.24) are displayed in Figure 1.4. Consider the case of $n = 6$ for simplicity. $\times$ indicates the observed data series. The true regression line (1.22) is represented by the solid line, while the estimated regression line (1.24) is drawn with the dotted line. Based on the observed data, β_1 and β_2 are estimated as: $\hat{\beta}_1$ and $\hat{\beta}_2$.

In the next section, we consider how to obtain the estimates of β_1 and β_2, i.e., $\hat{\beta}_1$ and $\hat{\beta}_2$.

1.9.2 Ordinary Least Squares Estimation

Suppose that (X_1, Y_1), (X_2, Y_2), $\cdots$, (X_n, Y_n) are available. For the regression model (1.21), we consider estimating β_1 and β_2. Replacing β_1 and β_2 by their estimates $\hat{\beta}_1$ and $\hat{\beta}_2$, remember that the residual e_i is given by:

$$e_i = Y_i - \hat{Y}_i = Y_i - \hat{\beta}_1 - \hat{\beta}_2 X_i.$$

The sum of squared residuals is defined as follows:

$$S(\hat{\beta}_1, \hat{\beta}_2) = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (Y_i - \hat{\beta}_1 - \hat{\beta}_2 X_i)^2.$$

It might be plausible to choose the $\hat{\beta}_1$ and $\hat{\beta}_2$ which minimize the sum of squared residuals, i.e., $S(\hat{\beta}_1, \hat{\beta}_2)$. This method is called the **ordinary least squares (OLS) estimation**. To minimize $S(\hat{\beta}_1, \hat{\beta}_2)$ with respect to $\hat{\beta}_1$ and $\hat{\beta}_2$, we set the partial derivatives equal to zero:

$$\frac{\partial S(\hat{\beta}_1, \hat{\beta}_2)}{\partial \hat{\beta}_1} = -2 \sum_{i=1}^{n} (Y_i - \hat{\beta}_1 - \hat{\beta}_2 X_i) = 0,$$
$$\frac{\partial S(\hat{\beta}_1, \hat{\beta}_2)}{\partial \hat{\beta}_2} = -2 \sum_{i=1}^{n} X_i (Y_i - \hat{\beta}_1 - \hat{\beta}_2 X_i) = 0,$$

which yields the following two equations:

$$\overline{Y} = \hat{\beta}_1 + \hat{\beta}_2 \overline{X}, \tag{1.25}$$
$$\sum_{i=1}^{n} X_i Y_i = n\overline{X}\hat{\beta}_1 + \hat{\beta}_2 \sum_{i=1}^{n} X_i^2, \tag{1.26}$$

where $\overline{Y} = (1/n) \sum_{i=1}^{n} Y_i$ and $\overline{X} = (1/n) \sum_{i=1}^{n} X_i$. Multiplying (1.25) by $n\overline{X}$ and subtracting (1.26), we can derive $\hat{\beta}_2$ as follows:

$$\hat{\beta}_2 = \frac{\sum_{i=1}^{n} X_i Y_i - n\overline{X}\overline{Y}}{\sum_{i=1}^{n} X_i^2 - n\overline{X}^2} = \frac{\sum_{i=1}^{n} (X_i - \overline{X})(Y_i - \overline{Y})}{\sum_{i=1}^{n} (X_i - \overline{X})^2}. \tag{1.27}$$

From equation (1.25), $\hat{\beta}_1$ is directly obtained as follows:

$$\hat{\beta}_1 = \overline{Y} - \hat{\beta}_2 \overline{X}. \tag{1.28}$$

When the observed values are taken for Y_i and X_i for $i = 1, 2, \cdots, n$, we say that $\hat{\beta}_1$ and $\hat{\beta}_2$ are called the **ordinary least squares estimate**s (or simply the **least squares estimate**s) of β_1 and β_2. When Y_i for $i = 1, 2, \cdots, n$ are regarded as the random sample, we say that $\hat{\beta}_1$ and $\hat{\beta}_2$ are called the **ordinary least squares estimator**s (or the **least squares estimator**s) of β_1 and β_2.

1.9.3 Properties of Least Squares Estimator

Equation (1.27) is rewritten as:

$$\begin{aligned}\hat{\beta}_2 &= \frac{\sum_{i=1}^{n} (X_i - \overline{X})(Y_i - \overline{Y})}{\sum_{i=1}^{n} (X_i - \overline{X})^2} = \frac{\sum_{i=1}^{n} (X_i - \overline{X}) Y_i}{\sum_{i=1}^{n} (X_i - \overline{X})^2} - \frac{\overline{Y} \sum_{i=1}^{n} (X_i - \overline{X})}{\sum_{i=1}^{n} (X_i - \overline{X})^2} \\ &= \sum_{i=1}^{n} \frac{X_i - \overline{X}}{\sum_{i=1}^{n} (X_i - \overline{X})^2} Y_i = \sum_{i=1}^{n} \omega_i Y_i. \end{aligned} \tag{1.29}$$

In the third equality, $\sum_{i=1}^{n}(X_i - \overline{X}) = 0$ is utilized because of $\overline{X} = (1/n)\sum_{i=1}^{n} X_i$. In the fourth equality, ω_i is defined as:

$$\omega_i = \frac{X_i - \overline{X}}{\sum_{i=1}^{n}(X_i - \overline{X})^2}.$$

ω_i is nonstochastic because X_i is assumed to be nonstochastic. ω_i has the following properties:

$$\sum_{i=1}^{n}\omega_i = \sum_{i=1}^{n}\frac{X_i - \overline{X}}{\sum_{i=1}^{n}(X_i - \overline{X})^2} = \frac{\sum_{i=1}^{n}(X_i - \overline{X})}{\sum_{i=1}^{n}(X_i - \overline{X})^2} = 0, \tag{1.30}$$

$$\sum_{i=1}^{n}\omega_i X_i = \sum_{i=1}^{n}\omega_i(X_i - \overline{X}) = \frac{\sum_{i=1}^{n}(X_i - \overline{X})^2}{\sum_{i=1}^{n}(X_i - \overline{X})^2} = 1, \tag{1.31}$$

$$\begin{aligned}\sum_{i=1}^{n}\omega_i^2 &= \sum_{i=1}^{n}\left(\frac{X_i - \overline{X}}{\sum_{i=1}^{n}(X_i - \overline{X})^2}\right)^2 = \frac{\sum_{i=1}^{n}(X_i - \overline{X})^2}{\left(\sum_{i=1}^{n}(X_i - \overline{X})^2\right)^2} \\ &= \frac{1}{\sum_{i=1}^{n}(X_i - \overline{X})^2}.\end{aligned} \tag{1.32}$$

The first equality of equation (1.31) comes from equation (1.30).

From now on, we focus only on $\hat{\beta}_2$, because usually β_2 is more important than β_1 in the regression model (1.21). In order to obtain the properties of the least squares estimator $\hat{\beta}_2$, we rewrite equation (1.29) as:

$$\begin{aligned}\hat{\beta}_2 &= \sum_{i=1}^{n}\omega_i Y_i = \sum_{i=1}^{n}\omega_i(\beta_1 + \beta_2 X_i + u_i) \\ &= \beta_1\sum_{i=1}^{n}\omega_i + \beta_2\sum_{i=1}^{n}\omega_i X_i + \sum_{i=1}^{n}\omega_i u_i \\ &= \beta_2 + \sum_{i=1}^{n}\omega_i u_i.\end{aligned} \tag{1.33}$$

In the fourth equality of (1.33), equations (1.30) and (1.31) are utilized.

Mean and Variance of $\hat{\beta}_2$: $u_1, u_2, \cdots, u_n$ are assumed to be mutually independently and identically distributed with mean zero and variance σ^2, but they are not necessarily normal. Remember that we do not need normality assumption to obtain mean and variance but the normality assumption is required to test a hypothesis.

From equation (1.33), the expectation of $\hat{\beta}_2$ is derived as follows:

$$\begin{aligned}\mathrm{E}(\hat{\beta}_2) &= \mathrm{E}(\beta_2 + \sum_{i=1}^{n}\omega_i u_i) = \beta_2 + \mathrm{E}(\sum_{i=1}^{n}\omega_i u_i) \\ &= \beta_2 + \sum_{i=1}^{n}\omega_i \mathrm{E}(u_i) = \beta_2.\end{aligned} \tag{1.34}$$

It is shown from (1.34) that the ordinary least squares estimator $\hat{\beta}_2$ is an unbiased estimator of β_2.

From (1.33), the variance of $\hat{\beta}_2$ is computed as:

$$\begin{aligned} \mathrm{V}(\hat{\beta}_2) &= \mathrm{V}(\beta_2 + \sum_{i=1}^{n} \omega_i u_i) = \mathrm{V}(\sum_{i=1}^{n} \omega_i u_i) = \sum_{i=1}^{n} \mathrm{V}(\omega_i u_i) = \sum_{i=1}^{n} \omega_i^2 \mathrm{V}(u_i) \\ &= \sigma^2 \sum_{i=1}^{n} \omega_i^2 = \frac{\sigma^2}{\sum_{i=1}^{n}(X_i - \overline{X})^2}. \end{aligned} \tag{1.35}$$

From Theorem on p.15, the second and the fourth equalities hold. The third equality holds because $u_1, u_2, \cdots, u_n$ are mutually independent (see the theorem on p.20). The last equality comes from equation (1.32).

Thus, $\mathrm{E}(\hat{\beta}_2)$ and $\mathrm{V}(\hat{\beta}_2)$ are given by (1.34) and (1.35).

Gauss-Markov Theorem: It has been discussed above that $\hat{\beta}_2$ is represented as (1.29), which implies that $\hat{\beta}_2$ is a linear estimator, i.e., linear in Y_i. In addition, (1.34) indicates that $\hat{\beta}_2$ is an unbiased estimator. Therefore, summarizing these two facts, it is shown that $\hat{\beta}_2$ is a **linear unbiased estimator**. Furthermore, here we show that $\hat{\beta}_2$ has minimum variance within a class of the linear unbiased estimators.

Consider the alternative linear unbiased estimator $\widetilde{\beta}_2$ as follows:

$$\widetilde{\beta}_2 = \sum_{i=1}^{n} c_i Y_i = \sum_{i=1}^{n} (\omega_i + d_i) Y_i,$$

where $c_i = \omega_i + d_i$ is defined and d_i is nonstochastic. Then, $\widetilde{\beta}_2$ is transformed into:

$$\begin{aligned} \widetilde{\beta}_2 &= \sum_{i=1}^{n} c_i Y_i = \sum_{i=1}^{n} (\omega_i + d_i)(\beta_1 + \beta_2 X_i + u_i) \\ &= \beta_1 \sum_{i=1}^{n} \omega_i + \beta_2 \sum_{i=1}^{n} \omega_i X_i + \sum_{i=1}^{n} \omega_i u_i + \beta_1 \sum_{i=1}^{n} d_i + \beta_2 \sum_{i=1}^{n} d_i X_i + \sum_{i=1}^{n} d_i u_i \\ &= \beta_2 + \beta_1 \sum_{i=1}^{n} d_i + \beta_2 \sum_{i=1}^{n} d_i X_i + \sum_{i=1}^{n} \omega_i u_i + \sum_{i=1}^{n} d_i u_i. \end{aligned}$$

Equations (1.30) and (1.31) are used in the forth equality. Taking the expectation on both sides of the above equation, we obtain:

$$\begin{aligned} \mathrm{E}(\widetilde{\beta}_2) &= \beta_2 + \beta_1 \sum_{i=1}^{n} d_i + \beta_2 \sum_{i=1}^{n} d_i X_i + \sum_{i=1}^{n} \omega_i \mathrm{E}(u_i) + \sum_{i=1}^{n} d_i \mathrm{E}(u_i) \\ &= \beta_2 + \beta_1 \sum_{i=1}^{n} d_i + \beta_2 \sum_{i=1}^{n} d_i X_i. \end{aligned}$$

Note that d_i is not a random variable and that $\mathrm{E}(u_i) = 0$. Since $\widetilde{\beta}_2$ is assumed to be unbiased, we need the following conditions:

$$\sum_{i=1}^{n} d_i = 0, \qquad \sum_{i=1}^{n} d_i X_i = 0.$$

When these conditions hold, we can rewrite $\widetilde{\beta}_2$ as:

$$\widetilde{\beta}_2 = \beta_2 + \sum_{i=1}^{n} (\omega_i + d_i) u_i.$$

The variance of $\widetilde{\beta}_2$ is derived as:

$$\begin{aligned}
\mathrm{V}(\widetilde{\beta}_2) &= \mathrm{V}\Big(\beta_2 + \sum_{i=1}^{n} (\omega_i + d_i) u_i\Big) = \mathrm{V}\Big(\sum_{i=1}^{n} (\omega_i + d_i) u_i\Big) = \sum_{i=1}^{n} \mathrm{V}\Big((\omega_i + d_i) u_i\Big) \\
&= \sum_{i=1}^{n} (\omega_i + d_i)^2 \mathrm{V}(u_i) = \sigma^2 \Big(\sum_{i=1}^{n} \omega_i^2 + 2 \sum_{i=1}^{n} \omega_i d_i + \sum_{i=1}^{n} d_i^2\Big) \\
&= \sigma^2 \Big(\sum_{i=1}^{n} \omega_i^2 + \sum_{i=1}^{n} d_i^2\Big).
\end{aligned}$$

From unbiasedness of $\widetilde{\beta}_2$, using $\sum_{i=1}^{n} d_i = 0$ and $\sum_{i=1}^{n} d_i X_i = 0$, we obtain:

$$\sum_{i=1}^{n} \omega_i d_i = \frac{\sum_{i=1}^{n} (X_i - \overline{X}) d_i}{\sum_{i=1}^{n} (X_i - \overline{X})^2} = \frac{\sum_{i=1}^{n} X_i d_i - \overline{X} \sum_{i=1}^{n} d_i}{\sum_{i=1}^{n} (X_i - \overline{X})^2} = 0,$$

which is utilized to obtain the variance of $\widetilde{\beta}_2$ in the third line of the above equation. From (1.35), the variance of $\hat{\beta}_2$ is given by: $\mathrm{V}(\hat{\beta}_2) = \sigma^2 \sum_{i=1}^{n} \omega_i^2$. Therefore, we have:

$$\mathrm{V}(\widetilde{\beta}_2) \geq \mathrm{V}(\hat{\beta}_2),$$

because of $\sum_{i=1}^{n} d_i^2 \geq 0$. When $\sum_{i=1}^{n} d_i^2 = 0$, i.e., when $d_1 = d_2 = \cdots = d_n = 0$, we have the equality: $\mathrm{V}(\widetilde{\beta}_2) = \mathrm{V}(\hat{\beta}_2)$. Thus, in the case of $d_1 = d_2 = \cdots = d_n = 0$, $\hat{\beta}_2$ is equivalent to $\widetilde{\beta}_2$.

As shown above, the least squares estimator $\hat{\beta}_2$ gives us the **linear unbiased minimum variance estimator**, or equivalently the **best linear unbiased estimator (BLUE)**, which is called the **Gauss-Markov theorem**.

Asymptotic Properties of $\hat{\beta}_2$: We assume that as n goes to infinity we have the following:

$$\frac{1}{n} \sum_{i=1}^{n} (X_i - \overline{X})^2 \longrightarrow M < \infty,$$

where M is a constant value. From (1.32), we obtain:

$$n\sum_{i=1}^{n}\omega_i^2 = \frac{1}{(1/n)\sum_{i=1}^{n}(X_i-\overline{X})} \longrightarrow \frac{1}{M}.$$

Note that $f(x_n) \longrightarrow f(m)$ when $x_n \longrightarrow m$, where m is a constant value and $f(\cdot)$ is a function.

Here, we show both consistency of $\hat{\beta}_2$ and asymptotic normality of $\sqrt{n}(\hat{\beta}_2-\beta_2)$. First, we prove that $\hat{\beta}_2$ is a consistent estimator of β_2. As in (1.10), Chebyshev's inequality is given by:

$$P(|X-\mu|>\epsilon) \leq \frac{\sigma^2}{\epsilon^2},$$

where $\mu = \mathrm{E}(X)$ and $\sigma^2 = \mathrm{V}(X)$. Here, we replace X, $\mathrm{E}(X)$ and $\mathrm{V}(X)$ by $\hat{\beta}_2$,

$$\mathrm{E}(\hat{\beta}_2)=\beta_2, \quad \mathrm{V}(\hat{\beta}_2)=\sigma^2\sum_{i=1}^{n}\omega_i^2 = \frac{\sigma^2}{\sum_{i=1}^{n}(X_i-\overline{X})},$$

respectively. Then, when $n \longrightarrow \infty$, we obtain the following result:

$$P(|\hat{\beta}_2-\beta_2|>\epsilon) \leq \frac{\sigma^2\sum_{i=1}^{n}\omega_i^2}{\epsilon^2} = \frac{\sigma^2}{\epsilon^2\sum_{i=1}^{n}(X_i-\overline{X})} \longrightarrow 0,$$

where $\sum_{i=1}^{n}\omega_i^2 \longrightarrow 0$ because $n\sum_{i=1}^{n}\omega_i^2 \longrightarrow 1/M$ from the assumption. Thus, we obtain the result that $\hat{\beta}_2 \longrightarrow \beta_2$ as $n \longrightarrow \infty$. Therefore, we can conclude that $\hat{\beta}_2$ is a consistent estimator of β_2.

Next, we want to show that $\sqrt{n}(\hat{\beta}_2-\beta_2)$ is asymptotically normal. Noting that $\hat{\beta}_2 = \beta_2 + \sum_{i=1}^{n}\omega_i u_i$ as in (1.33) from Corollary 2 on p.35 (central limit theorem), asymptotic normality is shown as follows:

$$\frac{\sum_{i=1}^{n}\omega_i u_i - \mathrm{E}(\sum_{i=1}^{n}\omega_i u_i)}{\sqrt{\mathrm{V}(\sum_{i=1}^{n}\omega_i u_i)}} = \frac{\sum_{i=1}^{n}\omega_i u_i}{\sigma\sqrt{\sum_{i=1}^{n}\omega_i^2}} = \frac{\hat{\beta}_2-\beta_2}{\sigma/\sqrt{\sum_{i=1}^{n}(X_i-\overline{X})^2}} \longrightarrow N(0,1),$$

where $\mathrm{E}(\sum_{i=1}^{n}\omega_i u_i)=0$, $\mathrm{V}(\sum_{i=1}^{n}\omega_i u_i)=\sigma^2\sum_{i=1}^{n}\omega_i^2$ and $\sum_{i=1}^{n}\omega_i u_i = \hat{\beta}_2-\beta_2$ are substituted in the second equality. Moreover, we can rewrite as follows:

$$\frac{\hat{\beta}_2-\beta_2}{\sigma/\sqrt{\sum_{i=1}^{n}(X_i-\overline{X})^2}} = \frac{\sqrt{n}(\hat{\beta}_2-\beta_2)}{\sigma/\sqrt{(1/n)\sum_{i=1}^{n}(X_i-\overline{X})^2}} \longrightarrow \frac{\sqrt{n}(\hat{\beta}_2-\beta_2)}{\sigma/\sqrt{M}} \longrightarrow N(0,1),$$

or equivalently,

$$\sqrt{n}(\hat{\beta}_2-\beta_2) \longrightarrow N(0,\frac{\sigma^2}{M}).$$

Thus, asymptotic normality of $\sqrt{n}(\hat{\beta}_2-\beta_2)$ is shown.

Finally, replacing σ^2 by its consistent estimator s^2, it is known as follows:

$$\frac{\hat{\beta}_2 - \beta_2}{s\sqrt{\sum_{i=1}^n (X_i - \overline{X})^2}} \longrightarrow N(0,1), \tag{1.36}$$

where s^2 is defined as:

$$s^2 = \frac{1}{n-2}\sum_{i=1}^n e_i^2 = \frac{1}{n-2}\sum_{i=1}^n (Y_i - \hat{\beta}_1 - \hat{\beta}_2 X_i)^2, \tag{1.37}$$

which is a consistent and unbiased estimator of σ^2.

Thus, using (1.36), in large sample we can construct the confidence interval discussed in Section 1.7.6 and test the hypothesis discussed in Section 1.8.

Exact Distribution of $\hat{\beta}_2$: We have shown asymptotic normality of $\sqrt{n}(\hat{\beta}_2 - \beta_2)$, which is one of the large sample properties. Now, we discuss the small sample properties of $\hat{\beta}_2$. In order to obtain the distribution of $\hat{\beta}_2$ in small sample, the distribution of the error term has to be assumed. Therefore, the extra assumption is that $u_i \sim N(0, \sigma^2)$. Writing equation (1.33), again, $\hat{\beta}_2$ is represented as:

$$\hat{\beta}_2 = \beta_2 + \sum_{i=1}^n \omega_i u_i.$$

First, we obtain the distribution of the second term in the above equation. From Theorem on p.29, $\sum_{i=1}^n \omega_i u_i$ is distributed as:

$$\sum_{i=1}^n \omega_i u_i \sim N(0, \sigma^2 \sum_{i=1}^n \omega_i^2),$$

which is easily shown using the moment-generating function. Therefore, from Example 1.9 on p.23, $\hat{\beta}_2$ is distributed as:

$$\hat{\beta}_2 = \beta_2 + \sum_{i=1}^n \omega_i u_i \sim N(\beta_2, \sigma^2 \sum_{i=1}^n \omega_i^2),$$

or equivalently,

$$\frac{\hat{\beta}_2 - \beta_2}{\sigma\sqrt{\sum_{i=1}^n \omega_i^2}} = \frac{\hat{\beta}_2 - \beta_2}{\sigma/\sqrt{\sum_{i=1}^n (X_i - \overline{X})^2}} \sim N(0,1),$$

for any n.

Moreover, replacing σ^2 by its estimator s^2 defined in (1.37), it is known that we have:

$$\frac{\hat{\beta}_2 - \beta_2}{s/\sqrt{\sum_{i=1}^n (X_i - \overline{X})^2}} \sim t(n-2),$$

where $t(n-2)$ denotes t distribution with $n-2$ degrees of freedom. See Section 2.2.10 for derivation of the t distribution. Thus, under normality assumption on the error term u_i, the $t(n-2)$ distribution is used for the confidence interval and the testing hypothesis in small sample.

1.9.4 Multiple Regression Model

In Sections 1.9.1 – 1.9.3, only one independent variable, i.e., X_i, is taken into the regression model. In this section, we extend it to more independent variables, which is called the **multiple regression**. We consider the following regression model:

$$\begin{aligned} Y_i &= \beta_1 X_{i,1} + \beta_2 X_{i,2} + \cdots + \beta_k X_{i,k} + u_i \\ &= X_i \beta + u_i, \end{aligned}$$

for $i = 1, 2, \cdots, n$, where X_i and β denote a $1 \times k$ vector of the independent variables and a $k \times 1$ vector of the unknown parameters to be estimated, which are represented as:

$$X_i = (X_{i,1}, X_{i,2}, \cdots, X_{i,k}), \qquad \beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{pmatrix}.$$

$X_{i,j}$ denotes the ith observation of the jth independent variable. The case of $k = 2$ and $X_{i,1} = 1$ for all i is exactly equivalent to (1.21). Therefore, the matrix form above is a generalization of (1.21). Writing all the equations for $i = 1, 2, \cdots, n$, we have:

$$\begin{aligned} Y_1 &= \beta_1 X_{1,1} + \beta_2 X_{1,2} + \cdots + \beta_k X_{1,k} + u_1, \\ Y_2 &= \beta_1 X_{2,1} + \beta_2 X_{2,2} + \cdots + \beta_k X_{2,k} + u_2, \\ &\vdots \\ Y_n &= \beta_1 X_{n,1} + \beta_2 X_{n,2} + \cdots + \beta_k X_{n,k} + u_n, \end{aligned}$$

which is rewritten as:

$$\begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix} = \begin{pmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,k} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n,1} & X_{n,2} & \cdots & X_{n,k} \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{pmatrix} + \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_k \end{pmatrix}.$$

Again, the above equation is compactly rewritten as:

$$Y = X\beta + u. \tag{1.38}$$

where Y, X and u are denoted by:

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix}, \qquad X = \begin{pmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,k} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n,1} & X_{n,2} & \cdots & X_{n,k} \end{pmatrix}, \qquad u = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_k \end{pmatrix}.$$

Utilizing the matrix form (1.38), we derive the ordinary least squares estimator of β, denoted by $\hat{\beta}$. In equation (1.38), replacing β by $\hat{\beta}$, we have the following equation:

$$Y = X\hat{\beta} + e,$$

where e denotes a $1 \times n$ vector of the residuals. The ith element of e is given by e_i. The sum of squared residuals is written as follows:

$$\begin{aligned} S(\hat{\beta}) &= \sum_{i=1}^{n} e_i^2 = e'e = (Y - X\hat{\beta})'(Y - X\hat{\beta}) = (Y' - \hat{\beta}'X')(Y - X\hat{\beta}) \\ &= Y'Y - Y'X\hat{\beta} - \hat{\beta}'X'Y + \hat{\beta}'X'X\hat{\beta} = Y'Y - 2Y'X\hat{\beta} + \hat{\beta}'X'X\hat{\beta}. \end{aligned}$$

See Appendix 1.5 for the transpose in the fourth equality. In the last equality, note that $\hat{\beta}'X'Y = Y'X\hat{\beta}$ because both are scalars. To minimize $S(\hat{\beta})$ with respect to $\hat{\beta}$, we set the first derivative of $S(\hat{\beta})$ equal to zero, i.e.,

$$\frac{\partial S(\hat{\beta})}{\partial \hat{\beta}} = -2X'Y + 2X'X\hat{\beta} = 0.$$

See Appendix 1.5 for the derivatives of matrices. Solving the equation above with respect to $\hat{\beta}$, the ordinary least squares estimator of β is given by:

$$\hat{\beta} = (X'X)^{-1}X'Y. \tag{1.39}$$

See Appendix 1.5 for the inverse of the matrix. Thus, the ordinary least squares estimator is derived in the matrix form.

Now, in order to obtain the properties of $\hat{\beta}$ such as mean, variance, distribution and so on, (1.39) is rewritten as follows:

$$\begin{aligned} \hat{\beta} &= (X'X)^{-1}X'Y = (X'X)^{-1}X'(X\beta + u) = (X'X)^{-1}X'X\beta + (X'X)^{-1}X'u \\ &= \beta + (X'X)^{-1}X'u. \end{aligned} \tag{1.40}$$

Taking the expectation on both sides of equation (1.40), we have the following:

$$\mathrm{E}(\hat{\beta}) = \mathrm{E}(\beta + (X'X)^{-1}X'u) = \beta + (X'X)^{-1}X'\mathrm{E}(u) = \beta,$$

because of $\mathrm{E}(u) = 0$ by the assumption of the error term u_i. Thus, unbiasedness of $\hat{\beta}$ is shown.

The variance of $\hat{\beta}$ is obtained as:

$$\begin{aligned} \mathrm{V}(\hat{\beta}) &= \mathrm{E}((\hat{\beta} - \beta)(\hat{\beta} - \beta)') = \mathrm{E}\Big((X'X)^{-1}X'u((X'X)^{-1}X'u)'\Big) \\ &= \mathrm{E}((X'X)^{-1}X'uu'X(X'X)^{-1}) = (X'X)^{-1}X'\mathrm{E}(uu')X(X'X)^{-1} \\ &= \sigma^2(X'X)^{-1}X'X(X'X)^{-1} = \sigma^2(X'X)^{-1}. \end{aligned}$$

The first equality is the definition of variance in the case of vector. In the fifth equality, $\mathrm{E}(uu') = \sigma^2 I_n$ is used, which implies that $\mathrm{E}(u_i^2) = \sigma^2$ for all i and $\mathrm{E}(u_i u_j) = 0$ for

$i \neq j$. Remember that $u_1, u_2, \cdots, u_n$ are assumed to be mutually independently and identically distributed with mean zero and variance σ^2.

Under normality assumption on the error term u, it is known that the distribution of $\hat{\beta}$ is given by:

$$\hat{\beta} \sim N(\beta, \sigma^2(X'X)^{-1}).$$

Taking the jth element of $\hat{\beta}$, its distribution is given by:

$$\hat{\beta}_j \sim N(\beta_j, \sigma^2 a_{jj}), \qquad \text{i.e.,} \qquad \frac{\hat{\beta}_j - \beta_j}{\sigma\sqrt{a_{jj}}} \sim N(0,1),$$

where a_{jj} denotes the jth diagonal element of $(X'X)^{-1}$.

Replacing σ^2 by its estimator s^2, we have the following t distribution:

$$\frac{\hat{\beta}_j - \beta_j}{s\sqrt{a_{jj}}} \sim t(n-k),$$

where $t(n-k)$ denotes the t distribution with $n-k$ degrees of freedom. s^2 is taken as follows:

$$s^2 = \frac{1}{n-k}\sum_{i=1}^{n} e_i^2 = \frac{1}{n-k}e'e = \frac{1}{n-k}(Y - X\hat{\beta})'(Y - X\hat{\beta}),$$

which leads to an unbiased estimator of σ^2.

Using the central limit theorem, without normality assumption we can show that as $n \longrightarrow \infty$, under the condition of $(1/n)X'X \longrightarrow M$ we have the following result:

$$\frac{\hat{\beta}_j - \beta_j}{s\sqrt{a_{jj}}} \longrightarrow N(0,1),$$

where M denotes a $k \times k$ constant matrix.

Thus, we can construct the confidence interval and the testing procedure, using the t distribution under the normality assumption or the normal distribution without the normality assumption.

Appendix 1.1: Integration by Substitution

Univariate Case: For a function of x, $f(x)$, we perform integration by substitution, using $x = \psi(y)$. Then, it is easy to obtain the following formula:

$$\int f(x)\,\mathrm{d}x = \int \psi'(y) f(\psi(y))\,\mathrm{d}y,$$

which formula is called the **integration by substitution**.

Proof:
Let $F(x)$ be the integration of $f(x)$, i.e.,

$$F(x) = \int_{-\infty}^{x} f(t)\,\mathrm{d}t,$$

which implies that $F'(x) = f(x)$.

Differentiating $F(x) = F(\psi(y))$ with respect to y, we have:

$$f(x) \equiv \frac{\mathrm{d}F(\psi(y))}{\mathrm{d}y} = \frac{\mathrm{d}F(x)}{\mathrm{d}x}\frac{\mathrm{d}x}{\mathrm{d}y} = f(x)\psi'(y) = f(\psi(y))\psi'(y).$$

Bivariate Case: For $f(x, y)$, define $x = \psi_1(u, v)$ and $y = \psi_2(u, v)$.

$$\iint f(x, y)\,\mathrm{d}x\,\mathrm{d}y = \iint J f(\psi_1(u, v), \psi_2(u, v))\,\mathrm{d}u\,\mathrm{d}v,$$

where J is called the **Jacobian**, which represents the following determinant:

$$J = \begin{vmatrix} \dfrac{\partial x}{\partial u} & \dfrac{\partial x}{\partial v} \\ \dfrac{\partial y}{\partial u} & \dfrac{\partial y}{\partial v} \end{vmatrix} = \frac{\partial x}{\partial u}\frac{\partial y}{\partial v} - \frac{\partial x}{\partial v}\frac{\partial y}{\partial u}.$$

Appendix 1.2: Integration by Parts

Let $h(x)$ and $g(x)$ be functions of x. Then, we have the following formula:

$$\int h(x)g'(x)\,\mathrm{d}x = h(x)g(x) - \int h'(x)g(x)\,\mathrm{d}x,$$

which is called the **integration by parts**.

Proof:
Consider the derivative of $f(x)g(x)$ with respect to x, i.e.,

$$\big(f(x)g(x)\big)' = f'(x)g(x) + f(x)g'(x).$$

Integrating the above equation on both sides, we have:

$$\int \big(f(x)g(x)\big)'\,\mathrm{d}x = \int f'(x)g(x)\,\mathrm{d}x + \int f(x)g'(x)\,\mathrm{d}x.$$

Therefore, we obtain:

$$f(x)g(x) = \int f'(x)g(x)\,\mathrm{d}x + \int f(x)g'(x)\,\mathrm{d}x.$$

Thus, the following result is derived.

$$\int f(x)g'(x)\,\mathrm{d}x = f(x)g(x) - \int f'(x)g(x)\,\mathrm{d}x.$$

When we want to integrate $f(x)g'(x)$ within the range between a and b for $a < b$, the above formula is modified as:

$$\int_a^b f(x)g'(x)\,\mathrm{d}x = \Big[f(x)g(x)\Big]_a^b - \int_a^b f'(x)g(x)\,\mathrm{d}x.$$

Appendix 1.3: Taylor Series Expansion

Consider approximating $f(x)$ around $x = x_0$ by the **Taylor series expansion**.. Then, $f(x)$ is approximated as follows:

$$\begin{aligned} f(x) &= f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2!}f''(x_0)(x - x_0)^2 + \frac{1}{3!}f'''(x_0)(x - x_0)^3 + \cdots \\ &= \sum_{n=0}^{\infty} \frac{1}{n!} f^{(n)}(x_0)(x - x_0)^n, \end{aligned}$$

where $f^{(n)}(x_0)$ denotes the nth derivative of $f(x)$ evaluated at $x = x_0$. Note that $f^{(0)}(x_0) = f(x_0)$ and $0! = 1$.

In addition, the following approximation is called the ***k*th order Taylor series expansion**:

$$f(x) \approx \sum_{n=0}^{k} \frac{1}{n!} f^{(n)}(x_0)(x - x_0)^n.$$

Appendix 1.4: Cramer-Rao Inequality

As seen in (1.13) and (1.14), the Cramer-Rao inequality is given by:

$$\mathrm{V}(\hat{\theta}_n) \geq \frac{\sigma^2(\theta)}{n},$$

where

$$\sigma^2(\theta) = \frac{1}{\mathrm{E}\left(\left(\dfrac{\partial \log f(X;\theta)}{\partial \theta}\right)^2\right)} = \frac{1}{\mathrm{V}\left(\left(\dfrac{\partial \log f(X;\theta)}{\partial \theta}\right)\right)} = -\frac{1}{\mathrm{E}\left(\dfrac{\partial^2 \log f(X;\theta)}{\partial \theta^2}\right)}.$$

Proof:
We prove the above inequality and the equalities in $\sigma^2(\theta)$. The likelihood function $l(\theta; x) = l(\theta; x_1, x_2, \cdots, x_n)$ is a joint density of $X_1, X_2, \cdots, X_n$. Therefore, the integration of $l(\theta; x_1, x_2, \cdots, x_n)$ with respect to $x_1, x_2, \cdots, x_n$ is equal to one. See Section 1.7.5 for the likelihood function. That is, we have the following equation:

$$1 = \int l(\theta; x)\, \mathrm{d}x, \tag{1.41}$$

where the likelihood function $l(\theta; x)$ is given by $l(\theta; x) = \prod_{i=1}^{n} f(x_i; \theta)$ and $\int \cdots \mathrm{d}x$ implies n-tuple integral.

Differentiating both sides of equation (1.41) with respect to θ, we obtain the following equation:

$$\begin{aligned} 0 &= \int \frac{\partial l(\theta; x)}{\partial \theta}\, \mathrm{d}x = \int \frac{1}{l(\theta; x)} \frac{\partial l(\theta; x)}{\partial \theta} l(\theta; x)\, \mathrm{d}x \\ &= \int \frac{\partial \log l(\theta; x)}{\partial \theta} l(\theta; x)\, \mathrm{d}x = \mathrm{E}\Big(\frac{\partial \log l(\theta; X)}{\partial \theta}\Big), \end{aligned} \tag{1.42}$$

which implies that the expectation of $\partial \log l(\theta; X)/\partial\theta$ is equal to zero. In the third equality, note that $\mathrm{d}\log x/\,\mathrm{d}x = 1/x$.

Now, let $\hat{\theta}_n$ be an estimator of θ. The definition of the mathematical expectation of the estimator $\hat{\theta}_n$ is represented as:

$$\mathrm{E}(\hat{\theta}_n) = \int \hat{\theta}_n l(\theta; x)\, \mathrm{d}x. \tag{1.43}$$

Differentiating equation (1.43) with respect to θ on both sides, we can rewrite as follows:

$$\begin{aligned} \frac{\partial \mathrm{E}(\hat{\theta}_n)}{\partial \theta} &= \int \hat{\theta}_n \frac{\partial l(\theta; x)}{\partial \theta}\, \mathrm{d}x = \int \hat{\theta}_n \frac{\partial \log l(\theta; x)}{\partial \theta} l(\theta; x)\, \mathrm{d}x \\ &= \int \Big(\hat{\theta}_n - \mathrm{E}(\hat{\theta}_n)\Big)\Big(\frac{\partial \log l(\theta; x)}{\partial \theta} - \mathrm{E}(\frac{\partial \log l(\theta; x)}{\partial \theta})\Big) l(\theta; x)\, \mathrm{d}x \\ &= \mathrm{Cov}\Big(\hat{\theta}_n, \frac{\partial \log l(\theta; X)}{\partial \theta}\Big). \end{aligned} \tag{1.44}$$

In the second equality, $\mathrm{d}\log x/\,\mathrm{d}x = 1/x$ is utilized. The third equality holds because of $\mathrm{E}(\partial \log l(\theta; X)/\partial\theta) = 0$ from equation (1.42).

For simplicity of discussion, suppose that θ is a scalar. Taking the square on both sides of equation (1.44), we obtain the following expression:

$$\begin{aligned} \Big(\frac{\partial \mathrm{E}(\hat{\theta}_n)}{\partial \theta}\Big)^2 &= \left(\mathrm{Cov}\Big(\hat{\theta}_n, \frac{\partial \log l(\theta; X)}{\partial \theta}\Big)\right)^2 = \rho^2 \mathrm{V}(\hat{\theta}_n) \mathrm{V}\Big(\frac{\partial \log l(\theta; X)}{\partial \theta}\Big) \\ &\le \mathrm{V}(\hat{\theta}_n) \mathrm{V}\left(\frac{\partial \log l(\theta; X)}{\partial \theta}\right), \end{aligned} \tag{1.45}$$

where ρ denotes the correlation coefficient between $\hat{\theta}_n$ and $\partial \log l(\theta; X)/\partial\theta$. Note that we have the definition of ρ is given by:

$$\rho = \frac{\mathrm{Cov}\Big(\hat{\theta}_n, \dfrac{\partial \log l(\theta; X)}{\partial\theta}\Big)}{\sqrt{\mathrm{V}(\hat{\theta}_n)}\sqrt{\mathrm{V}\Big(\dfrac{\partial \log l(\theta; X)}{\partial\theta}\Big)}}.$$

Moreover, we have $-1 \leq \rho \leq 1$ (i.e., $\rho^2 \leq 1$). Then, the inequality (1.45) is obtained, which is rewritten as:

$$\mathrm{V}(\hat{\theta}_n) \geq \frac{\Big(\dfrac{\partial \mathrm{E}(\hat{\theta}_n)}{\partial\theta}\Big)^2}{\mathrm{V}\Big(\dfrac{\partial \log l(\theta; X)}{\partial\theta}\Big)}. \tag{1.46}$$

When $\mathrm{E}(\hat{\theta}_n) = \theta$, i.e., when $\hat{\theta}_n$ is an unbiased estimator of θ, the numerator in the right-hand side of equation (1.46) is equal to one. Therefore, we have the following result:

$$\mathrm{V}(\hat{\theta}_n) \geq \frac{1}{\mathrm{V}\Big(\dfrac{\partial \log l(\theta; X)}{\partial\theta}\Big)} = \frac{1}{\mathrm{E}\left(\Big(\dfrac{\partial \log l(\theta; X)}{\partial\theta}\Big)^2\right)}.$$

Note that we have $\mathrm{V}(\partial \log l(\theta; X)/\partial\theta) = \mathrm{E}((\partial \log l(\theta; X)/\partial\theta)^2)$ in the equality above, because of $\mathrm{E}(\partial \log l(\theta; X)/\partial\theta) = 0$.

Moreover, the denominator in the right-hand side of the above inequality is rewritten as follows:

$$\begin{aligned}\mathrm{E}\left(\Big(\frac{\partial \log l(\theta; X)}{\partial\theta}\Big)^2\right) &= \mathrm{E}\left(\Big(\sum_{i=1}^{n} \frac{\partial \log f(X_i; \theta)}{\partial\theta}\Big)^2\right) = \sum_{i=1}^{n} \mathrm{E}\left(\Big(\frac{\partial \log f(X_i; \theta)}{\partial\theta}\Big)^2\right) \\ &= n\mathrm{E}\left(\Big(\frac{\partial \log f(X; \theta)}{\partial\theta}\Big)^2\right) = n\int_{-\infty}^{\infty}\Big(\frac{\partial \log f(x; \theta)}{\partial\theta}\Big)^2 f(x; \theta)\,\mathrm{d}x.\end{aligned}$$

In the first equality, $\log l(\theta; X) = \sum_{i=1}^{n} \log f(X_i; \theta)$ is utilized. Since X_i, $i = 1, 2, \cdots, n$, are mutually independent, the second equality holds. The third equality holds because $X_1, X_2, \cdots, X_n$ are identically distributed.

Therefore, we obtain the following inequality:

$$\mathrm{V}(\hat{\theta}_n) \geq \frac{1}{\mathrm{E}\left(\Big(\dfrac{\partial \log l(\theta; X)}{\partial\theta}\Big)^2\right)} = \frac{1}{n\mathrm{E}\left(\Big(\dfrac{\partial \log f(X; \theta)}{\partial\theta}\Big)^2\right)} = \frac{\sigma^2(\theta)}{n},$$

which is equivalent to (1.13).

Next, we prove the equalities in (1.14), i.e.,

$$-\mathrm{E}\Big(\frac{\partial^2 \log f(X; \theta)}{\partial\theta^2}\Big) = \mathrm{E}\left(\Big(\frac{\partial \log f(X; \theta)}{\partial\theta}\Big)^2\right) = \mathrm{V}\Big(\frac{\partial \log f(X; \theta)}{\partial\theta}\Big).$$

Differentiating $\int f(x;\theta)\,\mathrm{d}x = 1$ with respect to θ, we obtain as follows:

$$\int \frac{\partial f(x;\theta)}{\partial \theta}\,\mathrm{d}x = 0.$$

We assume that the range of x does not depend on the parameter θ and that $\partial f(x;\theta)/\partial\theta$ exists. The above equation is rewritten as:

$$\int \frac{\partial \log f(x;\theta)}{\partial \theta} f(x;\theta)\,\mathrm{d}x = 0, \tag{1.47}$$

or equivalently,

$$\mathrm{E}\Big(\frac{\partial \log f(X;\theta)}{\partial \theta}\Big) = 0. \tag{1.48}$$

Again, differentiating equation (1.47) with respect to θ,

$$\int \frac{\partial^2 \log f(x;\theta)}{\partial \theta^2} f(x;\theta)\,\mathrm{d}x + \int \frac{\partial \log f(x;\theta)}{\partial \theta}\frac{\partial f(x;\theta)}{\partial \theta}\,\mathrm{d}x = 0,$$

i.e.,

$$\int \frac{\partial^2 \log f(x;\theta)}{\partial \theta^2} f(x;\theta)\,\mathrm{d}x + \int \Big(\frac{\partial \log f(x;\theta)}{\partial \theta}\Big)^2 f(x;\theta)\,\mathrm{d}x = 0,$$

i.e.,

$$\mathrm{E}\Big(\frac{\partial^2 \log f(x;\theta)}{\partial \theta^2}\Big) + \mathrm{E}\left(\Big(\frac{\partial \log f(x;\theta)}{\partial \theta}\Big)^2\right) = 0.$$

Thus, we obtain:

$$-\mathrm{E}\Big(\frac{\partial^2 \log f(x;\theta)}{\partial \theta^2}\Big) = \mathrm{E}\left(\Big(\frac{\partial \log f(x;\theta)}{\partial \theta}\Big)^2\right).$$

Moreover, from equation (1.48), the following equation is derived.

$$\mathrm{E}\left(\Big(\frac{\partial \log f(x;\theta)}{\partial \theta}\Big)^2\right) = \mathrm{V}\Big(\frac{\partial \log f(x;\theta)}{\partial \theta}\Big).$$

Therefore, we have:

$$-\mathrm{E}\Big(\frac{\partial^2 \log f(X;\theta)}{\partial \theta^2}\Big) = \mathrm{E}\left(\Big(\frac{\partial \log f(X;\theta)}{\partial \theta}\Big)^2\right) = \mathrm{V}\Big(\frac{\partial \log f(X;\theta)}{\partial \theta}\Big).$$

Thus, the Cramer-Rao inequality is derived as:

$$\mathrm{V}(\hat{\theta}_n) \geq \frac{\sigma^2(\theta)}{n},$$

where

$$\sigma^2(\theta) = \frac{1}{\mathrm{E}\left(\Big(\frac{\partial \log f(X;\theta)}{\partial \theta}\Big)^2\right)} = \frac{1}{\mathrm{V}\left(\Big(\frac{\partial \log f(X;\theta)}{\partial \theta}\Big)\right)} = -\frac{1}{\mathrm{E}\Big(\frac{\partial^2 \log f(X;\theta)}{\partial \theta^2}\Big)}.$$

Appendix 1.5: Some Formulas of Matrix Algebra

1. Let $A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1k} \\ a_{21} & a_{22} & \cdots & a_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{l1} & a_{l2} & \cdots & a_{lk} \end{pmatrix} = [a_{ij}]$, which is a $l \times k$ matrix, where a_{ij} denotes ith row and jth column of A. The **transpose** of A, denoted by A', is defined as:

   $$A' = \begin{pmatrix} a_{11} & a_{21} & \cdots & a_{l1} \\ a_{12} & a_{22} & \cdots & a_{l2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1k} & a_{2k} & \cdots & a_{lk} \end{pmatrix} = [a_{ji}],$$

   where the ith row of A' is the ith column of A.

2. $(Ax)' = x'A'$,

   where A and x are a $l \times k$ matrix and a $k \times 1$ vector, respectively.

3. $a' = a$,

   where a denotes a scalar.

4. $\dfrac{\partial a'x}{\partial x} = a$,

   where a and x are $k \times 1$ vectors.

5. $\dfrac{\partial x'Ax}{\partial x} = (A + A')x$,

   where A and x are a $k \times k$ matrix and a $k \times 1$ vector, respectively.

   Especially, when A is symmetric,

   $\dfrac{\partial x'Ax}{\partial x} = 2Ax$.

6. Let A and B be $k \times k$ matrices, and I_k be a $k \times k$ **identity matrix** (one in the diagonal elements and zero in the other elements).

   When $AB = I_k$, B is called the **inverse** of A, denoted by $B = A^{-1}$.

   That is, $AA^{-1} = A^{-1}A = I_k$.

7. Let A be a $k \times k$ matrix and x be a $k \times 1$ vector.

   If A is a **positive definite matrix**, for any x we have:

   $$x'Ax > 0.$$

   If A is a **positive semidefinite matrix**, for any x we have:

   $$x'Ax \geq 0.$$

If A is a **negative definite matrix**, for any x we have:

$$x'Ax < 0.$$

If A is a **negative semidefinite matrix**, for any x we have:

$$x'Ax \leq 0.$$

References

Greene, W.H., 1993, *Econometric Analysis* (Second Edition), Prentice Hall.

Greene, W.H., 1997, *Econometric Analysis* (Third Edition), Prentice-Hall.

Greene, W.H., 2000, *Econometric Analysis* (Fourth Edition), Prentice-Hall.

Hogg, R.V. and Craig, A.T., 1995, *Introduction to Mathematical Statistics* (Fifth Edition), Prentice Hall.

Judge, G., Hill, C., Griffiths, W. and Lee, T., 1980, *The Theory and Practice of Econometrics*, John Wiley & Sons.

Mood, A.M., Graybill, F.A. and Boes, D.C., 1974, *Introduction to the Theory of Statistics* (Third Edition), McGraw-Hill.

Stuart, A. and Ord, J.K., 1991, *Kendall's Advanced Theory of Statistics, Vol.2* (Fifth Edition), Edward Arnold.

Stuart, A. and Ord, J.K., 1994, *Kendall's Advanced Theory of Statistics, Vol.1* (Sixth Edition), Edward Arnold.

Part I

Monte Carlo Statistical Methods

Chapter 2

Random Number Generation I

In Chapter 1, we have discussed the probability function for discrete random variables and the density function for continuous random variables. In this chapter, we introduce as many distributions as possible and discuss how to generate random draws, where the source code written by Fortran 77 is also shown for each random number generator. Note that a lot of distribution functions are introduced in Kotz, Balakrishman and Johnson (2000a, 2000b, 2000c, 2000d, 2000e). The random draws discussed in this chapter are based on uniform random draws between zero and one.

2.1 Uniform Distribution: $U(0, 1)$

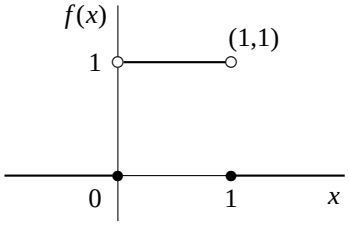
2.1.1 Properties of Uniform Distribution

The most heuristic and simplest distribution is uniform. The **uniform distribution** between zero and one is given by:

$$f(x) = \begin{cases} 1, & \text{for } 0 < x < 1, \\ 0, & \text{otherwise,} \end{cases}$$

which is displayed in Figure 2.1.

Figure 2.1: Uniform Distribution: $U(0, 1)$



As shown in Example 1.3 (p.6 in Section 1.2.1), the above function has the properties of the probability density function. Moreover, as in Example 1.6 (p.13), mean,

variance and the moment-generating function are given by:

$$\mathrm{E}(X) = \frac{1}{2}, \qquad \mathrm{V}(X) = \frac{1}{12}, \qquad \phi(\theta) = \frac{e^{\theta} - 1}{\theta}.$$

Use L'Hospital's theorem on 123 to derive $\mathrm{E}(X)$ and $\mathrm{V}(X)$ using $\phi(\theta)$.

In the next section, we introduce an idea of generating uniform random draws, which in turn yield the other random draws by the transformation of variables, the inverse transform algorithm and so on.

2.1.2 Uniform Random Number Generators

It is no exaggeration to say that all the random draws are based on a uniform random number. Once uniform random draws are generated, the various random draws such as exponential, normal, logistic, Bernoulli and other distributions are obtained by transforming the uniform random draws. Thus, it is important to consider how to generate a uniform random number. However, generally there is no way to generate exact uniform random draws. As shown in Ripley (1987) and Ross (1997), a deterministic sequence that appears at random is taken as a sequence of random numbers.

First, consider the following relation:

$$m = k - [k/n]n,$$

where k, m and n are integers. $[k/n]$ denotes the largest integer less than or equal to the argument. In Fortran 77, it is written as `m=k-int(k/n)*n`, where $0 \leq m < n$. m indicates the **remainder** when k is divided by n. n is called the **modulus**. We define the right hand side in the equation above as:

$$k - [k/n]n \equiv k \bmod n.$$

Then, using the modular arithmetic we can rewrite the above equation as follows:

$$m = k \bmod n,$$

which is represented by: `m=mod(k,n)` in Fortran 77 and `m=k%n` in C language.

A basic idea of the uniform random draw is as follows. Given x_{i-1}, x_i is generated by:

$$x_i = (ax_{i-1} + c) \bmod n,$$

where $0 \leq x_i < n$. a and c are positive integers, called the **multiplier** and the **increment**, respectively. The generator above have to be started by an initial value, which is called the **seed**. $u_i = x_i/n$ is regarded as a uniform random number between zero and one. This generator is called the **linear congruential generator**. Especially, when $c = 0$, the generator is called the **multiplicative linear congruential generator**. This method was proposed by Lehmer in 1948 (see Lehmer, 1951). If n, a and c are properly chosen, the period of the generator is n. However, when they are not chosen very

carefully, there may be a lot of serial correlation among the generated values. Therefore, the performance of the congruential generators depend heavily on the choice of (a, c). It is shown in Knuth (1981) that the generator above has a full period n if and only if:

(i) c is relatively prime to n, that is, c and n have no common divisors greater than one.

(ii) $1 = a \bmod g$,

for every prime factor g of n.

(iii) $1 = a \bmod 4$,

if n is a multiple of 4.

(i) implies that the greatest common divisor of c and n is unity. (ii) means that $a = g[a/g] + 1$. (iii) implies that $a = 4[a/4] + 1$ if $n/4$ is an integer.

There is a great amount of literature on uniform random number generation. See, for example, Fishman (1996), Gentle (1998), Kennedy and Gentle (1980), Law and Kelton (2000), Niederreiter (1992), Ripley (1987), Robert and Casella (1999), Rubinstein and Melamed (1998), Thompson (2000) and so on for the other congruential generators. However, we introduce only two uniform random number generators.

Wichmann and Hill (1982 and corrigendum, 1984) describe a combination of three congruential generators for 16-bit computers. The generator is given by:

$$x_i = 171x_{i-1} \bmod 30269,$$
$$y_i = 172y_{i-1} \bmod 30307,$$
$$z_i = 170z_{i-1} \bmod 30323,$$

and

$$u_i = \left(\frac{x_i}{30269} + \frac{y_i}{30307} + \frac{z_i}{30323}\right) \bmod 1.$$

We need to set three seeds, i.e., x_0, y_0 and z_0, for this random number generator. u_i is regarded as a uniform random draw within the interval between zero and one. The period is of the order of 10^{12} (more precisely the period is 6.95×10^{12}). The source code of this generator is given by `urnd16(ix,iy,iz,rn)`, where `ix`, `iy` and `iz` are seeds and `rn` represents the uniform random number between zero and one.

`urnd16(ix,iy,iz,rn)`

```
1:       subroutine urnd16(ix,iy,iz,rn)
2: c
3: c  Input:
4: c    ix, iy, iz:  Seeds
5: c  Output:
6: c    rn: Uniform Random Draw U(0,1)
7: c
```

```
 8:      1 ix=mod( 171*ix,30269 )
 9:        iy=mod( 172*iy,30307 )
10:        iz=mod( 170*iz,30323 )
11:        rn=ix/30269.+iy/30307.+iz/30323.
12:        rn=rn-int(rn)
13:        if( rn.le.0 ) go to 1
14:        return
15:        end
```

We exclude one in Line 12 and zero in Line 13 from `rn`. That is, $0 <$ `rn` < 1 is generated in `urnd16(ix,iy,iz,rn)`. Zero and one in the uniform random draw sometimes cause the complier errors in programming, when the other random draws are derived based on the transformation of the uniform random variable. De Matteis and Pagnutti (1993) examine the Wichmann-Hill generator with respect to the higher order autocorrelations in sequences, and conclude that the Wichmann-Hill generator performs well.

For 32-bit computers, L'Ecuyer (1988) proposed a combination of k congruential generators that have prime moduli n_j, such that all values of $(n_j - 1)/2$ are relatively prime, and with multipliers that yield full periods. Let the sequence from jth generator be $x_{j,1}$, $x_{j,2}$, $x_{j,3}$, $\cdots$. Consider the case where each individual generator j is a maximum-period multiplicative linear congruential generator with modulus n_j and multiplier a_j, i.e.,

$$x_{j,i} \equiv a_j x_{j,i-1} \bmod n_j.$$

Assuming that the first generator is a relatively good one and that n_1 is fairly large, we form the ith integer in the sequence as:

$$x_i = \sum_{j=1}^{k} (-1)^{j-1} x_{j,i} \bmod (n_1 - 1),$$

where the other moduli n_j, $j = 2, 3, \cdots, k$, do not need to be large. The normalization takes care of the possibility of zero occurring in this sequence:

$$u_i = \begin{cases} \dfrac{x_i}{n_1}, & \text{if } x_i > 0, \\ \dfrac{n_1 - 1}{n_1}, & \text{if } x_i = 0. \end{cases}$$

As for each individual generator j, note as follows. Define $q = [n/a]$ and $r \equiv n \bmod a$, i.e., n is decomposed as $n = aq + r$, where $r < a$. Therefore, for $0 < x < n$, we have:

$$\begin{aligned} ax \bmod n &= (ax - [x/q]n) \bmod n \\ &= \Big(ax - [x/q](aq + r)\Big) \bmod n \\ &= \Big(a(x - [x/q]q) - [x/q]r\Big) \bmod n \\ &= \Big(a(x \bmod q) - [x/q]r\Big) \bmod n. \end{aligned}$$

Practically, L'Ecuyer (1988) suggested combining two multiplicative congruential generators, where $k = 2$, (a_1, n_1, q_1, r_1) = (40014, 2147483563, 53668, 12211) and (a_2, n_2, q_2, r_2) = (40692, 2147483399, 52774, 3791) are chosen. Two seeds are required to implement the generator. The source code is shown in `urnd(ix,iy,rn)`, where `ix` and `iy` are inputs, i.e., seeds, and `rn` is an output, i.e., a uniform random number between zero and one.

`urnd(ix,iy,rn)`

```
1:        subroutine urnd(ix,iy,rn)
2:  c
3:  c   Input:
4:  c     ix, iy:  Seeds
5:  c   Output:
6:  c     rn: Uniform Random Draw U(0,1)
7:  c
8:       1 kx=ix/53668
9:         ix=40014*(ix-kx*53668)-kx*12211
10:        if(ix.lt.0) ix=ix+2147483563
11: c
12:        ky=iy/52774
13:        iy=40692*(iy-ky*52774)-ky*3791
14:        if(iy.lt.0) iy=iy+2147483399
15: c
16:        rn=ix-iy
17:        if( rn.lt.1.) rn=rn+2147483562
18:        rn=rn*4.656613e-10
19:        if( rn.le.0.) go to 1
20: c
21:        return
22:        end
```

The period of the generator proposed by L'Ecuyer (1988) is of the order of 10^{18} (more precisely 2.31×10^{18}), which is quite long and practically long enough.

L'Ecuyer (1988) presents the results of both theoretical and empirical tests, where the above generator performs well. Furthermore, L'Ecuyer (1988) gives an additional portable generator for 16-bit computers. Also, see L'Ecuyer(1990, 1998).

To improve the length of period, the above generator proposed by L'Ecuyer (1988) is combined with the shuffling method suggested by Bays and Durham (1976), and it is introduced as `ran2` in Press, Teukolsky, Vetterling and Flannery (1992a, 1992b). However, from relatively long period and simplicity of the source code, hereafter the subroutine `urnd(ix,iy,rn)` is utilized for the uniform random number generation method, and we will obtain various random draws based on the uniform random draws.

2.2 Transforming $U(0,1)$: Continuous Type

In this section, we focus on a continuous type of distributions, in which density functions are derived from the uniform distribution $U(0,1)$ by transformation of variables.

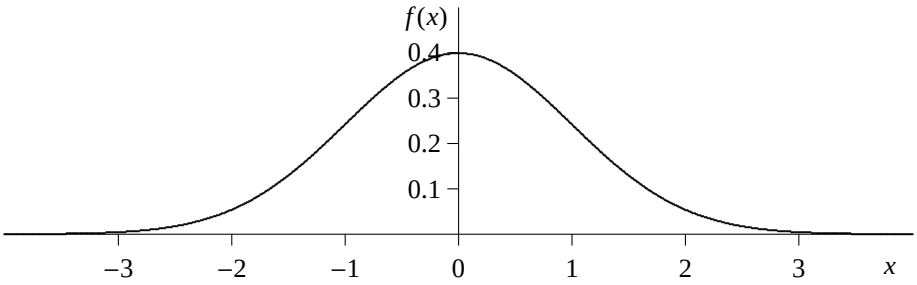
2.2.1 Normal Distribution: $N(0,1)$

The normal distribution with mean zero and variance one, i.e, the standard normal distribution, is represented by:

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2},$$

for $-\infty < x < \infty$, which is described in Figure 2.2.

Figure 2.2: Standard Normal Distribution: $N(0,1)$



In Example 1.4 (p.6), we have shown that the above function can be a probability density function. Moreover, as shown in Example 1.7 (p.14), mean, variance and the moment-generating function are given by:

$$\mathrm{E}(X) = 0, \qquad \mathrm{V}(X) = 1, \qquad \phi(\theta) = \exp\Big(\frac{1}{2}\theta^2\Big).$$

The normal random variable is constructed using two independent uniform random variables. This transformation is well known as the Box-Muller (1958) transformation and is shown as follows.

Let U_1 and U_2 be uniform random variables between zero and one. Suppose that U_1 is independent of U_2. Consider the following transformation:

$$\begin{aligned} X_1 &= \sqrt{-2\log(U_1)}\cos(2\pi U_2), \\ X_2 &= \sqrt{-2\log(U_1)}\sin(2\pi U_2). \end{aligned}$$

where we have $-\infty < X_1 < \infty$ and $-\infty < X_2 < \infty$ when $0 < U_1 < 1$ and $0 < U_2 < 1$. Then, the inverse transformation is given by:

$$u_1 = \exp\left(-\frac{x_1^2 + x_2^2}{2}\right), \qquad u_2 = \frac{1}{2\pi}\arctan\frac{x_2}{x_1}.$$

As shown in Section 1.4.2, we perform transformation of variables in multivariate cases. From this transformation, the Jacobian is obtained as:

$$J = \begin{vmatrix} \frac{\partial u_1}{\partial x_1} & \frac{\partial u_1}{\partial x_2} \\ \frac{\partial u_2}{\partial x_1} & \frac{\partial u_2}{\partial x_2} \end{vmatrix} = \begin{vmatrix} -x_1 \exp\Big(-\frac{1}{2}(x_1^2 + x_2^2)\Big) & -x_2 \exp\Big(-\frac{1}{2}(x_1^2 + x_2^2)\Big) \\ \frac{1}{2\pi}\frac{-x_2}{x_1^2 + x_2^2} & \frac{1}{2\pi}\frac{x_1}{x_1^2 + x_2^2} \end{vmatrix}$$

$$= -\frac{1}{2\pi}\exp\Big(-\frac{1}{2}(x_1^2 + x_2^2)\Big).$$

Let $f_x(x_1, x_2)$ be the joint density of X_1 and X_2 and $f_u(u_1, u_2)$ be the joint density of U_1 and U_2. Since U_1 and U_2 are assumed to be independent, we have the following:

$$f_u(u_1, u_2) = f_1(u_1) f_2(u_2) = 1,$$

where $f_1(u_1)$ and $f_2(u_2)$ are the density functions of U_1 and U_2, respectively. Note that $f_1(u_1) = f_2(u_2) = 1$ because U_1 and U_2 are uniform random variables between zero and one, which are shown in Section 2.1.1. Accordingly, the joint density of X_1 and X_2 is:

$$\begin{aligned} f_x(x_1, x_2) &= |J| f_u\Big(\exp(-\frac{x_1^2 + x_2^2}{2}), \frac{1}{2\pi}\arctan\frac{x_2}{x_1}\Big) \\ &= \frac{1}{2\pi}\exp\Big(-\frac{1}{2}(x_1^2 + x_2^2)\Big) \\ &= \frac{1}{\sqrt{2\pi}}\exp\Big(-\frac{1}{2}x_1^2\Big) \times \frac{1}{\sqrt{2\pi}}\exp\Big(-\frac{1}{2}x_2^2\Big), \end{aligned}$$

which is a product of two standard normal distributions. Thus, X_1 and X_2 are mutually independently distributed as normal random variables with mean zero and variance one. See Hogg and Craig (1995, pp.177 – 178). The source code of the standard normal random number generator shown above is given by `snrnd(ix,iy,rn)`.

`snrnd(ix,iy,rn)`

```
1:        subroutine snrnd(ix,iy,rn)
2:  c
3:  c  Use "snrnd(ix,iy,rn)"
4:  c  together with "urnd(ix,iy,rn)".
5:  c
6:  c  Input:
7:  c    ix, iy:  Seeds
8:  c  Output:
9:  c    rn: Standard Normal Random Draw N(0,1)
10: c
11:       pi= 3.1415926535897932385
12:       call urnd(ix,iy,rn1)
13:       call urnd(ix,iy,rn2)
14:       rn=sqrt(-2.0*log(rn1))*sin(2.0*pi*rn2)
```

```
15:         return
16:         end
```

`snrnd(ix,iy,rn)` should be used together with the uniform random number generator `urnd(ix,iy,rn)` shown in Section 2.1.2 (p.83).

`rn` in `snrnd(ix,iy,rn)` corresponds to X_2. Conventionally, one of X_1 and X_2 is taken as the random number which we use. Here, X_1 is excluded from consideration. `snrnd(ix,iy,rn)` includes the sine, which takes a lot of time computationally. Therefore, to avoid computation of the sine, various algorithms have been invented (Ahrens and Dieter (1988), Fishman (1996), Gentle (1998), Marsaglia, MacLaren and Bray (1964) and so on).

The alternative generator which does not have to compute the sine is shown below. Suppose that V_1 and V_2 are the uniform random variables within the unit circle, i.e., suppose that V_1 and V_2 are mutually independent uniform random variables between -1 and 1 which satisfies $V_1^2 + V_2^2 < 1$, $V_1 = 2U_1 - 1$ and $V_2 = 2U_2 - 1$, where both U_1 and U_2 are mutually independent uniform random variables between zero and one. Let R and Θ be the polar coordinates of (V_1, V_2), i.e., $R^2 = V_1^2 + V_2^2 < 1$. Define $S \equiv R^2$. Then, S and Θ are independent, where S is uniformly distributed over $(0, 1)$ and Θ is uniformly distributed over $(0, 2\pi)$. We can show the above fact as follows. The joint density of V_1 and V_2, denoted by $f_{12}(v_1, v_2)$, is given by:

$$f_{12}(v_1, v_2) = \begin{cases} \dfrac{1}{\pi}, & \text{for } v_1^2 + v_2^2 < 1, \\ 0, & \text{otherwise.} \end{cases}$$

Consider the following transformation:

$$V_1 = S^{1/2} \sin\Theta, \qquad V_2 = S^{1/2} \cos\Theta,$$

where $0 < S < 1$ and $0 < \Theta < 2\pi$. From this transformation, the Jacobian is obtained as:

$$J = \begin{vmatrix} \dfrac{\partial v_1}{\partial s} & \dfrac{\partial v_1}{\partial \theta} \\ \dfrac{\partial v_2}{\partial s} & \dfrac{\partial v_2}{\partial \theta} \end{vmatrix} = \begin{vmatrix} \dfrac{1}{2} s^{-1/2} \sin\theta & s^{1/2} \cos\theta \\ \dfrac{1}{2} s^{-1/2} \cos\theta & -s^{1/2} \sin\theta \end{vmatrix} = -\frac{1}{2}.$$

The joint density of S and Θ, $f_{s\theta}(s, \theta)$, is given by:

$$f_{s\theta}(s, \theta) = |J| f_{12}(s^{1/2} \cos\theta, s^{1/2} \sin\theta) = \frac{1}{2\pi},$$

where $0 < s < 1$ and $0 < \theta < 2\pi$. Let $f_s(s)$ and $f_\theta(\theta)$ be the marginal densities of S and Θ, respectively. When we take $f_s(s) = 1$ and $f_\theta(\theta) = 1/2\pi$, we can rewrite as $f_{s\theta}(s, \theta) = f_s(s) f_\theta(\theta)$, which implies that $S = R^2$ is independent of Θ, S is uniformly

distributed over $(0,1)$ and Θ is uniformly distributed on $(0, 2\pi)$. Thus, we have shown that both S and Θ are uniform.

Using S and Θ, the Box-Muller transformation is rewritten as:

$$\begin{aligned} X_1 &= \sqrt{-2\log(U_1)}\cos(2\pi U_2) \\ &= \sqrt{-2\log(U_1)}\frac{V_1}{S^{1/2}} = \sqrt{-2\log(S)}\frac{V_1}{S^{1/2}} = V_1\sqrt{\frac{-2\log(S)}{S}}, \end{aligned}$$

$$\begin{aligned} X_2 &= \sqrt{-2\log(U_1)}\sin(2\pi U_2) \\ &= \sqrt{-2\log(U_1)}\frac{V_2}{S^{1/2}} = \sqrt{-2\log(S)}\frac{V_2}{S^{1/2}} = V_2\sqrt{\frac{-2\log(S)}{S}}. \end{aligned}$$

Note that $V_1/S^{1/2}$ and $V_2/S^{1/2}$ are rewritten as $\cos(2\pi U_2)$ and $\sin(2\pi U_2)$, respectively. Moreover, S is uniformly distributed over $(0,1)$ and is independent of $V_1 S^{-1/2}$ and $V_2 S^{-1/2}$. Therefore, $\sqrt{-2\log(S)}$ is independent of $V_i/S^{1/2}$ for $i = 1, 2$. Thus, `snrnd2(ix,iy,rn)` is obtained, where we do not have to evaluate the sine or the cosine.

`snrnd2(ix,iy,rn)`

```
1:       subroutine snrnd2(ix,iy,rn)
2: c
3: c  Use "snrnd2(ix,iy,rn)"
4: c  together with "urnd(ix,iy,rn)".
5: c
6: c  Input:
7: c    ix, iy:  Seeds
8: c  Output:
9: c    rn: Standard Normal Random Draw N(0,1)
10: c
11:     1 call urnd(ix,iy,rn1)
12:       call urnd(ix,iy,rn2)
13:       s=( (2.*rn1-1.)**2 )+( (2.*rn2-1.)**2 )
14:       if(s.gt.1.) go to 1
15:       rn=sqrt(-2.*log(s)/s)*(2.*rn2-1.)
16:       return
17:       end
```

`snrnd2(ix,iy,rn)` should be used together with the uniform random number generator `urnd(ix,iy,rn)` shown in Section 2.1.2 (p.83). See Ross (1997) for the above discussion.

According to the past research (Gentle (1998, p.89), Kennedy and Gentle (1980, p.202), Ross (1997, pp.72 – 75) and so on), the Box-Muller transformation method `snrnd` is computationally not very efficient. However, recent development of personal computers decreases CPU time and therefore difference between `snrnd` and `snrnd2` is negligible, which will be discussed later in Section 3.5.1, where various standard

normal random number generators shown in this book are compared with respect to computation time.

As another random number generation method of standard normal distribution, we can utilize the central limit theorem. Let U_i be a uniform random draw between zero and one. Define $X = (1/n)\sum_{i=1}^{n} U_i$. Because $\mathrm{E}(U_i) = 1/2$ and $\mathrm{V}(U_i) = 1/12$, the central limit theorem shown in Section 1.6.3 indicates that

$$\frac{X - \mathrm{E}(X)}{\sqrt{\mathrm{V}(X)}} = \frac{X - 1/2}{\sqrt{1/12}/\sqrt{n}} \longrightarrow N(0,1),$$

where $n = 12$ is often taken. In `snrnd3(ix,iy,rn)`, we also use $n = 12$.

`snrnd3(ix,iy,rn)`

```
1:        subroutine snrnd3(ix,iy,rn)
2:  c
3:  c  Use "snrnd3(ix,iy,rn)"
4:  c  together with "urnd(ix,iy,rn)".
5:  c
6:  c  Input:
7:  c    ix, iy:  Seeds
8:  c  Output:
9:  c    rn: Standard Normal Random Draw N(0,1)
10: c
11:       n=12
12:       rn=0.0
13:         do 1 i=1,n
14:       call urnd(ix,iy,rn1)
15:     1 rn=rn+rn1/n
16:       rn=(rn-0.5)/sqrt(1./12./n)
17:       return
18:       end
```

`snrnd3(ix,iy,rn)` requires `urnd(ix,iy,rn)` on p.83.

As it is easily expected, because `snrnd3(ix,iy,rn)` requires 12 uniform random draws but both `snrnd(ix,iy,rn)` and `snrnd2(ix,iy,rn)` utilize two uniform random draws, it is obvious that `snrnd3(ix,iy,rn)` is computationally much slower than `snrnd(ix,iy,rn)` and `snrnd2(ix,iy,rn)`. In addition to computational point of view, `snrnd3(ix,iy,rn)` is out of the question from precision of the random draws, because it is the asymptotic result that the arithmetic average of uniform random draws goes to a normal distribution.

Standard Normal Probabilities

When $X \sim N(0,1)$, we have the case where we want to approximate p such that $p = F(x)$ given x, where $F(x) = \int_{-\infty}^{x} f(t)\,\mathrm{d}t = P(X < x)$. Adams (1969) reports that

$$P(X > x) = \int_x^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}t^2}\,\mathrm{d}t = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}\Big(\frac{1}{x+}\frac{1}{x+}\frac{2}{x+}\frac{3}{x+}\frac{4}{x+}\cdots\Big),$$

for $x > 0$, where the form in the parenthesis is called the continued fraction, which is defined as follows:

$$\frac{a_1}{x_1+}\frac{a_2}{x_2+}\frac{a_3}{x_3+}\cdots = \cfrac{a_1}{x_1+\cfrac{a_2}{x_2+\cfrac{a_3}{x_3+\cdots}}}.$$

A lot of approximations on the continued fraction shown above have been proposed. See Kennedy and Gentle (1980), Marsaglia (1964) and Marsaglia and Zaman (1994). Here, we introduce the following approximation (see Takeuchi (1989)):

$$P(X > x) = \frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}x^2}(b_1t + b_2t^2 + b_3t^3 + b_4t^4 + b_5t^5), \qquad t = \frac{1}{1+a_0x},$$
$$a_0 = 0.2316419, \quad b_1 = 0.319381530, \quad b_2 = -0.356563782,$$
$$b_3 = 1.781477937, \quad b_4 = -1.821255978, \quad b_5 = 1.330274429.$$

In `snprob(x,p)` below, $P(X < x)$ is shown. That is, `p` up to Line 19 is equal to $P(X > x)$ in `snprob(x,p)`. In Line 20, $P(X < x)$ is obtained.

`snprob(x,p)`

```
1:       subroutine snprob(x,p)
2: c
3: c  Input:
4: c    x:  N(0,1) Percent Point
5: c  Output:
6: c    p:  Probability corresponding to x
7: c
8:       pi= 3.1415926535897932385
9:       a0= 0.2316419
10:      b1= 0.319381530
11:      b2=-0.356563782
12:      b3= 1.781477937
13:      b4=-1.821255978
14:      b5= 1.330274429
15: c
16:      z=abs(x)
17:      t=1.0/(1.0+a0*z)
18:      pr=exp(-.5*z*z)/sqrt(2.0*pi)
19:      p=pr*t*(b1+t*(b2+t*(b3+t*(b4+b5*t))))
20:      if(x.gt.0.0) p=1.0-p
21: c
22:      return
23:      end
```

The maximum error of approximation of p is 7.5×10^{-8}, which practically gives us enough precision.

Standard Normal Percent Points

When $X \sim N(0,1)$, we approximate x such that $p = F(x)$ given p, where $F(x)$ indicates the standard normal cumulative distribution function, i.e., $F(x) = P(X < x)$, and p denotes probability. As shown in Odeh and Evans (1974), the approximation of a percent point is of the form:

$$x = y + \frac{S_4(y)}{T_4(y)} = y + \frac{p_0 + p_1 y + p_2 y^2 + p_3 y^3 + p_4 y^4}{q_0 + q_1 y + q_2 y^2 + q_3 y^3 + q_4 y^4},$$

where $y = \sqrt{-2\log(p)}$. $S_4(y)$ and $T_4(y)$ denote polynomials degree 4. The source code is shown in `snperpt(p,x)`, where x is obtained within $10^{-20} < p < 1 - 10^{-20}$.

snperpt(p,x)

```
      subroutine snperpt(p,x)
c
c  Input:
c    p:  Probability
c       (err<p<1-err, where err=1e-20)
c  Output:
c    x:  N(0,1) Percent Point corresponding to p
c
      p0=-0.322232431088
      p1=-1.0
      p2=-0.342242088547
      p3=-0.204231210245e-1
      p4=-0.453642210148e-4
      q0= 0.993484626060e-1
      q1= 0.588581570495
      q2= 0.531103462366
      q3= 0.103537752850
      q4= 0.385607006340e-2
      ps=p
      if( ps.gt.0.5 ) ps=1.0-ps
      if( ps.eq.0.5 ) x=0.0
      y=sqrt( -2.0*log(ps) )
      x=y+((((y*p4+p3)*y+p2)*y+p1)*y+p0)
     &   /((((y*q4+q3)*y+q2)*y+q1)*y+q0)
      if( p.lt.0.5 ) x=-x
      return
      end
```

The maximum error of approximation of x is 1.5×10^{-8} if the function is evaluated in double precision and 1.8×10^{-6} if it is evaluated in single precision.

The approximation of the form $x = y + S_2(y)/T_3(y)$ by Hastings (1955) gives a maximum error of 4.5×10^{-4}. To improve accuracy of the approximation, Odeh and Evans (1974) proposed the algorithm above.

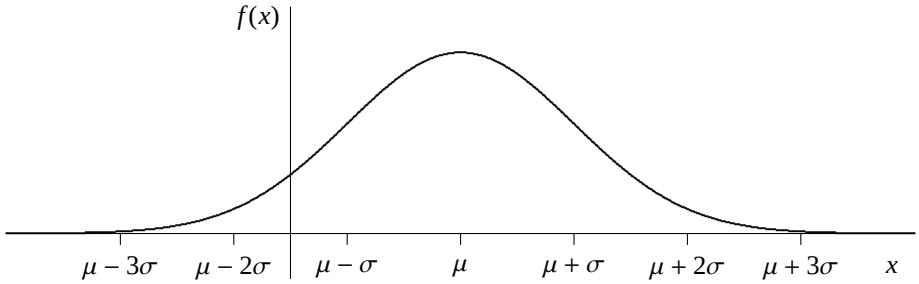
2.2.2 Normal Distribution: $N(\mu, \sigma^2)$

The normal distribution denoted by $N(\mu, \sigma^2)$ is represented as follows:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2},$$

for $-\infty < x < \infty$, which is displayed in Figure 2.3. μ is called a **location parameter** and σ^2 is a **scale parameter**.

Figure 2.3: Normal Distribution: $N(\mu, \sigma^2)$



Mean, variance and the moment-generating function of the normal distribution $N(\mu, \sigma^2)$ are given by:

$$\mathrm{E}(X) = \mu, \qquad \mathrm{V}(X) = \sigma^2, \qquad \phi(\theta) = \exp\left(\mu\theta + \frac{1}{2}\sigma^2\theta^2\right).$$

When $\mu = 0$ and $\sigma^2 = 1$ are taken, the above density function reduces to the standard normal distribution in Section 2.2.1.

As shown in Example 1.8 (p.15) and Example 1.9 (p.23), $X = \sigma Z + \mu$ is normally distributed with mean μ and variance σ^2, when $Z \sim N(0,1)$. Therefore, the source code is represented by `nrnd(ix,iy,ave,var,rn)`, where `ave` and `var` correspond to μ and σ^2, respectively.

`nrnd(ix,iy,ave,var,rn)`

```
1:        subroutine nrnd(ix,iy,ave,var,rn)
2:  C
3:  C  Use "nrnd(ix,iy,ave,var,rn)"
4:  C  together with "urnd(ix,iy,rn)"
5:  C            and  "snrnd(ix,iy,rn)".
6:  C
7:  C  Input:
8:  C    ix, iy:  Seeds
9:  C    ave: Mean
10: C    var: Variance
11: C  Output:
12: C    rn: Normal Random Draw N(ave,var)
13: C
```

```
14:       call snrnd(ix,iy,rn1)
15:       rn=ave+sqrt(var)*rn1
16:       return
17:       end
```

`nrnd(ix,iy,ave,var,rn)` should be used together with `urnd(ix,iy,rn)` on p.83 and `snrnd(ix,iy,rn)` on p.85. It is possible to replace `snrnd(ix,iy,rn)` by `snrnd2(ix,iy,rn)` or `snrnd3(ix,iy,rn)`. However, some applications in subsequent chapters utilize `snrnd(ix,iy,rn)`.

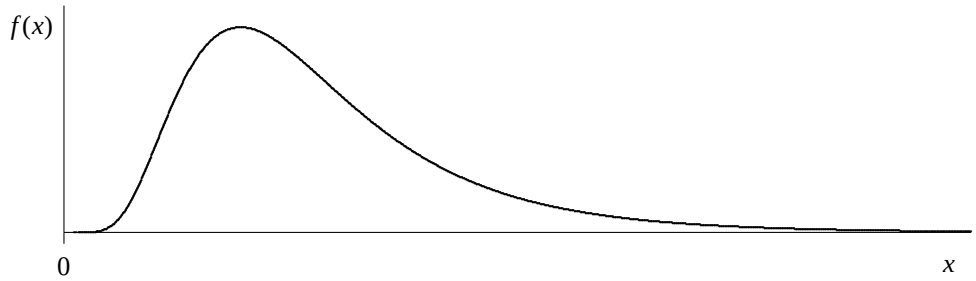
2.2.3 Log-Normal Distribution

It is known that the log-normal distribution with parameters μ and σ^2 is written as follows:

$$f(x) = \begin{cases} \dfrac{1}{x\sqrt{2\pi\sigma^2}} \exp\Big(-\dfrac{1}{2\sigma^2}(\log x - \mu)^2\Big), & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise.} \end{cases}$$

The log-normal distribution is shown in Figure 2.4.

Figure 2.4: Log-Normal Distribution



Mean and variance are given by:

$$\mathrm{E}(X) = \exp\Big(\mu + \frac{1}{2}\sigma^2\Big), \qquad \mathrm{V}(X) = \exp(2\mu + 2\sigma^2) - \exp(2\mu + \sigma^2).$$

For $Z \sim N(\mu, \sigma^2)$ and $X = e^Z$, it is known that X is distributed as a log-normal random variable. Because we have $z = \log(x)$, the Jacobian is:

$$J = \frac{\mathrm{d}z}{\mathrm{d}x} = \frac{1}{x}.$$

Therefore, the log-normal distribution with parameters μ and σ^2 is given by:

$$f(x) = |J| f_z(\log(x)) = \frac{1}{x\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(\log x - \mu)^2},$$

where $f(\cdot)$ and $f_z(\cdot)$ denote the probability density functions of X and Z, respectively. Note that $0 < x < \infty$ because of $x = e^z$ and $-\infty < z < \infty$. Thus, the log-normal random draws are based on the normal distribution. The source code is written as `lognrnd(ix,iy,ave,var,rn)`.

`lognrnd(ix,iy,ave,var,rn)`

```
      subroutine lognrnd(ix,iy,ave,var,rn)
c
c  Use "lognrnd(ix,iy,ave,var,rn)"
c  together with "urnd(ix,iy,rn)"
c           and  "snrnd(ix,iy,rn)".
c
c  Input:
c    ix, iy:  Seeds
c    ave: Mean of N(ave,var)
c    var: Variance of N(ave,var)
c  Output:
c    rn: Log-Normal Random Draw,
c        i.e.,  exponential of N(ave,var)
c
       call snrnd(ix,iy,rn1)
       rn=exp(ave+sqrt(var)*rn1)
       return
       end
```

`lognrnd(ix,iy,ave,var,rn)` should be used together with `urnd(ix,iy,rn)` on p.83 and `snrnd(ix,iy,rn)` on p.85.

2.2.4 Exponential Distribution

The exponential distribution with parameter β is written as:

$$f(x) = \begin{cases} \dfrac{1}{\beta} e^{-\frac{x}{\beta}}, & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

for $\beta > 0$. β indicates a scale parameter. The functional form of the exponential density is described in Figure 2.5.

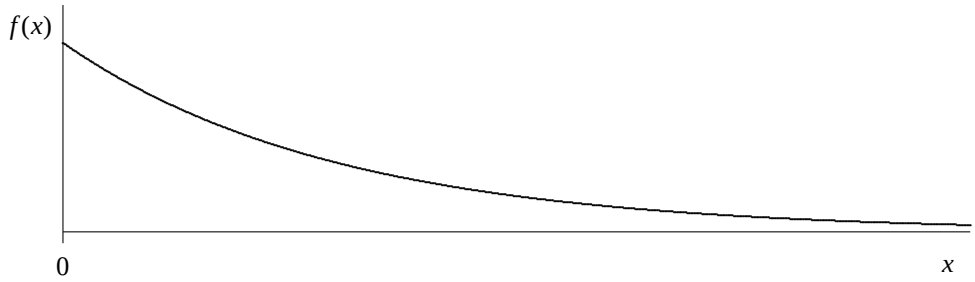
Mean, variance and the moment-generating function are obtained as follows:

$$\mathrm{E}(X) = \beta, \qquad \mathrm{V}(X) = \beta^2, \qquad \phi(\theta) = \frac{1}{1-\beta\theta}.$$

The relation between the exponential random variable the uniform random variable is shown as follows: When $U \sim U(0,1)$, consider the following transformation:

$$X = -\beta \log(U).$$

Figure 2.5: Exponential Distribution



Then, X is an exponential distribution with parameter β.

Because the transformation is given by $u = \exp(-x/\beta)$, the Jacobian is:

$$J = \frac{\mathrm{d}u}{\mathrm{d}x} = -\frac{1}{\beta}\exp\Big(-\frac{1}{\beta}x\Big).$$

By transforming the variables, the density function of X is represented as:

$$f(x) = |J| f_u\Big(\exp(-\frac{1}{\beta}x)\Big) = \frac{1}{\beta}\exp\Big(-\frac{1}{\beta}x\Big),$$

where $f(\cdot)$ and $f_u(\cdot)$ denote the probability density functions of X and U, respectively. Note that $0 < x < \infty$ because of $x = -\beta\log(u)$ and $0 < u < 1$. Thus, the exponential distribution with parameter β is obtained from the uniform random draw between zero and one.

exprnd(ix,iy,beta,rn)

```
1:       subroutine exprnd(ix,iy,beta,rn)
2: c
3: c  Use "exprnd(ix,iy,beta,rn)"
4: c  together with "urnd(ix,iy,rn)".
5: c
6: c  Input:
7: c    ix, iy:  Seeds
8: c    beta: Parameter
9: c  Output:
10: c    rn: Exponential Random Draw
11: c        with Parameter beta
12: c
13:       call urnd(ix,iy,rn1)
14:       rn=-beta*log(rn1)
15:       return
16:       end
```

`exprnd(ix,iy,beta,rn)` should be used together with `urnd(ix,iy,rn)` on p.83.

When $\beta = 2$, the exponential distribution reduces to the chi-square distribution with 2 degrees of freedom. See Section 2.2.8 for the chi-square distribution.

2.2.5 Gamma Distribution: $G(\alpha,\beta)$

The gamma distribution with parameters α and β, denoted by $G(\alpha,\beta)$, is represented as follows:

$$f(x) = \begin{cases} \dfrac{1}{\beta^{\alpha}\Gamma(\alpha)} x^{\alpha-1} e^{-\frac{x}{\beta}}, & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

for $\alpha > 0$ and $\beta > 0$, where α is called a **shape parameter** and β denotes a scale parameter. $\Gamma(\cdot)$ is called the **gamma function**, which is the following function of α:

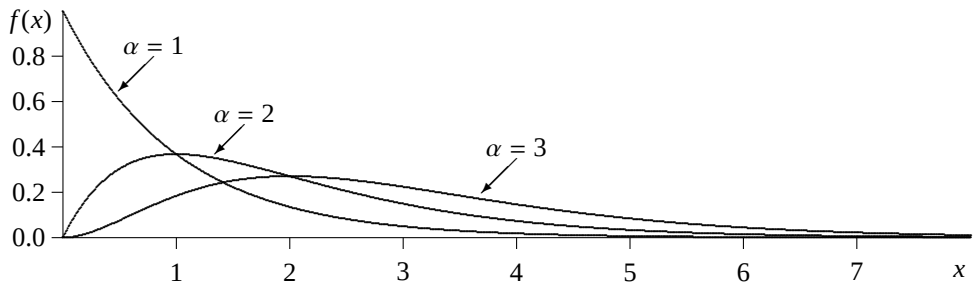
$$\Gamma(\alpha) = \int_0^{\infty} x^{\alpha-1} e^{-x}\, \mathrm{d}x.$$

The gamma function has the following features:

$$\Gamma(\alpha+1) = \alpha\Gamma(\alpha), \qquad \Gamma(1) = 1, \qquad \Gamma\Big(\frac{1}{2}\Big) = 2\Gamma\Big(\frac{3}{2}\Big) = \sqrt{\pi}.$$

The gamma distributions with parameters $\alpha = 1, 2, 3$ and $\beta = 1$ are displayed in Figure 2.6.

Figure 2.6: Gamma Distribution: $\alpha = 1, 2, 3$ and $\beta = 1$



Mean, variance and the moment-generating function are given by:

$$\mathrm{E}(X) = \alpha\beta, \qquad \mathrm{V}(X) = \alpha\beta^2, \qquad \phi(\theta) = \frac{1}{(1-\beta\theta)^{\alpha}}.$$

The gamma distribution with $\alpha = 1$ is equivalent to the exponential distribution shown in Section 2.2.4. This fact is easily checked by comparing both moment-generating functions.

Now, utilizing the uniform random variable, the gamma distribution with parameters α and β are derived as follows. The derivation shown in this section deals with the case where α is a positive integer, i.e., $\alpha = 1, 2, 3, \cdots$. The random variables Z_1, Z_2, $\cdots$, Z_{α} are assumed to be mutually independently distributed as exponential random variables with parameter β, which are shown in Section 2.2.4. Define $X = \sum_{i=1}^{\alpha} Z_i$.

Then, X has distributed as a gamma distribution with parameters α and β, where α should be an integer, which is proved as follows:

$$\phi_x(\theta) = \mathrm{E}(e^{\theta X}) = \mathrm{E}(e^{\theta \sum_{i=1}^{\alpha} Z_i}) = \prod_{i=1}^{\alpha} \mathrm{E}(e^{\theta Z_i}) = \prod_{i=1}^{\alpha} \phi_i(\theta) = \prod_{i=1}^{\alpha} \frac{1}{1-\beta\theta}$$
$$= \frac{1}{(1-\beta\theta)^{\alpha}},$$

where $\phi_x(\theta)$ and $\phi_i(\theta)$ represent the moment-generating functions of X and Z_i, respectively. Thus, sum of the α exponential random variables yields the gamma random variable with parameters α and β. Therefore, the source code which generates gamma random numbers is shown in `gammarnd(ix,iy,alpha,beta,rn)`.

`gammarnd(ix,iy,alpha,beta,rn)`

```
1:        subroutine gammarnd(ix,iy,alpha,beta,rn)
2:  c
3:  c  Use "gammarnd(ix,iy,alpha,beta,rn)"
4:  c  together with "exprnd(ix,iy,beta,rn)"
5:  c           and  "urnd(ix,iy,rn)".
6:  c
7:  c  Input:
8:  c    ix, iy:   Seeds
9:  c    alpha:    Shape Parameter (which should be an integer)
10: c    beta:     Scale Parameter
11: c  Output:
12: c    rn: Gamma Random Draw with alpha and beta
13: c
14:       rn=0.0
15:        do 1 i=1,nint(alpha)
16:       call exprnd(ix,iy,beta,rn1)
17:     1 rn=rn+rn1
18:       return
19:       end
```

`gammarnd(ix,iy,alpha,beta,rn)` is utilized together with `urnd(ix,iy,rn)` on p.83 and `exprnd(ix,iy,rn)` on p.94.

As pointed out above, α should be an integer in the source code. When α is large, we have serious problems computationally in the above algorithm, because α exponential random draws have to be generated to obtain one gamma random draw with parameters α and β. To improve these problems, see Section 3.5.2, where α takes any positive real number and we can generate random draws very quickly.

When $\alpha = k/2$ and $\beta = 2$, the gamma distribution reduces to the chi-square distribution with k degrees of freedom. See Section 2.2.8 for the chi-square distribution.

2.2.6 Inverse Gamma Distribution: $IG(\alpha,\beta)$

The inverse gamma distribution with parameters α and β, denoted by $IG(\alpha,\beta)$, is represented as:

$$f(x) = \begin{cases} \dfrac{2}{\Gamma(\alpha)\beta^{\alpha}x^{2\alpha+1}}\exp\Big(-\dfrac{1}{\beta x^2}\Big), & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

for $\alpha > 0$ and $\beta > 0$.

The inverse gamma distribution is derived from the gamma distribution in Section 2.2.5. Let Z be the gamma distribution with parameters α and β, which is denoted by $f_z(z)$. Define $X = Z^{-1/2}$. Then, X has the inverse gamma distribution shown above. Let $f(x)$ be the density function of X. Transforming the variable from Z to X, the density function $f(x)$ is derived as:

$$\begin{aligned} f(x) &= |J| f_z(x^{-2}) = |-2x^{-3}|\frac{1}{\beta^{\alpha}\Gamma(\alpha)}\Big(\frac{1}{x^2}\Big)^{\alpha-1}\exp\Big(-\frac{1}{\beta x^2}\Big) \\ &= \frac{2}{\Gamma(\alpha)\beta^{\alpha}x^{2\alpha+1}}\exp\Big(-\frac{1}{\beta x^2}\Big), \end{aligned}$$

where the Jacobian is given by $J = \mathrm{d}z/\mathrm{d}x = -2x^{-3}$. Therefore, we can obtain the distribution of X as $X \sim IG(\alpha,\beta)$ for $Z \sim G(\alpha,\beta)$ and $X = Z^{-1/2}$. The Fortran 77 source program for the inverse gamma random number generator with parameters α and β is shown as `igammarnd(ix,iy,alpha,beta,rn)`.

`igammarnd(ix,iy,alpha,beta,rn)`

```
      subroutine igammarnd(ix,iy,alpha,beta,rn)
C
C  Use "igammarnd(ix,iy,alpha,beta,rn)"
C  together with "gammarnd(ix,iy,alpha,beta,rn)",
C                "exprnd(ix,iy,beta,rn)"
C           and  "urnd(ix,iy,rn)".
C
C  Input:
C    ix, iy:   Seeds
C    alpha:    Shape Parameter
C    beta:     Scale Parameter
C  Output:
C    rn: Inverse Gamma Random Draw
C                 with alpha and beta
C
       call gammarnd(ix,iy,alpha,beta,rn1)
       rn=1./sqrt(rn1)
       return
       end
```

Note that `igammarnd(ix,iy,alpha,beta,rn)` have to be utilized together with the three sub-programs: `urnd(ix,iy,rn)` on p.83, `exprnd(ix,iy,rn)` on p.94 and `gammarnd(ix,iy,alpha,beta,rn)` on p.96.

As discussed in Section 2.2.5, `gammarnd(ix,iy,alpha,beta,rn)` is not efficient in the sense of computation time. When `gammarnd(ix,iy,alpha,beta,rn)` on p.96 in Line 16 is replaced by `gammarnd8(ix,iy,alpha,beta,rn)` in Section 3.5.2, p.213, the shape parameter `alpha` can take any positive real number and `igammarnd(ix,iy,alpha,beta,rn)` becomes computationally efficient.

We can rewrite the inverse gamma distribution above, letting $\sigma = x$, $\alpha = \dfrac{m}{2}$ and $\beta = \dfrac{2}{ms^2}$ to obtain:

$$f_\sigma(\sigma) = \begin{cases} \dfrac{2}{\Gamma(\frac{m}{2})}\Big(\dfrac{ms^2}{2}\Big)^{m/2}\dfrac{1}{\sigma^{m+1}}\exp\Big(-\dfrac{ms^2}{2\sigma^2}\Big), & \text{for } 0 < \sigma < \infty, \\ 0, & \text{otherwise,} \end{cases} \tag{2.1}$$

where $m > 0$ and $s > 0$. The first- and the second-moments are given by:

$$\mathrm{E}(\sigma) = \frac{\Gamma(\frac{m-1}{2})}{\Gamma(\frac{m}{2})}\Big(\frac{m}{2}\Big)^{\frac{1}{2}}s, \qquad \text{for } m > 1,$$

$$\mathrm{E}(\sigma^2) = \frac{\Gamma(\frac{m}{2}-1)}{\Gamma(\frac{m}{2})}\Big(\frac{m}{2}\Big)s^2 = \frac{m}{m-2}s^2, \qquad \text{for } m > 2.$$

This inverse gamma distribution is often used in a Bayesian framework, which is taken as the prior density function of the standard deviation associated with the normal density. See Zellner (1971, pp.371 – 373) for the inverse gamma distribution.

Finally, note as follows. Suppose that $Z \sim G(\alpha,\beta)$. Let us define $X = Z^{-1}$, not $X = Z^{-1/2}$. The distribution of X is also sometimes called the inverse gamma distribution $IG(\alpha,\beta)$. In this case, the probability density function of X, $f(x)$, is given by:

$$f(x) = |J| f_z(x^{-1}) = \frac{1}{\beta^\alpha\Gamma(\alpha)}\frac{1}{x^{\alpha+1}}\exp\Big(-\frac{1}{\beta x}\Big),$$

where the Jacobian is given by $J = \mathrm{d}z/\mathrm{d}x = -1/x^2$ and $f_z(z)$ is the gamma distribution shown in Section 2.2.5.

2.2.7 Beta Distribution

The beta distribution with parameters α and β is of the form:

$$f(x) = \begin{cases} \dfrac{1}{B(\alpha,\beta)}x^{\alpha-1}(1-x)^{\beta-1}, & \text{for } 0 < x < 1, \\ 0, & \text{otherwise,} \end{cases}$$

for $\alpha > 0$ and $\beta > 0$. $B(\cdot,\cdot)$ is called the **beta function**, which is shown as:

$$B(\alpha,\beta) = \int_0^1 x^{\alpha-1}(1-x)^{\beta-1}\,\mathrm{d}x = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}.$$

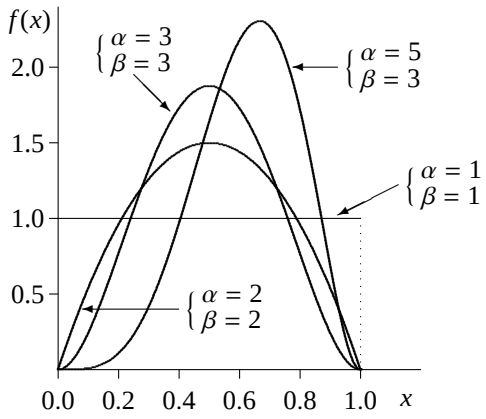
Thus, the beta function is related to the gamma function.

Mean and variance are as follows:

$$\mathrm{E}(X) = \frac{\alpha}{\alpha+\beta}, \qquad \mathrm{V}(X) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}.$$

The beta distributions with $(\alpha,\beta) = (1,1), (2,2), (3,3), (5,3)$ are in Figure 2.7. As shown in Figure 2.7, the beta distribution with $\alpha = \beta = 1$ reduces to the uniform distribution between zero and one, i.e., $U(0,1)$.

Figure 2.7: Beta Distribution



From two gamma random variables, one beta random variable can be derived. Suppose that Z_1 and Z_2 are independently distributed with $Z_1 \sim G(\alpha,1)$ and $Z_2 \sim G(\beta,1)$. Then, $X = Z_1/(Z_1+Z_2)$ has a beta distribution with parameters α and β, which is shown as follows. Define $X = Z_1/(Z_1+Z_2)$ and $Y = Z_1+Z_2$, where we have $0 < X < 1$ and $0 < Y < \infty$. Since $z_1 = xy$ and $z_2 = y(1-x)$, the Jacobian is written as:

$$J = \begin{vmatrix} \dfrac{\partial z_1}{\partial x} & \dfrac{\partial z_1}{\partial y} \\ \dfrac{\partial z_2}{\partial x} & \dfrac{\partial z_2}{\partial y} \end{vmatrix} = \begin{vmatrix} y & x \\ -y & 1-x \end{vmatrix} = y.$$

The joint density of X and Y is represented as:

$$\begin{aligned} f_{xy}(x,y) &= |J| f_{12}(xy, y(1-x)) = |J| f_1(xy) f_2(y(1-x)) \\ &= y\Big(\frac{1}{\Gamma(\alpha)}(xy)^{\alpha-1}e^{-xy}\Big)\Big(\frac{1}{\Gamma(\beta)}(y(1-x))^{\beta-1}e^{-y(1-x)}\Big) \\ &= \frac{1}{\Gamma(\alpha)\Gamma(\beta)}x^{\alpha-1}(1-x)^{\beta-1}y^{\alpha+\beta-1}e^{-y}, \end{aligned}$$

for $0 < x < 1$ and $0 < y < \infty$. Note that the joint density of Z_1 and Z_2 is given by: $f_{12}(z_1, z_2) = f_1(z_1) f_2(z_2)$, because Z_1 is assumed to be independent of Z_2, where $f_{12}(z_1, z_2)$, $f_1(z_1)$ and $f_2(z_2)$ denote the joint density of Z_1 and Z_2, the marginal density of Z_1 and the marginal density of Z_2, respectively. The marginal density function of X is obtained as:

$$\begin{aligned} f(x) &= \int_0^\infty f_{xy}(x,y)\,\mathrm{d}y = \frac{1}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1} \int_0^\infty y^{\alpha+\beta-1} e^{-y}\,\mathrm{d}y \\ &= \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1} \int_0^\infty \frac{1}{\Gamma(\alpha+\beta)} y^{\alpha+\beta-1} e^{-y}\,\mathrm{d}y \\ &= \frac{1}{B(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1}. \end{aligned}$$

The function in the integration of the second line corresponds to the gamma distribution function $G(\alpha+\beta, 1)$, and the integration of the gamma distribution function is equal to one. Thus, the marginal density of X is given by the beta distribution. Using two independent gamma random draws, a beta random draw is generated as shown in `betarnd(ix,iy,alpha,beta,rn)`.

`betarnd(ix,iy,alpha,beta,rn)`

```
1:        subroutine betarnd(ix,iy,alpha,beta,rn)
2:  c
3:  c  Use "betarnd(ix,iy,alpha,beta,rn)"
4:  c  together with "gammarnd(ix,iy,alpha,beta,rn)",
5:  c                "exprnd(ix,iy,alpha,rn)"
6:  c           and  "urnd(ix,iy,rn)".
7:  c
8:  c  Input:
9:  c    ix, iy:  Seeds
10: c    alpha, beta: Parameters
11: c  Output:
12: c    rn: Beta Random Draw with Parameters alpha and beta
13: c
14:       call gammarnd(ix,iy,alpha,1.0,rn1)
15:       call gammarnd(ix,iy,beta ,1.0,rn2)
16:       rn=rn1/(rn1+rn2)
17:       return
18:       end
```

In the source code shown above, note that `betarnd(ix,iy,alpha,beta,rn)` should be used together with `urnd(ix,iy,rn)` on p.83, `exprnd(ix,iy,rn)` on p.94 and `gammarnd(ix,iy,alpha,beta,rn)` on p.96.

Because the `alpha` included in `gammarnd(ix,iy,alpha,beta,rn)` of Section 2.2.5 (p.96) should be a positive integer, both `alpha` and `beta` in the above algorithm `betarnd(ix,iy,alpha,beta,rn)` have to be integers. If we use the gamma random number generator `gammarnd8(ix,iy,alpha,beta,rn)`, which is shown in

Section 3.5.2 (p.213), both `alpha` and `beta` in `betarnd(ix,iy,alpha,beta,rn)` can take any positive real numbers. In addition to this crucial problem, the subroutine `gammarnd(ix,iy,alpha,beta,rn)` computationally takes much more time than `gammarnd8(ix,iy,alpha,beta,rn)`. Therefore, in Lines 14 and 15 of `betarnd`, it is much better for us to use `gammarnd8(ix,iy,alpha,beta,rn)` on p.213, rather than `gammarnd(ix,iy,alpha,beta,rn)` on p.96.

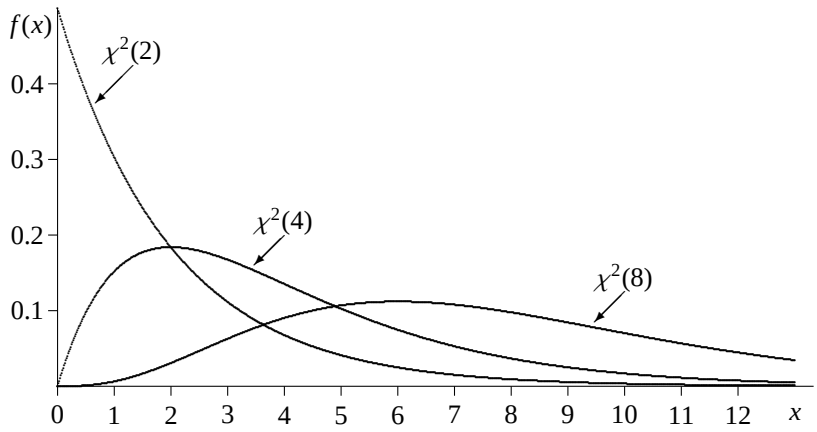
2.2.8 Chi-Square Distribution: $\chi^2(k)$

The chi-square distribution with k degrees of freedom, denoted by $\chi^2(k)$, is written as follows:

$$f(x) = \begin{cases} \dfrac{1}{2^{k/2}\Gamma(\frac{k}{2})} x^{\frac{k}{2}-1} e^{-\frac{1}{2}x}, & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

where k is a positive integer. The chi-square distribution is displayed in Figure 2.8 for $k = 2, 4, 8$. The chi-square distribution with $k = 2$ reduces to the exponential distribution with $\beta = 2$, shown in Section 2.2.4.

Figure 2.8: Chi-Square Distribution: $\chi^2(k)$



Mean, variance and the moment-generating function are given by:

$$\mathrm{E}(X) = k, \qquad \mathrm{V}(X) = 2k, \qquad \phi(\theta) = \frac{1}{(1-2\theta)^{k/2}}.$$

Suppose that $Z_1, Z_2, \cdots, Z_k$ are mutually independently distributed as standard normal random variables. Then, $X = \sum_{i=1}^{k} Z_i^2$ has a chi-square distribution with k degrees of freedom. We prove this in two steps: (i) $X_1 \sim \chi^2(1)$ for $Z \sim N(0,1)$ and $X_1 = Z^2$, and (ii) $X \sim \chi^2(k)$ for $X_i \sim \chi^2(1)$ and $X = \sum_{i=1}^{k} X_i$, where $X_1, X_2, \cdots, X_n$ are assumed to be mutually independent.

First, we show that $X_1 = Z^2 \sim \chi^2(1)$ for $Z \sim N(0,1)$. Remember that the cumulative distribution function of Z and its derivative (i.e., the probability density function) are represented as (see Section 2.2.1 for the standard normal density function):

$$F_z(z) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} \exp\Big(-\frac{1}{2}u^2\Big)\,\mathrm{d}u, \qquad f_z(z) = F_z'(z) = \frac{1}{\sqrt{2\pi}} \exp\Big(-\frac{1}{2}z^2\Big).$$

When we have the transformation of $X_1 = Z^2$, the probability density function of X_1, i.e., $f_1(x_1)$, is:

$$\begin{aligned} f_1(x_1) = F_1'(x_1) &= \frac{1}{2\sqrt{x_1}}\Big(F_z'(\sqrt{x_1}) + F_z'(-\sqrt{x_1})\Big) = \frac{1}{\sqrt{2\pi}\sqrt{x_1}} \exp\Big(-\frac{1}{2}x_1\Big) \\ &= \frac{1}{\Gamma(\frac{1}{2})2^{\frac{1}{2}}} x_1^{\frac{1}{2}-1} \exp\Big(-\frac{1}{2}x_1\Big), \end{aligned}$$

for $x_1 > 0$, which corresponds to the chi-square distribution with one degree of freedom, i.e., $X_1 \sim \chi^2(1)$. See p.23 for the second equality. Note that $\Gamma(\frac{1}{2}) = \sqrt{\pi}$ in the last equality.

Second, we show that $X = \sum_{i=1}^{k} X_i \sim \chi^2(k)$, where $X_1, X_2, \cdots, X_k$ are mutually independently distributed as $X_i \sim \chi^2(1)$ for $i = 1, 2, \cdots, k$. The moment-generating function of X_i, $\phi_i(\theta)$, is given by $\phi_i(\theta) = (1-2\theta)^{-1/2}$ for $i = 1, 2, \cdots, k$. Therefore, the moment-generating function of X, $\phi_x(\theta)$, is obtained as follows:

$$\begin{aligned} \phi_x(\theta) = \mathrm{E}(e^{\theta X}) = \mathrm{E}(e^{\theta \sum_{i=1}^{k} X_i}) &= \prod_{i=1}^{k} \mathrm{E}(e^{\theta X_i}) = \prod_{i=1}^{k} \phi_i(\theta) = \prod_{i=1}^{k} \frac{1}{(1-2\theta)^{1/2}} \\ &= \frac{1}{(1-2\theta)^{k/2}}, \end{aligned}$$

which is equivalent to the moment-generating function of $\chi^2(k)$ distribution.

Thus, it is shown that $X = \sum_{i=1}^{k} Z_i^2$ has the chi-square distribution with k degrees of freedom when $Z_1, Z_2, \cdots, Z_k$ are mutually independently distributed as standard normal random variables. Therefore, we see that the chi-square random draw is obtained from the sum of squared standard normal random draws. The source code is shown in `chi2rnd(ix,iy,k,rn)`.

`chi2rnd(ix,iy,k,rn)`

```
1:       subroutine chi2rnd(ix,iy,k,rn)
2: c
3: c  Use "chi2rnd(ix,iy,k,rn)"
4: c  together with "snrnd(ix,iy,rn)"
5: c            and  "urnd(ix,iy,rn)".
6: c
7: c  Input:
8: c    ix, iy:  Seeds
```

```
c    k:  Degree of Freedom
c  Output:
c    rn: Chi-Square Random Draw
c        with k Degrees of Freedom
c
       rn=0.0
        do 1 i=1,k
       call snrnd(ix,iy,rn1)
     1 rn=rn+rn1*rn1
       return
       end
```

Note that chi2rnd(ix,iy,k,rn) should be used with urnd(ix,iy,rn) on p.83 and snrnd(ix,iy,rn) on p.85.

In order to perform chi2rnd(ix,iy,k,rn), we need to generate k standard normal random draws. One standard normal random draw consists of two uniform random draws (see snrnd(ix,iy,rn) on p.85). Therefore, one chi-square random draw with k degrees of freedom requires $2k$ uniform random draws. Thus, we can see that chi2rnd(ix,iy,k,rn) is computationally an inefficient random number generator. To improve this computational inefficiency, various chi-square random number generators have been invented.

The exponential distribution with parameter $\beta = 2$ is equivalent to the chi-square distribution with 2 degrees of freedom, which can be checked by comparing both the moment-generating function of chi-square distribution and that of exponential distribution. Therefore, as an alternative random number generation, when k is even we need to generate independent $k/2$ exponential random draws, and when k is odd we generate $[k/2]$ exponential random draws independently and one standard normal random draw, where $[k/2]$ indicates the maximum integer not greater than $k/2$. In addition, remember that one exponential random draw comes from one uniform random draw, as shown in Section 2.2.4. The random number generator based on this approach is given by chi2rnd2(ix,iy,k,rn). chi2rnd(ix,iy,k,rn) requires k standard normal random draws (or $2k$ uniform random draws). Accordingly, chi2rnd2(ix,iy,k,rn) has almost a quarter of computational burden of chi2rnd(ix,iy,k,rn). See Rubinstein (1981, pp.93 – 94) for this algorithm.

chi2rnd2(ix,iy,k,rn)

```
      subroutine chi2rnd2(ix,iy,k,rn)
c
c  Use "chi2rnd2(ix,iy,k,rn)"
c  together with "snrnd(ix,iy,rn)",
c                "exprnd(ix,iy,alpha,rn)"
c           and  "urnd(ix,iy,rn)".
c
c  Input:
c    ix, iy:  Seeds
c    k:  Degree of Freedom
c  Output:
```

```
12: c     rn: Chi-Square Random Draw
13: c         with k Degrees of Freedom
14: c
15:           if(k-(k/2)*2.eq.1) then
16:         call snrnd(ix,iy,rn1)
17:         rn=rn1*rn1
18:           else
19:         rn=0.0
20:           endif
21:           do 1 i=1,k/2
22:         call exprnd(ix,iy,2.0,rn1)
23:       1 rn=rn+rn1
24:         return
25:         end
```

`chi2rnd2(ix,iy,k,rn)` is used simultaneously with `urnd(ix,iy,rn)` on p.83, `snrnd(ix,iy,rn)` on p.85 and `exprnd(ix,iy,alpha,rn)` on p.94.

We can see that `chi2rnd2(ix,iy,k,rn)` is computationally much more efficient than `chi2rnd(ix,iy,k,rn)`. However, `chi2rnd2(ix,iy,k,rn)` is still computer-intensive when k is large. It is the most efficient to utilize the gamma random draw, which is discussed later in Section 3.5.2.

Now we introduce one more generator, which gives us the least computational generator but the most imprecise generator. The central limit theorem is utilized, which is shown in Section 1.6.3. We want to generate random draws of $X = \sum_{i=1}^{k} Z_i$. Since $Z_i \sim \chi^2(1)$, we know that $\mathrm{E}(Z_i) = 1$ and $\mathrm{V}(Z_i) = 2$. Therefore, mean and variance of the sample mean X/k are given by $\mathrm{E}(X/k) = 1$ and $\mathrm{V}(X/k) = 2/k$. Then, the central limit theorem indicates that as $k \longrightarrow \infty$, we have:

$$\frac{X/k - 1}{\sqrt{2/k}} \longrightarrow N(0, 1).$$

Thus, when k is large, X is approximately distributed as a normal random draw with mean k and variance $2k$. Therefore, one standard normal random draw yields one $\chi^2(k)$ random draw. That is, X can be approximated as the $\chi^2(k)$ random variable when $Z \sim N(0, 1)$ and $X = Z\sqrt{2k} + k$. The source code is shown in `chi2rnd3(ix,iy,k,rn)`.

`chi2rnd3(ix,iy,k,rn)`

```
1:        subroutine chi2rnd3(ix,iy,k,rn)
2: c
3: c  Use "chi2rnd3(ix,iy,k,rn)"
4: c  together with "snrnd(ix,iy,rn)"
5: c            and  "urnd(ix,iy,rn)".
6: c
7: c  Input:
8: c    ix, iy:  Seeds
9: c    k:  Degree of Freedom
10: c  Output:
11: c    rn: Chi-Square Random Draw
12: c        with k Degrees of Freedom
```

```
13: c
14:       call snrnd(ix,iy,rn1)
15:       rn=k+sqrt(2.*k)*rn1
16:       return
17:       end
```

`chi2rnd3(ix,iy,k,rn)` should be used simultaneously with `urnd(ix,iy,rn)` on p.83 and `snrnd(ix,iy,rn)` on p.85.

The random draws of X generated from `chi2rnd3(ix,iy,k,rn)` are possibly negative. Therefore, `chi2rnd3(ix,iy,k,rn)` might be practically useless. The $\chi^2(k)$ random number generator will be discussed in Section 3.5.2, which is much more precise and relatively less computational generator.

Chi-Square Probabilities

Let $Q(x;k)$ be $P(X > x)$, where $X \sim \chi^2(k)$. Using the integration by parts, we have the following recursive algorithm:

$$f(x;i) = \frac{1}{i-2} x f(x;i-2),$$
$$Q(x;i) = Q(x;i-2) + 2f(x;i)$$

for $i = 3, 4, \cdots, k$, where the initial values are given by:

$$f(x;1) = \frac{1}{\sqrt{2\pi x}} \exp(-\frac{1}{2}x), \qquad f(x;2) = \frac{1}{2}\exp(-\frac{1}{2}x),$$
$$Q(x;1) = 2(1-\Phi(\sqrt{x})), \qquad Q(x;2) = \exp(-\frac{1}{2}x).$$

In `chi2prob(x,k,p)` below, $P(X < x)$ is shown. That is, `q(k)` in Line 19 is equivalent to $P(X > x)$ and `p` represents $P(X < x)$.

`chi2prob(x,k,p)`

```
1:        subroutine chi2prob(x,k,p)
2:        dimension f(1000),q(1000)
3:  c
4:  c  Input:
5:  c    x:  Chi^2(k) Percent Point
6:  c    k:  Degree of freedom (less than or equal to 1000)
7:  c  Output:
8:  c    p:  Probability corresponding to x, i.e., Prob(X<x)=p
9:  c
10:       pi= 3.1415926535897932385
11:       f(1)=exp(-.5*x)/sqrt(2.*pi*x)
12:       f(2)=exp(-.5*x)/2.
13:       call snprob(sqrt(x),pr)
14:       q(1)=2.*(1.-pr)
15:       q(2)=exp(-.5*x)
```

```
16:         do 1 i=3,k
17:       f(i)=x*f(i-2)/float(i-2)
18:     1 q(i)=q(i-2)+2.*f(i)
19:       p=1.-q(k)
20:       return
21:       end
```

chi2prob(x,k,p) is based on snprob(x,p), which is shown on p.89. Given x and k, the probability p is obtained. This algorithm is discussed in Hill and Pike (1967), Kennedy and Gentle (1980) and Takeuchi (1989). In the case of very large degrees of freedom, the above algorithm is not good from computational point of view

Chi-Square Percent Points

In order to have percent points of chi-square distribution, chi2perpt(p,k,x) is introduced, which is discussed in Shibata (1981), where the Cornish-Fisher expansion is utilized. For example, see Johnson and Kotz (1970a) and Kotz and Johnson (1982) for the Cornish-Fisher expansion.

chi2perpt(p,k,x)

```
1:       subroutine chi2perpt(p,k,x)
2: c
3: c  Input:
4: c    p:  Probability
5: c    k:  Degree of freedom
6: c  Output:
7: c    x:  chi^2(k) Percent Point corresponding to p,
8: c        i.e., Prob(X<x)=p
9: c
10:       p=1.-p
11:       call snperpt(p,z)
12:       g1=z*sqrt(2.*k)
13:       g2=(2./3.)*(z*z-1.)
14:       g3=z*(z*z-7.)/(9.*sqrt(2.*k))
15:       g4=-2.*(3.*z*z*z*z+7.*z*z-16.)/(405.*k)
16:       x=float(k)+g1+g2+g3+g4
17:       return
18:       end
```

In Line 11, chi2perpt(p,k,x) is based on snperpt(p,x), which is shown on p.90. when p is close to one and k is large, it is known that the approximation above is good, e.g., only 0.18% error when p is 0.99 and k is 5.

Distribution of $(n-1)S^2/\sigma^2$

Suppose that X_1, X_2, $\cdots$, X_n are mutually independently, identically and normally distributed with mean μ and variance σ^2. Define $S^2 = \sum_{i=1}^{n}(X_i - \overline{X})^2/(n-1)$ and

$\overline{X} = (1/n)\sum_{i=1}^{n} X_i$. In Example 1.16 on p.42, $(n-1)S^2/\sigma^2 \sim \chi^2(n-1)$ is utilized without any proof. In this section, we want to prove $(n-1)S^2/\sigma^2 \sim \chi^2(n-1)$. See Hogg and Craig (1995, p.214) for this proof.

Consider the following transformation:

$$Y_1 = \overline{X}, \quad Y_2 = X_2 - \overline{X}, \quad Y_3 = X_3 - \overline{X}, \quad \cdots, \quad Y_n = X_n - \overline{X}.$$

The corresponding inverse transformation is given by:

$$\begin{aligned} x_1 &= y_1 - y_2 - y_3 - \cdots - y_n, \\ x_2 &= y_1 + y_2, \\ x_3 &= y_1 + y_3, \\ &\vdots \\ x_n &= y_1 + y_n. \end{aligned}$$

The Jacobian of the transformation above is:

$$J = \begin{vmatrix} \frac{\partial x_1}{\partial y_1} & \frac{\partial x_1}{\partial y_2} & \cdots & \frac{\partial x_1}{\partial y_n} \\ \frac{\partial x_2}{\partial y_1} & \frac{\partial x_2}{\partial y_2} & \cdots & \frac{\partial x_2}{\partial y_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial x_n}{\partial y_1} & \frac{\partial x_n}{\partial y_2} & \cdots & \frac{\partial x_n}{\partial y_n} \end{vmatrix} = \begin{vmatrix} 1 & -1 & \cdots & -1 \\ 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \cdots & 1 \end{vmatrix} = n.$$

Since $\sum_{i=1}^{n}(x_i-\mu)^2 = \sum_{i=1}^{n}(x_i-\overline{x})^2 + n(\overline{x}-\mu)^2$ because $2(\overline{x}-\mu)\sum_{i=1}^{n}(x_i-\overline{x}) = 0$, The joint density of $X_1, X_2, \cdots, X_n$, denoted by $f_x(x_1, x_2, \cdots, x_n)$, can be written as:

$$\begin{aligned} f_x(x_1, x_2, \cdots, x_n) &= \prod_{i=1}^{n} f(x_i) = \prod_{i=1}^{n} (2\pi\sigma^2)^{-1/2} \exp\Big(-\frac{1}{2\sigma^2}(x_i-\mu)^2\Big) \\ &= (2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2\Big) \\ &= (2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{1}{2\sigma^2}\Big(\sum_{i=1}^{n}(x_i-\overline{x})^2 + n(\overline{x}-\mu)^2\Big)\Big). \end{aligned}$$

Noting that $y_1 = \overline{x}$ and $x_1 - \overline{x} = -y_2 - y_3 - \cdots - y_n$, we find that the joint distribution of $Y_1, Y_2, \cdots, Y_n$, denoted by $f_y(y_1, y_2, \cdots, y_n)$, is represented as:

$$\begin{aligned} f_y(y_1, y_2, \cdots, y_n) &= |J| f_x(y_1 - y_2 - \cdots - y_n, y_1 + y_2, y_1 + y_3, \cdots, y_1 + y_n) \\ &= n(2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{1}{2\sigma^2}\Big((-y_2 - y_3 - \cdots - y_n)^2 + \sum_{i=2}^{n} y_i^2 + n(y_1-\mu)^2\Big)\Big) \end{aligned}$$

$$
\begin{aligned}
&= n(2\pi\sigma^2)^{-n/2}\exp\Big(-\frac{1}{2\sigma^2}((-y_2-y_3-\cdots-y_n)^2+\sum_{i=2}^{n}y_i^2+n(y_1-\mu)^2)\Big)\\
&= \sqrt{n}(2\pi\sigma^2)^{-(n-1)/2}\exp\Big(-\frac{1}{2\sigma^2}((-y_2-y_3-\cdots-y_n)^2+\sum_{i=2}^{n}y_i^2)\Big)\\
&\quad\times(2\pi\sigma^2/n)^{-1/2}\exp\Big(-\frac{1}{2\sigma^2/n}(y_1-\mu)^2\Big)\\
&= f_2(y_2,y_3,\cdots,y_n)f_1(y_1),
\end{aligned}
$$

where $f_2(y_2,y_3,\cdots,y_n)$ is denoted by the joint density of Y_2, Y_3, $\cdots$, Y_n and $f_1(y_1)$ is the marginal density of Y_1, which implies that $n(Y_1-\mu)^2/\sigma^2$ is independent of $\Big((-Y_2-Y_3-\cdots-Y_n)^2+\sum_{i=2}^{n}Y_i^2\Big)/\sigma^2$. In other words, substituting X_i into Y_i for $i=1,2,\cdots,n$, we obtain the result that $n(\overline{X}-\mu)/\sigma^2 \equiv U_1$ is independent of $\sum_{i=1}^{n}(X_i-\overline{X})^2/\sigma^2 \equiv U_2$. From $U \equiv \sum_{i=1}^{n}(X_i-\mu)^2/\sigma^2 = \sum_{i=1}^{n}(X_i-\overline{X})^2/\sigma^2 + n(\overline{X}-\mu)/\sigma^2$, we have $U = U_2+U_1$. Moreover, from $U \sim \chi^2(n)$, $U_1 \sim \chi^2(1)$, and independence between U_1 and U_2, we have:

$$\mathrm{E}(e^{\theta U}) = \mathrm{E}(e^{\theta(U_1+U_2)}) = \mathrm{E}(e^{\theta(U_1)})\,\mathrm{E}(e^{\theta(U_2)}).$$

Using $\mathrm{E}(e^{\theta U}) = (1-2\theta)^{-n/2}$ and $\mathrm{E}(e^{\theta U_1}) = (1-2\theta)^{-1/2}$, the moment-generating function of U_2 is:

$$\mathrm{E}(e^{\theta(U_2)}) = \frac{(1-2\theta)^{-n/2}}{(1-2\theta)^{-1/2}} = (1-2\theta)^{-(n-1)/2},$$

which is equivalent to the moment-generating function of $\chi^2(n-1)$ random variable. Thus, $U_2 = \sum_{i=1}^{n}(X_i-\overline{X})^2/\sigma^2 = (n-1)S^2/\sigma^2 \sim \chi^2(n-1)$ is obtained. Summarizing, we have shown in this section that $n(\overline{X}-\mu)/\sigma^2$ is independent of $(n-1)S^2/\sigma^2$ and $(n-1)S^2/\sigma^2 \sim \chi^2(n-1)$.

2.2.9 F Distribution: $F(m,n)$

The F distribution with m and n degrees of freedom, denoted by $F(m,n)$, is represented as:

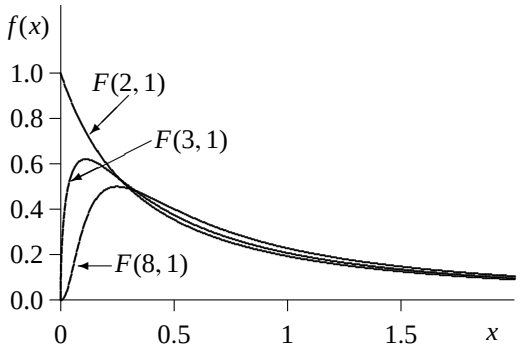
$$
f(x) = \begin{cases} \dfrac{\Gamma(\frac{m+n}{2})}{\Gamma(\frac{m}{2})\Gamma(\frac{n}{2})}\Big(\dfrac{m}{n}\Big)^{\frac{m}{2}}x^{\frac{m}{2}-1}\Big(1+\dfrac{m}{n}x\Big)^{-\frac{m+n}{2}}, & \text{for } 0<x<\infty,\\ 0, & \text{otherwise,}\end{cases}
$$

where m and n are positive integers. The F distributions are shown in Figure 2.9, in which the cases of $(m,n)=(2,1)$, $(3,1)$, $(8,1)$ are taken as some examples.

Mean and variance are given by:

$$
\begin{aligned}
\mathrm{E}(X) &= \frac{n}{n-2}, \qquad \text{for } n>2,\\
\mathrm{V}(X) &= \frac{2n^2(m+n-2)}{m(n-2)^2(n-4)}, \qquad \text{for } n>4.
\end{aligned}
$$

Figure 2.9: F Distribution



The moment-generating function of F distribution does not exist.

One F random variable is derived from two chi-square random variables. Suppose that U and V are independently distributed as chi-square random variables, i.e., $U \sim \chi^2(m)$ and $V \sim \chi^2(n)$. Then, it is shown that $X = \dfrac{U/m}{V/n}$ has a F distribution with (m, n) degrees of freedom.

To show this, consider the following transformation:

$$X = \frac{U/m}{V/n}, \qquad Y = V,$$

which is a one-to-one transformation from (U, V) to (X, Y). Because we have $u = (m/n)xy$ and $v = y$, the Jacobian is:

$$J = \begin{vmatrix} \dfrac{\partial u}{\partial x} & \dfrac{\partial u}{\partial y} \\ \dfrac{\partial v}{\partial x} & \dfrac{\partial v}{\partial y} \end{vmatrix} = \begin{vmatrix} (m/n)y & (m/n)x \\ 0 & 1 \end{vmatrix} = \frac{m}{n}y.$$

Let $f_{uv}(\cdot,\cdot)$, $f_u(\cdot)$ and $f_v(\cdot)$ be the joint density of U and V, the marginal density of U and the marginal density of V, respectively. The joint density of X and Y is given by:

$$\begin{aligned} f_{xy}(x,y) &= |J| f_{uv}(\frac{m}{n}xy, y) = |J| f_u(\frac{m}{n}xy) f_v(y) \\ &= \frac{m}{n}y \frac{1}{2^{m/2}\Gamma(\frac{m}{2})}\Big(\frac{m}{n}xy\Big)^{\frac{m}{2}-1} e^{-\frac{1}{2}\frac{m}{n}xy} \frac{1}{2^{n/2}\Gamma(\frac{n}{2})} y^{\frac{n}{2}-1} e^{-\frac{1}{2}y} \\ &= \frac{(m/n)^{m/2} x^{m/2-1}}{2^{(m+n)/2}\Gamma(\frac{m}{2})\Gamma(\frac{n}{2})} y^{(m+n)/2-1} \exp\Big(-\frac{1}{2}(\frac{m}{n}x+1)y\Big). \end{aligned}$$

Integrating $f_{xy}(x,y)$ with respect to y, the marginal density of X, $f(x)$, is obtained as follows:

$$f(x) = \int_0^\infty f_{xy}(x,y)\, \mathrm{d}y$$

$$
\begin{aligned}
&= \frac{(m/n)^{m/2}x^{m/2-1}}{2^{(m+n)/2}\Gamma(\frac{m}{2})\Gamma(\frac{n}{2})}\int_0^\infty y^{(m+n)/2-1}\exp\Big(-\frac{1}{2}(\frac{m}{n}x+1)y\Big)\,\mathrm{d}y \\
&= \frac{(m/n)^{m/2}x^{m/2-1}(2(\frac{m}{n}x+1)^{-1})^{(m+n)/2}\Gamma(\frac{m+n}{2})}{2^{(m+n)/2}\Gamma(\frac{m}{2})\Gamma(\frac{n}{2})} \\
&\quad\times\int_0^\infty \frac{1}{\Big(2(\frac{m}{n}x+1)^{-1}\Big)^{(m+n)/2}\Gamma(\frac{m+n}{2})}y^{(m+n)/2-1}\exp\Big(-\frac{1}{2}(\frac{m}{n}x+1)y\Big)\,\mathrm{d}y \\
&= \frac{\Gamma(\frac{m+n}{2})}{\Gamma(\frac{m}{2})\Gamma(\frac{n}{2})}\frac{(m/n)^{m/2}x^{m/2-1}}{(\frac{m}{n}x+1)^{(m+n)/2}}.
\end{aligned}
$$

Note that the density function in the integration is a gamma distribution with parameters $\alpha = (m+n)/2$ and $\beta = 2(\frac{m}{n}x+1)^{-1}$. Thus, the F distribution with m and n degrees of freedom is derived. Therefore, using two independent chi-square random variables, the F random number generator is given by `frnd(ix,iy,m,n,rn)`.

`frnd(ix,iy,m,n,rn)`

```
      subroutine frnd(ix,iy,m,n,rn)
c
c  Use "frnd(ix,iy,m,n,rn)"
c  together with "chi2rnd(ix,iy,k,rn)",
c                "snrnd(ix,iy,rn)"
c           and  "urnd(ix,iy,rn)".
c
c  Input:
c    ix, iy:  Seeds
c    m, n:  Degrees of Freedom
c  Output:
c    rn: F Random Draw
c        with m and n Degrees of Freedom
c
       call chi2rnd(ix,iy,m,rn1)
       call chi2rnd(ix,iy,n,rn2)
       rn=(rn1/float(m))/(rn2/float(n))
       return
       end
```

Note that `frnd(ix,iy,m,n,rn)` should be used together with `urnd(ix,iy,rn)` on p.83, `snrnd(ix,iy,rn)` on p.85 and `chi2rnd(ix,iy,k,rn)` on p.102.

Because of computational efficiency, `chi2rnd(ix,iy,k,rn)` should be replaced by `chi2rnd6(ix,iy,k,rn)` on p.215. The latter is preferred to the former from computational point of view.

Finally, we point out as follows. The F distribution is derived from the beta distribution discussed in Section 2.2.7. Suppose that Y has a beta distribution with parameters $m/2$ and $n/2$. Define $X = \dfrac{Y/m}{(1-Y)/n}$. Then, X has a F distribution with parameters m and n, i.e., $F(m,n)$.

2.2.10 t Distribution: $t(k)$

The t distribution (or Student's t distribution) with k degrees of freedom, denoted by $t(k)$, is given by:

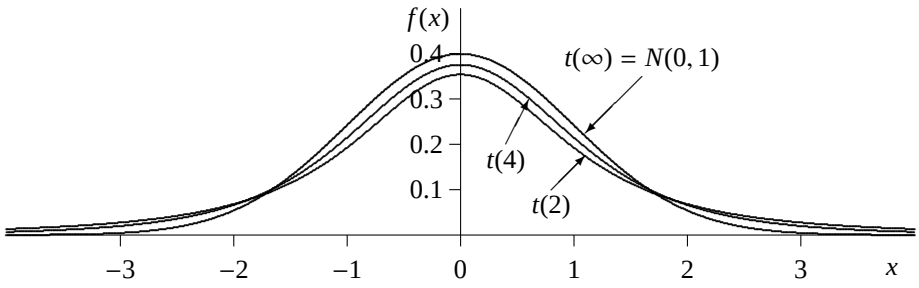
$$f(x) = \frac{\Gamma(\frac{k+1}{2})}{\Gamma(\frac{k}{2})} \frac{1}{\sqrt{k\pi}} \Big(1 + \frac{x^2}{k}\Big)^{-\frac{k+1}{2}},$$

for $-\infty < x < \infty$, where k does not have to be an integer but conventionally it is a positive integer. The t distributions with $k = 2, 4$ are shown in Figure 2.10, which are compared with $N(0,1)$. When k is small, the t distribution has fat tails. The t distribution with $k = 1$ is equivalent to the Cauchy distribution (see Section 2.3.5 for the Cauchy distribution). As k goes to infinity, the t distribution approaches the standard normal distribution, i.e., $t(\infty) = N(0,1)$, which is easily shown by using the definition of e on p.12, i.e.,

$$\Big(1 + \frac{x^2}{k}\Big)^{-\frac{k+1}{2}} = \Big(1 + \frac{1}{h}\Big)^{-\frac{hx^2+1}{2}} = \Big((1 + \frac{1}{h})^h\Big)^{-\frac{1}{2}x^2} \Big(1 + \frac{1}{h}\Big)^{-\frac{1}{2}} \longrightarrow e^{-\frac{1}{2}x^2},$$

where $h = k/x^2$ is set and h goes to infinity (equivalently, k goes to infinity). Thus, a kernel of the t distribution is equivalent to that of the standard normal distribution. Therefore, it is shown that as k is large the t distribution approaches the standard normal distribution.

Figure 2.10: t Distribution: $t(k)$



Mean and variance of the t distribution with k degrees of freedom are obtained as:

$$\mathrm{E}(X) = 0, \qquad \text{for } k > 1,$$
$$\mathrm{V}(X) = \frac{k}{k-2}, \qquad \text{for } k > 2.$$

In the case of the t distribution, the moment-generating function does not exist, because all the moments do not necessarily exist. For the t random variable X, we have the fact that $\mathrm{E}(X^p)$ exists when p is less than k. Therefore, all the moments exist only when k is infinity.

One t random variable is obtained from chi-square and standard normal random variables. Suppose that $Z \sim N(0,1)$ is independent of $U \sim \chi^2(k)$. Then, $X = Z/\sqrt{U/k}$

has a t distribution with k degrees of freedom. This result is shown as follows. Consider the following transformation:

$$X = \frac{Z}{\sqrt{U/k}}, \qquad Y = U,$$

which is a one-to-one transformation from (Z, U) to (X, Y). Because we have $z = x\sqrt{y/k}$ and $u = y$, the Jacobian is given by:

$$J = \begin{vmatrix} \frac{\partial z}{\partial x} & \frac{\partial z}{\partial y} \\ \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \end{vmatrix} = \begin{vmatrix} \sqrt{y/k} & \frac{1}{2}x/\sqrt{ky} \\ 0 & 1 \end{vmatrix} = \sqrt{\frac{y}{k}}.$$

Let $f_{xy}(\cdot,\cdot)$, $f_{zu}(\cdot,\cdot)$, $f_z(\cdot)$ and $f_u(\cdot)$ denote the joint density of X and Y, the joint density of Z and U, the marginal density of Z (i.e., standard normal distribution) and the marginal density of U (i.e., chi-square distribution with k degrees of freedom), respectively. The joint density of X and Y is derived as follows:

$$\begin{aligned} f_{xy}(x,y) &= |J| f_{zu}(x\sqrt{y/k}, y) = |J| f_z(x\sqrt{y/k}) f_u(y) \\ &= \sqrt{\frac{y}{k}} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2k}x^2 y} \frac{1}{2^{k/2}\Gamma(\frac{k}{2})} y^{\frac{k}{2}-1} e^{-\frac{1}{2}y} \\ &= \frac{1}{\sqrt{2\pi k}\Gamma(\frac{k}{2})2^{k/2}} y^{(k+1)/2-1} \exp\left(-\frac{1}{2}(1+\frac{x^2}{k})y\right). \end{aligned}$$

The second equality comes from independence between Z and U by assumption. Integrating $f_{xy}(x,y)$ with respect to y, the marginal density of X, $f(x)$, is obtained as:

$$\begin{aligned} f(x) &= \int_0^\infty f_{xy}(x,y)\,\mathrm{d}y = \int_0^\infty \frac{1}{\sqrt{2\pi k}\Gamma(\frac{k}{2})2^{k/2}} y^{(k+1)/2-1} \exp\Big(-\frac{1}{2}(1+\frac{x^2}{k})y\Big)\,\mathrm{d}y \\ &= \frac{\Big(2(1+x^2/k)^{-1}\Big)^{(k+1)/2}\Gamma(\frac{k+1}{2})}{\sqrt{2\pi k}\Gamma(\frac{k}{2})2^{k/2}} \\ &\quad \times \int_0^\infty \frac{1}{\Big(2(1+x^2/k)^{-1}\Big)^{(k+1)/2}\Gamma(\frac{k+1}{2})} y^{(k+1)/2-1} \exp\Big(-\frac{1}{2}(1+\frac{x^2}{k})y\Big)\,\mathrm{d}y \\ &= \frac{\Gamma(\frac{k+1}{2})}{\Gamma(\frac{k}{2})} \frac{1}{\sqrt{k\pi}} \Big(1+\frac{x^2}{k}\Big)^{-\frac{k+1}{2}}. \end{aligned}$$

Note that the density function in the integration of the third line corresponds to the gamma distribution with parameters $\alpha = (k+1)/2$ and $\beta = 2(1+x^2/k)^{-1}$. Thus, the t distribution with k degrees of freedom is obtained and the random number generator from the t distribution is shown in `trnd(ix,iy,k,rn)`.

trnd(ix,iy,k,rn)

```
      subroutine trnd(ix,iy,k,rn)
c
c  Use "trnd(ix,iy,k,rn)"
c  together with "chi2rnd(ix,iy,k,rn)",
c                "snrnd(ix,iy,rn)"
c           and  "urnd(ix,iy,rn)".
c
c  Input:
c    ix, iy:  Seeds
c    k:  Degree of Freedom
c  Output:
c    rn: t Random Draw
c        with k Degrees of Freedom
c
       call snrnd(ix,iy,rn1)
       call chi2rnd(ix,iy,k,rn2)
       rn=rn1/sqrt(rn2/k)
       return
       end
```

Note that `trnd(ix,iy,k,rn)` should be used simultaneously with `urnd(ix,iy,rn)` on p.83, `snrnd(ix,iy,rn)` on p.85 and `chi2rnd(ix,iy,k,rn)` on p.102. As mentioned in `frnd(ix,iy,m,n,rn)` of Section 2.2.9, `chi2rnd6(ix,iy,k,rn)` on p.215 should be used, rather than `chi2rnd(ix,iy,k,rn)`, from computational point of view.

Suppose that Y has the t distribution with n degrees of freedom. Define $X = Y^2$. Then, X has the F distribution with $(1, n)$ degrees of freedom. Thus, the squared $t(n)$ random variable yields the $F(1, n)$ random variable.

Marsaglia (1984) gives a very fast algorithm for generating t random draws, which is based on a transformed acceptance/rejection method (see Section 3.2 for rejection sampling).

t Probabilities

Let $Q(\theta; k)$ be $P(X < x)$, where $X \sim t(k)$ and $\theta = x/\sqrt{k}$. As shown in Chi-Square Probabilities on p.105, using the integration by parts, we have the following recursive algorithm:

$$f(\theta; i) = \frac{i-1}{i-2}\cos^2\theta\ f(\theta; i-2),$$
$$Q(\theta; i) = Q(\theta; i-2) + \frac{1}{i-2} f(\theta; i-2)$$

for $i = 3, 4, \cdots, k$, where the initial values are given by:

$$f(x; 1) = \frac{\cos\theta \sin\theta}{\pi}, \qquad f(x; 2) = \frac{\cos^2\theta \sin\theta}{2},$$

$$Q(x;1) = \frac{1}{2} + \frac{\theta}{\pi}, \qquad Q(x;2) = \frac{1}{2} + \frac{\sin\theta}{2}.$$

The source code `tprob(x,k,p)` is shown as follows.

tprob(x,k,p)

```
      subroutine tprob(x,k,p)
      dimension f(1000),q(1000)
c
c  Input:
c    x:  t(k) Percent Point
c    k:  Degree of freedom (less than or equal to 1000)
c  Output:
c    p:  Probability corresponding to x, i.e., Prob(X<x)=p
c
      pi= 3.1415926535897932385
      theta=atan( x/sqrt(float(k)) )
      f(1)=cos(theta)*sin(theta)/pi
      f(2)=cos(theta)*cos(theta)*sin(theta)/2.
      q(1)=.5+theta/pi
      q(2)=.5+sin(theta)/2.
        do 1 i=3,k
      f(i)=( float(i-1)/float(i-2) )*(cos(theta)**2)*f(i-2)
    1 q(i)=q(i-2)+f(i-2)/float(i-2)
      p=q(k)
      return
      end
```

p corresponds to $P(X < x)$. When the degree of freedom, k, is not very large, the above algorithm performs good. See Takeuchi (1989).

t Percent Points

To obtain the percent point of the *t* distribution, Takeuchi (1989, p.523) introduced `tperpt(p,k,x)`, where the Cornish-Fisher expansion is utilized (see Johnson and Kotz (1970a) and Kotz and Johnson (1982) for the Cornish-Fisher expansion).

tperpt(p,k,x)

```
      subroutine tperpt(p,k,x)
c
c  Input:
c    p:  Probability
c    k:  Degree of freedom
c  Output:
c    x:  t(k) Percent Point corresponding to p,
c        i.e., Prob(X<x)=p
c
      g1(z)=z*(    (z**2 )+    1.)/4.
      g2(z)=z*( 5.*(z**4 )+   16.*(z**2)+   3.)/96.
      g3(z)=z*( 3.*(z**6 )+   19.*(z**4)
```

```
13:       &     +   17.*(z**2 )-   15.)/384.
14:        g4(z)=z*(79.*(z**8 )+  776.*(z**6)+1482.*(z**4)
15:       &     - 1920.*(z**2 )-  945.)/92160.
16:        g5(z)=z*(27.*(z**10)+  339.*(z**8)
17:       &     +  930.*(z**6 )- 1782.*(z**4)
18:       &     -  765.*(z**2 )-17955.)/368640.
19:        call snperpt(p,z)
20:        x=z+g1(z)/ float(k)
21:       &   +g2(z)/(float(k)**2)
22:       &   +g3(z)/(float(k)**3)
23:       &   +g4(z)/(float(k)**4)
24:       &   +g5(z)/(float(k)**5)
25:        return
26:        end
```

In Line 19, `tperpt(p,k,x)` is based on `snperpt(p,x)`, which is shown on p.90. The approximation up to the term of `g2(z)` gives us 1.7% error when `p` is 0.95 and `k` is 3. Thus, the approximation is quite good. Also, see Johnson and Kotz (1970b) for approximation of t distribution.

Distribution of $\sqrt{n}(\overline{X}-\mu)/S$

Suppose that X_1, X_2, $\cdots$, X_n are mutually independently, identically and normally distributed with mean μ and variance σ^2. We have stated in Section 1.7.6, p.48 that $\sqrt{n}(\overline{X}-\mu)/S$ has a t distribution with $n-1$ degrees of freedom.

On p.106 in Section 2.2.8, we have shown that $n(\overline{X}-\mu)/\sigma^2$ is independent of $(n-1)S^2/\sigma^2$ and $(n-1)S^2/\sigma^2 \sim \chi^2(n-1)$.

In this section, we have shown $T = Z/\sqrt{U/k} \sim t(k)$ when $Z \sim N(0,1)$, $U \sim \chi^2(k)$, and Z is independent of U. Now, set $k = n-1$ and define Z and U as:

$$Z = \frac{\overline{X}-\mu}{\sqrt{\sigma^2/n}} \sim N(0,1), \qquad U = \frac{(n-1)S^2}{\sigma^2} \sim \chi^2(n-1).$$

Z is independent of U, because Z^2 is independent of U as shown on p.106 in Section 2.2.8. Then, we obtain:

$$T = \frac{Z}{\sqrt{U/(n-1)}} = \frac{\dfrac{\overline{X}-\mu}{\sqrt{\sigma^2/n}}}{\sqrt{\dfrac{(n-1)S^2}{\sigma^2} \Big/ (n-1)}} = \frac{\overline{X}-\mu}{\sqrt{S^2/n}} \sim t(n-1).$$

Thus, it is shown that $\sqrt{n}(\overline{X}-\mu)/S$ has the $t(n-1)$ distribution.

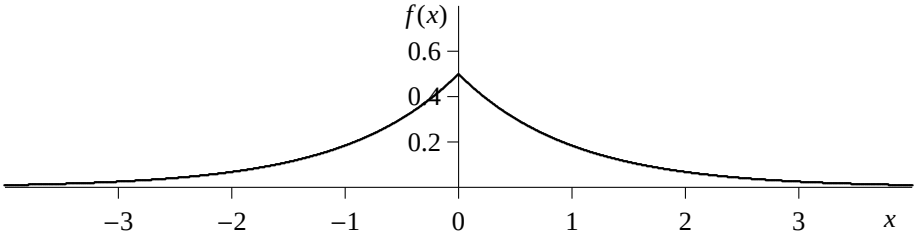
2.2.11 Double Exponential Distribution (LaPlace Distribution)

The double exponential distribution (or the LaPlace distribution) with parameters α and β is represented as follows:

$$f(x) = \frac{1}{2\beta} e^{-\frac{|x-\alpha|}{\beta}},$$

for $-\infty < x < \infty$, $-\infty < \alpha < \infty$ and $\beta > 0$, where α and β are known as a location parameter and a scale parameter, respectively. The double exponential distribution (LaPlace Distribution) with parameters $\alpha = 0$ and $\beta = 1$ are drawn in Figure 2.11.

Figure 2.11: Double Exponential Distribution (Laplace Distribution)
The case of $\alpha = 0$ and $\beta = 1$



Mean, variance and the moment-generating function are given by:

$$\mathrm{E}(X) = \alpha, \qquad \mathrm{V}(X) = 2\beta^2, \qquad \phi(\theta) = \frac{e^{\alpha\theta}}{1 - \beta^2\theta^2}.$$

The double exponential random variable is obtained from two independent exponential random variables. Suppose that $X_1 \sim \chi^2(2)$ and $X_2 \sim \chi^2(2)$ are stochastically independent. That is, both X_1 and X_2 have the exponential distribution with parameter $\beta = 2$ (see Section 2.2.4 for the exponential distribution). Define $Y = (X_1 - X_2)/2$. Then, Y has a double exponential distribution with $\alpha = 0$ and $\beta = 1$.

We show that Y has a double exponential distribution. Let $f_i(x)$ be the probability density function of X_i, which is given by:

$$f_i(x) = \frac{1}{2} \exp\Big(-\frac{1}{2}x\Big),$$

for $i = 1, 2$. Therefore, the joint density of X_1 and X_2, $f_x(x_1, x_2)$, is given by:

$$f_x(x_1, x_2) = f_1(x_1) f_2(x_2) = \frac{1}{4} \exp\Big(-\frac{1}{2}(x_1 + x_2)\Big),$$

for $0 < x_1 < \infty$ and $0 < x_2 < \infty$.

Consider the following transformation:

$$Y = \frac{1}{2}(X_1 - X_2), \qquad W = X_2,$$

i.e., $x_1 = 2y + w$ and $x_2 = w$, where $-2y < w$, $0 < w$ and $-\infty < y < \infty$. Therefore, the Jacobian of the transformation is:

$$J = \begin{vmatrix} \frac{\partial x_1}{\partial y} & \frac{\partial x_1}{\partial w} \\ \frac{\partial x_2}{\partial y} & \frac{\partial x_2}{\partial w} \end{vmatrix} = \begin{vmatrix} 2 & 1 \\ 0 & 1 \end{vmatrix} = 2.$$

The joint density of Y and W, $f_{yw}(y,w)$, is:

$$f_{yw}(y,w) = |J| f_x(2y+w, w) = |J| f_1(2y+w) f_2(w) = \frac{1}{2} e^{-(y+w)},$$

for $-2y < w$, $0 < w$ and $-\infty < y < \infty$. Integrating $f_{yw}(y,w)$ with respect to w, the marginal density of Y, $f_y(y)$, is:

$$f_y(y) = \begin{cases} \int_{-2y}^{\infty} \frac{1}{2} e^{-(y+w)} \, dw = \frac{1}{2} e^{y}, & -\infty < y < 0, \\ \int_{0}^{\infty} \frac{1}{2} e^{-(y+w)} \, dw = \frac{1}{2} e^{-y}, & 0 \le y < \infty, \end{cases}$$

That is, the marginal density function of Y is given by:

$$f_y(y) = \frac{1}{2} e^{-|y|},$$

for $-\infty < y < \infty$. See Hogg and Craig (1995, pp.175 – 176) for discussion about the double exponential density function.

Moreover, defining $X = \alpha + \beta Y$, X is distributed as $f(x)$, which is a double exponential random variable with parameters α and β, as indicated above. Thus, X is obtained from two independent exponential random variables. The source code is given by `dexprnd(ix,iy,alpha,beta,rn)`.

`dexprnd(ix,iy,alpha,beta,rn)`

```
1:       subroutine dexprnd(ix,iy,alpha,beta,rn)
2: C
3: C  Use "dexprnd(ix,iy,alpha,beta,rn)"
4: C  together with "exprnd(ix,iy,beta,rn)"
5: C            and  "urnd(ix,iy,rn)".
6: C
7: C  Input:
8: C    ix, iy:  Seeds
9: C    alpha:  Location Parameter
10: C   beta:   Scale Parameter
11: C Output:
12: C   rn: Double Exponential Random Draw
13: C
14:       call exprnd(ix,iy,2.0,rn1)
15:       call exprnd(ix,iy,2.0,rn2)
```

```
16:         rn=alpha+beta*0.5*(rn1-rn2)
17:         return
18:         end
```

Note that `dexprnd(ix,iy,alpha,beta,rn)` is utilized with `urnd(ix,iy,rn)` on p.83 and `exprnd(ix,iy,beta,rn)` on p.94.

2.2.12 Noncentral Chi-Square Distribution: $\chi^2(k;\alpha)$

The noncentral chi-square distribution with k degrees of freedom and noncentrality parameter α, denoted by $\chi^2(k;\alpha)$, is written as:

$$f(x) = \begin{cases} \dfrac{e^{-\frac{1}{2}(x+\alpha)}}{2^{\frac{k}{2}}} \displaystyle\sum_{j=0}^{\infty} \frac{x^{\frac{k}{2}+j-1}\alpha^j}{\Gamma(\frac{k}{2}+j)2^{2j}j!}, & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

for $\alpha \geq 0$, where k is a positive integer. α is called the **noncentrality parameter**. The noncentral chi-square distribution with noncentrality parameter $\alpha = 0$ is equivalent the chi-square distribution discussed in Section 2.2.8. In this sense, the chi-square distribution discussed in Section 2.2.8 is called the central chi-square distribution.

Mean, variance and the moment-generating function are:

$$\mathrm{E}(X) = k + \alpha, \qquad \mathrm{V}(X) = 2(k + 2\alpha), \qquad \phi(\theta) = \frac{1}{(1-2\theta)^{k/2}} \exp\Big(\frac{\alpha\theta}{1-2\theta}\Big).$$

The noncentral chi-square random variable is derived from k normal random variables. Let $Z_1, Z_2, \cdots, Z_k$ be mutually stochastically independent random variables. When $Z_i \sim N(\mu_i, \sigma^2)$ for $i = 1, 2, \cdots, k$, it is known that $X = \sum_{i=1}^{k} Z_i^2/\sigma^2$ is distributed as a noncentral chi-square random variable with k degrees of freedom and noncentrality parameter $\alpha = \sum_{i=1}^{k} \mu_i^2/\sigma^2$.

To show this fact, consider the moment-generating function of X, $\phi_x(\theta)$, as follows:

$$\phi_x(\theta) = \mathrm{E}(e^{\theta X}) = \mathrm{E}(e^{\theta \sum_{i=1}^{k} Z_i^2/\sigma^2}) = \prod_{i=1}^{k} \mathrm{E}(e^{\theta Z_i^2/\sigma^2}).$$

$\mathrm{E}(e^{\theta Z_i^2/\sigma^2})$ in $\phi_x(\theta)$ is rewritten as follows:

$$\begin{aligned} \mathrm{E}(e^{\theta Z_i^2/\sigma^2}) &= \int_{-\infty}^{\infty} \exp(\theta\frac{z_i^2}{\sigma^2}) \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{1}{2\sigma^2}(z_i-\mu_i)^2\Big)\, \mathrm{d}z_i \\ &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(\theta\frac{z_i^2}{\sigma^2} - \frac{1}{2\sigma^2}(z_i-\mu_i)^2\Big)\, \mathrm{d}z_i \\ &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(\frac{\theta\mu_i^2}{\sigma^2(1-2\theta)} - \frac{1}{2\sigma^2/(1-2\theta)}(z_i - \frac{\mu_i}{1-2\theta})^2\Big)\, \mathrm{d}z_i \end{aligned}$$

$$= \frac{1}{(1-2\theta)^{-1/2}} \exp\Big(\frac{\theta\mu_i^2/\sigma^2}{1-2\theta}\Big)$$
$$\times \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2/(1-2\theta)}} \exp\Big(-\frac{1}{2\sigma^2/(1-2\theta)}(z_i - \frac{\mu_i}{1-2\theta})^2\Big)\, dz_i$$
$$= \frac{1}{(1-2t)^{-1/2}} \exp\Big(\frac{\theta\mu_i^2/\sigma^2}{1-2\theta}\Big).$$

Note that the integration in the fifth line is equal to one, because the function in the integration corresponds to the normal distribution function with mean $\mu_i/(1-2\theta)$ and variance $\sigma^2/(1-2\theta)$. Accordingly, the moment-generating function of X is given by:

$$\phi_x(\theta) = \prod_{i=1}^{k} \mathrm{E}(e^{\theta Z_i^2/\sigma^2}) = \frac{1}{(1-2\theta)^{-k/2}} \exp\Big(\frac{\theta\sum_{i=1}^{k}\mu_i^2/\sigma^2}{1-2\theta}\Big),$$

which is equivalent to the noncentral chi-square distribution with k degrees of freedom and noncentrality parameter $\alpha = \sum_{i=1}^{k}\mu_i^2/\sigma^2$. See Hogg and Craig (1995, pp.458 – 460) for derivation of the noncentral chi-square distribution.

In other words, when $Z_i \sim N(\sqrt{\alpha/k}, 1)$, $X = \sum_{i=1}^{k} Z_i^2$ is distributed as a noncentral chi-square random variable with k degrees of freedom and noncentrality parameter α, i.e., $\chi^2(k;\alpha)$. Therefore, the source code of the noncentral chi-square random number generator with k degrees of freedom and noncentrality parameter α is given by `nchi2rnd(ix,iy,k,alpha,rn)`.

`nchi2rnd(ix,iy,k,alpha,rn)`

```
      subroutine nchi2rnd(ix,iy,k,alpha,rn)
C
C  Use "nchi2rnd(ix,iy,k,alpha,rn)"
C  together with "snrnd(ix,iy,rn)"
C           and  "urnd(ix,iy,rn)".
C
C  Input:
C    ix, iy:  Seeds
C    k:  Degree of Freedom
C    alpha:  Noncentrality Parameter
C  Output:
C    rn: Chi-Square Random Draw
C        with k Degrees of Freedom
C        and Noncentrality Parameter alpha
C
       rn=0.0
        do 1 i=1,k
       call snrnd(ix,iy,rn1)
       rn1=rn1+sqrt(alpha/float(k))
     1 rn=rn+rn1*rn1
       return
       end
```

Note that `nchi2rnd(ix,iy,k,alpha,rn)` utilizes `urnd(ix,iy,rn)` on p.83 and `snrnd(ix,iy,rn)` on p.85.

2.2.13 Noncentral F Distribution: $F(m, n; \alpha)$

The noncentral F distribution with (m, n) degrees of freedom and noncentrality parameter α, denoted by $F(m, n; \alpha)$, is represented as follows:

$$f(x) = \begin{cases} \displaystyle\sum_{j=0}^{\infty} \frac{\Gamma(\frac{2j+m+n}{2})\left(\frac{m}{n}\right)^{\frac{2j+m}{2}} x^{\frac{2j+m}{2}-1} e^{-\frac{\alpha}{2}}\left(\frac{\alpha}{2}\right)}{\Gamma(\frac{2j+m}{2})\Gamma(\frac{n}{2}) j!\left(1+\frac{m}{n}x\right)^{\frac{2j+m+n}{2}}}, & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

where $\alpha \geq 0$ should be taken. If $\alpha = 0$, the noncentral F distribution reduces to the F distribution discussed in Section 2.2.9, so-called the central F distribution.

Mean and variance of the noncentral F distribution with m and n degrees of freedom and noncentrality parameter α are given by:

$$\mathrm{E}(X) = \frac{(m+\alpha)n}{(n-2)m}, \qquad \text{for } n > 2,$$
$$\mathrm{V}(X) = \frac{(m+\alpha)^2 + 2(m+\alpha)n^2}{(n-2)(n-4)m^2} - \frac{(m+\alpha)^2 n^2}{(n-2)^2 m^2}, \qquad \text{for } n > 4.$$

The noncentral F distribution is derived as follows. Suppose that $U \sim \chi^2(m; \alpha)$ and $V \sim \chi^2(n)$ are stochastically independent. Define $X = \dfrac{U/m}{V/n}$. Then, X is distributed as a noncentral F random variable with m and n degrees of freedom and noncentrality parameter α. See Hogg and Craig (1995, p.460) for the noncentral F distribution. Therefore, the random number generator is written as `nfrnd(ix,iy,m,n,alpha,rn)`.

`nfrnd(ix,iy,m,n,alpha,rn)`

```
      subroutine nfrnd(ix,iy,m,n,alpha,rn)
C
C  Use "nfrnd(ix,iy,m,n,alpha,rn)"
C  together with "nchi2rnd(ix,iy,k,alpha,rn)",
C                "chi2rnd(ix,iy,k,rn)",
C                "snrnd(ix,iy,rn)"
C           and  "urnd(ix,iy,rn)".
C
C  Input:
C    ix, iy:  Seeds
C    m, n:  Degrees of Freedom
C    alpha: Noncentrality Parameter
C  Output:
C    rn: F Random Draw
C        with m and n Degrees of Freedom
C        and Noncentrality Parameter alpha
C
       call nchi2rnd(ix,iy,m,alpha,rn1)
       call chi2rnd(ix,iy,n,rn2)
       rn=(rn1/m)/(rn2/n)
       return
```

```
22:       end
```

Note that `nfrnd(ix,iy,m,n,alpha,rn)` should be utilized simultaneously with `urnd(ix,iy,rn)` on p.83, `snrnd(ix,iy,rn)` on p.85, `chi2rnd(ix,iy,k,rn)` on p.102 and `nchi2rnd(ix,iy,k,alpha,rn)` on p.119. As mentioned above, from computational point of view, the subroutine `chi2rnd(ix,iy,k,rn)` should be replaced by `chi2rnd6(ix,iy,k,rn)` on p.215.

2.2.14 Noncentral t Distribution: $t(k;\alpha)$

The noncentral t distribution with k degrees of parameter and noncentrality parameter α, denoted by $t(k;\alpha)$, is of the form:

$$f(x) = \frac{e^{-\frac{1}{2}\alpha^2}}{\sqrt{k\pi}\Gamma((\frac{k}{2})} \sum_{j=0}^{\infty} \Gamma\Big(\frac{k+j+1}{2}\Big)\frac{(\alpha x)^j}{j!}\Big(\frac{2}{k}\Big)^{\frac{j}{2}}\Big(1+\frac{x^2}{k}\Big)^{-\frac{k+j+1}{2}},$$

for $-\infty < x < \infty$ and $-\infty < \alpha < \infty$. When $\alpha = 0$, the noncentral t distribution is equivalent to the t distribution shown in Section 2.2.10, so-called the central t distribution. See Hogg and Craig (1995, p.420) for the noncentral t distribution.

The first- and second-moments of the noncentral t random variable X are given by:

$$\mathrm{E}(X) = \frac{k^{1/2}\alpha\Gamma(\frac{k-1}{2})}{2^{1/2}\Gamma(\frac{k}{2})}, \qquad \mathrm{E}(X^2) = (1+\alpha^2)\frac{k}{k-2}.$$

Note that $\mathrm{E}(X^p)$ exists only when $k > p$.

The noncentral t random variable is obtained as follows. Suppose that $W \sim N(\alpha, 1)$ and $U \sim \chi^2(k)$ are independent. Then, $X = W/\sqrt{U/k}$ has a noncentral t distribution with k degrees of freedom and noncentrality parameter α. Thus, the source code for the noncentral t random number generator with `k` degrees of freedom and noncentrality parameter `alpha` is in `ntrnd(ix,iy,k,alpha,rn)`.

`ntrnd(ix,iy,k,alpha,rn)`

```
1:        subroutine ntrnd(ix,iy,k,alpha,rn)
2:  c
3:  c  Use "ntrnd(ix,iy,k,alpha,rn)"
4:  c  together with "chi2rnd(ix,iy,k,rn)",
5:  c                "snrnd(ix,iy,rn)"
6:  c           and  "urnd(ix,iy,rn)".
7:  c
8:  c  Input:
9:  c    ix, iy:  Seeds
10: c    k:  Degree of Freedom
11: c    alpha:  Noncentrality Parameter
12: c  Output:
```

```
13: c     rn: t Random Draw
14: c         with k Degrees of Freedom
15: c         and Noncentrality Parameter alpha
16: c
17:         call snrnd(ix,iy,rn1)
18:         rn1=rn1+alpha
19:         call chi2rnd(ix,iy,k,rn2)
20:         rn=rn1/sqrt(rn2/k)
21:         return
22:         end
```

`ntrnd(ix,iy,k,alpha,rn)` have to be utilized with `urnd(ix,iy,rn)` on p.83, `snrnd(ix,iy,rn)` on p.85 and `chi2rnd(ix,iy,k,rn)` on p.102. Again, note that from computational point of view `chi2rnd(ix,iy,k,rn)` should be replaced by `chi2rnd6(ix,iy,k,rn)` on p.215.

2.3 Inverse Transform Method

In Section 2.2, we have introduced the probability density functions which can be derived by transforming the uniform random variables between zero and one. In this section, the probability density functions obtained by the inverse transform method are presented and the corresponding random number generators are shown.

The inverse transform method is represented as follows. Let X be a random variable which has a cumulative distribution function $F(\cdot)$. When $U \sim U(0,1)$, $F^{-1}(U)$ is equal to X. The proof is obtained from the following fact:

$$P(X < x) = P(F^{-1}(U) < x) = P(U < F(x)) = F(x).$$

In other words, let u be a random draw of U, where $U \sim U(0,1)$, and $F(\cdot)$ be a distribution function of X. When we perform the following inverse transformation:

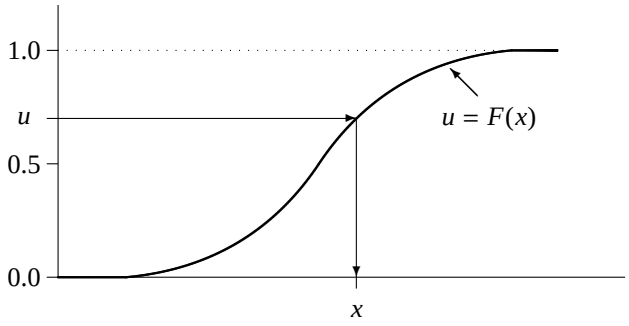
$$x = F^{-1}(u),$$

x implies the random draw generated from $F(\cdot)$. Thus, given u, x is obtained, which is described in Figure 2.12.

The inverse transform method shown above is useful when $F(\cdot)$ can be computed easily and the inverse distribution function, i.e., $F^{-1}(\cdot)$, has a closed form. For example, recall that $F(\cdot)$ cannot be obtained explicitly in the case of the normal distribution because the integration is included in the normal cumulative distribution (conventionally we approximate the normal cumulative distribution when we want to evaluate it). If no closed form of $F^{-1}(\cdot)$ is available but $F(\cdot)$ is still computed easily, an iterative method such as the Newton-Raphson method can be applied. Define $k(x) = F(x) - u$. The first order Taylor series expansion around $x = x^*$ is:

$$0 = k(x) \approx k(x^*) + k'(x^*)(x - x^*).$$

Figure 2.12: Inverse Transformation Method: Continuous Type



Then, we obtain:

$$x = x^* - \frac{k(x^*)}{k'(x^*)} = x^* - \frac{F(x^*) - u}{f(x^*)}.$$

Replacing x and x^* by $x^{(i)}$ and $x^{(i-1)}$, we have the following iteration:

$$x^{(i)} = x^{(i-1)} - \frac{F(x^{(i-1)}) - u}{f(x^{(i-1)})},$$

for $i = 1, 2, \cdots$. The convergence value of $x^{(i)}$ is taken as a solution of equation $u = F(x)$. Thus, based on u, a random draw x is derived from $F(\cdot)$. However, we should keep in mind that this procedure takes a lot of time computationally, because we need to repeat the convergence computation shown above as many times as we want to generate.

2.3.1 Uniform Distribution: $U(a, b)$

In Section 2.1.1, the uniform distribution between zero and one is discussed. In this section, we introduce the uniform distribution between a and b for $a < b$, i.e.,

$$f(x) = \begin{cases} \dfrac{1}{b-a}, & \text{for } a < x < b, \\ 0, & \text{otherwise,} \end{cases}$$

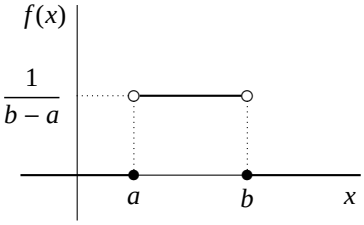
for $a < x < b$. When $a = 0$ and $b = 1$, the uniform distribution above is equivalent to the uniform distribution discussed in Section 2.1. The uniform distribution between a and b is displayed in Figure 2.13.

Mean, variance and the moment-generating function are given by:

$$\mathrm{E}(X) = \frac{a+b}{2}, \qquad \mathrm{V}(X) = \frac{(b-a)^2}{12}, \qquad \phi(\theta) = \frac{e^{b\theta} - e^{a\theta}}{(b-a)\theta}.$$

In the case of the uniform distribution, we have to use L'Hospital's theorem to obtain $\mathrm{E}(X^k)$, $k = 1, 2, \cdots$. When we have $0/0$ or ∞/∞, **L'Hospital's theorem** is helpful, which is shown as follows.

Figure 2.13: Uniform Distribution: $U(a, b)$



- If $\lim_{x\to a} f(x) = \lim_{x\to a} g(x) = 0$ and $\lim_{x\to a} \dfrac{f'(x)}{g'(x)} = l < \infty$, then we have $\lim_{x\to a} \dfrac{f(x)}{g(x)} = l < \infty$. We can replace $x \to a$ by $x \to \pm\infty$ without any modification.
- If $\lim_{x\to a} f(x) = \lim_{x\to a} g(x) = \infty$ and $\lim_{x\to a} \dfrac{f'(x)}{g'(x)} = l < \infty$, then we have $\lim_{x\to a} \dfrac{f(x)}{g(x)} = l < \infty$. We can replace $x \to a$ by $x \to \pm\infty$ without any modification.

In the case of the uniform distribution, $\phi'(\theta)$ and $\phi''(\theta)$ are given by:

$$\phi'(\theta) = \frac{be^{b\theta} - ae^{a\theta}}{(b-a)\theta} - \frac{e^{b\theta} - e^{a\theta}}{(b-a)\theta^2},$$
$$\phi''(\theta) = \frac{b^2e^{b\theta} - a^2e^{a\theta}}{(b-a)\theta} - \frac{2(be^{b\theta} - ae^{a\theta})}{(b-a)\theta^2} + \frac{2(e^{b\theta} - e^{a\theta})}{(b-a)\theta^3}.$$

When $\theta \longrightarrow 0$, both $\phi'(\theta)$ and $\phi''(\theta)$ are of the form $0/0$. Therefore, L'Hospital's theorem can be applied. Thus, we obtain $\phi'(0) = (b+a)/2$ and $\phi''(0) = (b^2+ab+a^2)/3$.

Suppose that X has the uniform distribution between a and b, i.e., $X \sim U(a, b)$. Then, the cumulative distribution function $F(x)$ is given by:

$$F(x) = \begin{cases} 0, & \text{if } x \le a, \\ \displaystyle\int_a^x \frac{1}{b-a}\,\mathrm{d}x = \frac{x-a}{b-a}, & \text{if } a < x < b, \\ 1, & \text{if } x \ge b. \end{cases}$$

Thus, generating u from $U(0, 1)$ and utilizing $x = F^{-1}(u)$, we can obtain x as follows:

$$x = F^{-1}(u) = (b-a)u + a,$$

which is a random draw generated from $U(a, b)$. The Fortran source program for the uniform random number generator is shown in `urnd_ab(ix,iy,a,b,rn)`.

`urnd_ab(ix,iy,a,b,rn)`

```
1:       subroutine urnd_ab(ix,iy,a,b,rn)
2: c
```

```
3: c  Use "urnd_ab(ix,iy,a,b,rn)"
4: c  together with "urnd(ix,iy,rn)".
5: c
6: c  Input:
7: c    ix, iy:  Seeds
8: c    a, b:    Range of rn
9: c  Output:
10: c   rn: Uniform Random Draw U(a,b)
11: c
12:       call urnd(ix,iy,rn1)
13:       rn=a+(b-a)*rn1
14:       return
15:       end
```

`urnd_ab(ix,iy,a,b,rn)` should be used together with `urnd(ix,iy,rn)` on p.83.

2.3.2 Normal Distribution: $N(0,1)$

The random number generation methods from the standard normal distribution are discussed in Section 2.2.1, where three random number generators are introduced. Two generators are based on the Box-Muller transformation, where two independent uniform random draws yield two independent standard normal random draws, which generators are shown in `snrnd(ix,iy,rn)` on p.85 and `snrnd2(ix,iy,rn)` on p.87. Another generator utilizes the central limit theorem, where the sample mean from n uniform random draws are approximated to be normal, which is found in `snrnd3(ix,iy,rn)` on p.88 ($n = 12$ is taken in `snrnd3`).

In this section, we consider the random number generator based on the inverse transform method. However, it is not easy to evaluate the integration in the normal cumulative distribution function, i.e., $F(x) = \int_{-\infty}^{x} (2\pi)^{-1/2} \exp(-\frac{1}{2}t^2)\, \mathrm{d}t$. Therefore, we cannot apply the inverse transform method directly. However, as an approximation of $F(x)$, we may utilize `snperpt(p,x)` on p.90 to obtain the percent points of the standard normal distribution. `snperpt(p,x)` gives us the approximations of the percent points. The precision is known to be very good. As mentioned on p.90, the maximum error of approximation of x is 1.5×10^{-8} if the function is evaluated in double precision and 1.8×10^{-6} if the function is evaluated in single precision.

Let us define $y = \sqrt{-2\log(u)}$. According to `snperpt(p,x)`, the approximation is of the form:

$$x = y + \frac{p_0 + p_1 y + p_2 y^2 + p_3 y^3 + p_4 y^4}{q_0 + q_1 y + q_2 y^2 + q_3 y^3 + q_4 y^4},$$

for $0 < u \leq 0.5$, where the coefficients are given by p_i and q_i for $i = 0, 1, \cdots, 4$, which are shown in `snperpt(p,x)`. See Odeh and Evans (1974) for the algorithm above.

Thus, given u, we can obtain x in the subroutine `snrnd4(ix,iy,rn)`.

snrnd4(ix,iy,rn)

```
1:        subroutine snrnd4(ix,iy,rn)
2:  c
3:  c  Use "snrnd4(ix,iy,rn)"
4:  c  together with "urnd(ix,iy,rn)"
5:  c            and  "snperpt(p,x)".
6:  c
7:  c  Input:
8:  c    ix, iy:  Seeds
9:  c  Output:
10: c    rn:  N(0,1) Random Draws
11: c
12:       call urnd(ix,iy,rn1)
13:       call snperpt(rn1,rn)
14:       return
15:       end
```

Note that `snrnd4(ix,iy,rn)` should be used together with `urnd(ix,iy,rn)` on p.83 and `snperpt(p,x)` on p.90.

Since `snperpt(p,x)` gives us good precision, `snrnd4(ix,iy,rn)` might have no problem as a random number generator. However, the approximation of the normal cumulative distribution function is utilized in `snrnd4(ix,iy,rn)`, which is not theoretically correct, even though the approximation gives us practically enough precision. However, in `snrnd4(ix,iy,rn)`, only one uniform random draw is generated to obtain one standard normal random draw. Therefore, it is clear that `snrnd4(ix,iy,rn)` is much less computer-intensive than `snrnd(ix,iy,rn)`, `snrnd2(ix,iy,rn)` and `snrnd3(ix,iy,rn)`. In Section 3.5.1, we compare various standard normal random number generators with respect to both precision and computational burden.

2.3.3 Exponential Distribution

In Section 2.2.4, the exponential distribution with parameter β has been already discussed. In this section, we consider generating exponential random draws using the inverse transformation. Since we have the density $f(x) = (1/\beta)e^{-x/\beta}$ for $0 < x < \infty$ and $\beta > 0$, the integration is given by:

$$F(x) = \int_0^x \frac{1}{\beta} e^{-\frac{t}{\beta}} \, \mathrm{d}t = 1 - e^{-\frac{x}{\beta}}.$$

Therefore, through the inverse transform method, we have the following equation:

$$x = F^{-1}(u) = -\beta \log(1 - u).$$

Thus, given u, we can obtain x using the above equation.

Since $1 - U$ is uniform when U is a uniform random variable, $x = -\beta \log(u)$ also gives us the exponential random draw. Therefore, here we have the exactly same source code as `exprnd(ix,iy,beta,rn)` in Section 2.2.4.

2.3.4 Double Exponential Distribution (LaPlace Distribution)

In Section 2.2.11, the double exponential distribution with parameters α and β has been already discussed. Since we have the density $f(x) = \frac{1}{2\beta}e^{-|x-\alpha|/\beta}$ for $-\infty < x < \infty$, $-\infty < \alpha < \infty$ and $\beta > 0$, the integration is given by:

$$F(x) = \begin{cases} \displaystyle\int_{-\infty}^{x} \frac{1}{2\beta}e^{(t-\alpha)/\beta}\,\mathrm{d}t, & \text{for } x \le \alpha, \\ \displaystyle\int_{-\infty}^{\alpha} \frac{1}{2\beta}e^{(t-\alpha)/\beta}\,\mathrm{d}t + \int_{\alpha}^{x} \frac{1}{2\beta}e^{-(t-\alpha)/\beta}\,\mathrm{d}t, & \text{for } x > \alpha, \end{cases}$$

$$= \begin{cases} \left[\frac{1}{2}e^{(t-\alpha)/\beta}\right]_{-\infty}^{x}, & \text{for } x \le \alpha, \\ \left[\frac{1}{2}e^{(t-\alpha)/\beta}\right]_{-\infty}^{\alpha} + \left[-\frac{1}{2}e^{-(t-\alpha)/\beta}\right]_{\alpha}^{x}, & \text{for } x > \alpha, \end{cases}$$

$$= \begin{cases} \frac{1}{2}e^{(x-\alpha)/\beta}, & \text{for } x \le \alpha, \\ 1 - \frac{1}{2}e^{-(x-\alpha)/\beta}, & \text{for } x > \alpha, \end{cases}$$

$x \le \alpha$ implies $F(x) \le 0.5$, and conversely $x > \alpha$ implies $F(x) > 0.5$. Therefore, by the inverse transform method, we have the following equation:

$$x = F^{-1}(u) = \begin{cases} \alpha + \beta\log(2u), & \text{for } u \le 0.5, \\ \alpha - \beta\log(2(1-u)), & \text{for } u > 0.5, \end{cases}$$

Thus, given u, we can obtain x using the above equation. Therefore, the source code is represented as `dexprnd2(ix,iy,alpha,beta,rn)`.

`dexprnd2(ix,iy,alpha,beta,rn)`

```
      subroutine dexprnd2(ix,iy,alpha,beta,rn)
c
c  Use "dexprnd2(ix,iy,alpha,beta,rn)"
c  together with "urnd(ix,iy,rn)".
c
c  Input:
c    ix, iy:  Seeds
c    alpha: Location Parameter
c    beta:  Scale Parameter
c  Output:
c    rn:  Double Exponential Random Draws
c
       call urnd(ix,iy,rn1)
          if( rn1.le.0.5) then
       rn=alpha+beta*log(2.*rn1)
```

```
16:          else
17:       rn=alpha-beta*log(2.-2.*rn1)
18:          endif
19:       return
20:       end
```

Note that `dexprnd2(ix,iy,alpha,beta,rn)` is used with `urnd(ix,iy,rn)` on p.83. See Cheng (1998) and Gentle (1998) for the double exponential random number generator.

Judging from the number of uniform random draws to be generated for a double exponential random draw, `dexprnd2(ix,iy,alpha,beta,rn)` might be much faster than `dexprnd(ix,iy,alpha,beta,rn)` on p.117 from computational point of view. Remember that `dexprnd(ix,iy,alpha,beta,rn)` requires two independent exponential random draws. Therefore, `dexprnd2` is recommended, rather than `dexprnd`.

2.3.5 Cauchy Distribution

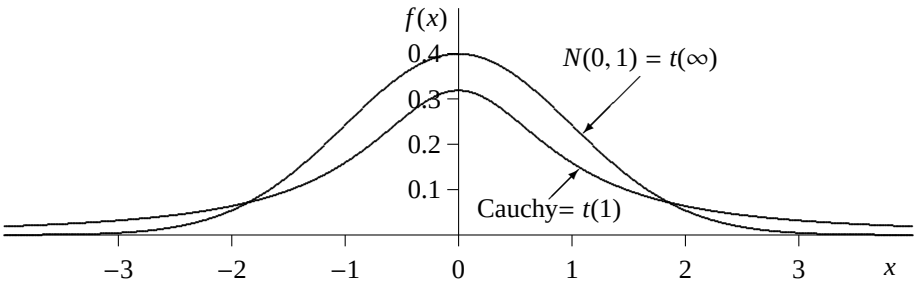
The Cauchy distribution is given by:

$$f(x) = \frac{1}{\beta\pi}\Big(1 + \frac{(x-\alpha)^2}{\beta^2}\Big)^{-1},$$

for $-\infty < x < \infty$, $-\infty < \alpha < \infty$ and $\beta > 0$, where α denotes a location parameter and β represents a scale parameter.

Given $\alpha = 0$ and $\beta = 1$, the Cauchy distribution is equivalent to the t distribution with one degree of freedom, i.e., $t(1)$. The t distribution is discussed in Section 2.2.10. The Cauchy distribution with parameters $\alpha = 0$ and $\beta = 1$ is displayed in Figure 2.14, which is compared with the standard normal distribution.

Figure 2.14: Cauchy Distribution: $\alpha = 0$ and $\beta = 1$



Mean, variance and the moment-generating function do not exist in the case of the Cauchy distribution. The tails of the Cauchy distribution are too fat to have mean and variance.

The distribution function is represented as follows:

$$F(x) = \int_{-\infty}^{x} \frac{1}{\beta\pi}\Big(1 + \frac{(t-\alpha)^2}{\beta^2}\Big)^{-1} \mathrm{d}t = \frac{1}{2} + \frac{1}{\pi}\tan^{-1}\Big(\frac{x-\alpha}{\beta}\Big).$$

The inverse function of the Cauchy distribution is written as:

$$x = F^{-1}(u) = \alpha + \beta\tan\Big(\pi(u - \frac{1}{2})\Big).$$

Thus, generating a random draw u from $U(0,1)$, we have a Cauchy random draw x from the above equation. Therefore, the source code of the Cauchy random number generator is shown in `crnd(ix,iy,alpha,beta,rn)`.

`crnd(ix,iy,alpha,beta,rn)`

```
1:        subroutine crnd(ix,iy,alpha,beta,rn)
2:  C
3:  C  Use "crnd(ix,iy,alpha,beta,rn)"
4:  C  together with "urnd(ix,iy,rn)".
5:  C
6:  C  Input:
7:  C    ix, iy:  Seeds
8:  C    alpha: Location Parameter
9:  C    beta:  Scale Parameter
10: C  Output:
11: C    rn: Cauchy Random Draw
12: C
13:       pi= 3.1415926535897932385
14:       call urnd(ix,iy,rn1)
15:       rn=alpha+beta*tan( pi*(rn1-0.5) )
16:       return
17:       end
```

`crnd(ix,iy,alpha,beta,rn)` requires `urnd(ix,iy,rn)` on p.83.

An alternative approach is to use two independent standard normal random variables, say Z_1 and Z_2. Let us define $X = Z_1/Z_2$. Then, X has the Cauchy distribution with parameters $\alpha = 0$ and $\beta = 1$. However, this random number generator is computationally inefficient, compared with `crnd(ix,iy,alpha,beta,rn)`. The random number generator based on $X = Z_1/Z_2$ and $Z_i \sim N(0,1)$ for $i = 1, 2$ uses two standard normal random draws (i.e., four independent uniform random draws), while `crnd(ix,iy,alpha,beta,rn)` utilizes only one uniform random draws. Therefore, `crnd(ix,iy,alpha,beta,rn)` is recommended in the case of generating Cauchy random draws.

2.3.6 Logistic Distribution

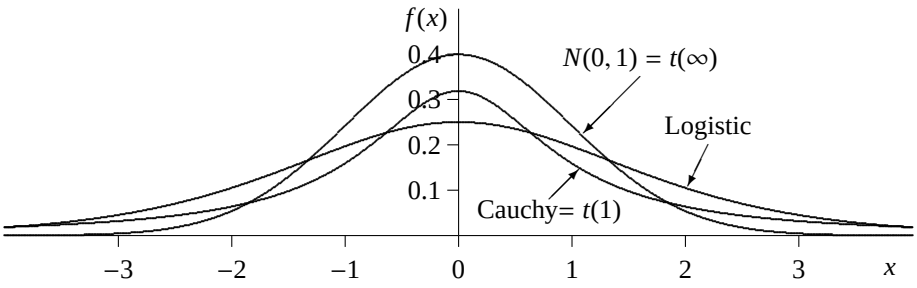
The logistic distribution with location parameter α and scale parameter β is written as follows:

$$f(x) = \frac{\exp(\frac{x-\alpha}{\beta})}{\beta\Big(1+\exp(\frac{x-\alpha}{\beta})\Big)^2},$$

for $-\infty < x < \infty$, $-\infty < \alpha < \infty$ and $\beta > 0$.

The logistic distribution with parameters $\alpha = 0$ and $\beta = 1$ is displayed in Figure 2.15, where the standard normal distribution and the t distribution with one degree of freedom are also described for comparison.

Figure 2.15: Logistic Distribution: $\alpha = 0$ and $\beta = 1$



Mean, variance and the moment-generating function of the logistic random variable are given by:

$$\mathrm{E}(X) = \alpha, \qquad \mathrm{V}(X) = \frac{\beta^2\pi^2}{3}, \qquad \phi(\theta) = e^{\alpha\theta}\pi\beta\theta\mathrm{cosec}(\pi\beta\theta).$$

The cumulative distribution function is given by:

$$F(x) = \int_{-\infty}^{x} \frac{\exp(\frac{x-\alpha}{\beta})}{\beta\Big(1+\exp(\frac{x-\alpha}{\beta})\Big)^2}\,\mathrm{d}t = \frac{1}{1+e^{-(x-\alpha)/\beta}}.$$

Therefore, based the uniform random draw u, the logistic random draw x is obtained from the following formula:

$$x = F^{-1}(u) = \alpha + \beta\log\Big(\frac{u}{1-u}\Big).$$

Thus, given u, x is easily computed. The Fortran 77 source code of the logistic random draw generator with location parameter α and scale parameter β is shown in `logisticrnd(ix,iy,alpha,beta,rn)`.

logisticrnd(ix,iy,alpha,beta,rn)

```
      subroutine logisticrnd(ix,iy,alpha,beta,rn)
c
c  Use "logisticrnd(ix,iy,alpha,beta,rn)"
c  together with "urnd(ix,iy,rn)".
c
c  Input:
c    ix, iy: Seeds
c    alpha: Location Parameter
c    beta:  Scale Parameter
c  Output:
c    rn: Logistic Random Draw
c
       call urnd(ix,iy,rn1)
       rn=alpha+beta*log( rn1/(1.-rn1) )
       return
       end
```

Note that `logisticrnd(ix,iy,alpha,beta,rn)` is used with `urnd(ix,iy,rn)` on p.83.

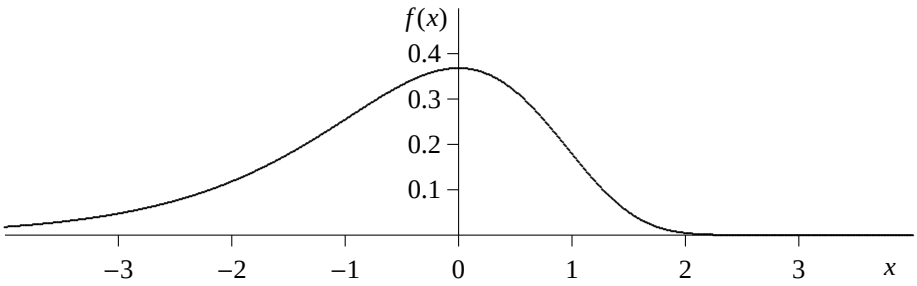
2.3.7 Extreme-Value Distribution (Gumbel Distribution)

The extreme-value distribution (or the Gumbel distribution) with parameters α and β is of the form:

$$f(x) = \frac{1}{\beta}\exp\Big(-\frac{x-\alpha}{\beta}\Big)\exp(-e^{-(x-\alpha)/\beta}),$$

for $-\infty < x < \infty$, $-\infty < \alpha < \infty$ and $\beta > 0$. α is called the location parameter and β is the scale parameter. In the case where $\alpha = 0$ and $\beta = 1$ are taken, the extreme-value distribution is skewed to the left, which is shown in Figure 2.16.

Figure 2.16: Extreme-Value Distribution (Gumbel Distribution): $\alpha = 0$ and $\beta = 1$



Mean, variance and the moment-generating function are given by:

$$\mathrm{E}(X) = \alpha + \beta\gamma, \qquad \mathrm{V}(X) = \frac{\beta^2\pi^2}{6}, \qquad \phi(\theta) = e^{\alpha\theta}\Gamma(1-\beta\theta),$$

where $\gamma \approx 0.5772156599$, which is called Euler's constant and defined as:

$$\sum_{n=1}^{\infty}\Big(\frac{1}{n} - \log(1+\frac{1}{n})\Big) = \lim_{n\to\infty}\Big(1+\frac{1}{2}+\frac{1}{3}+\cdots+\frac{1}{n}-\log n\Big) = \gamma.$$

The cumulative distribution function is computed as follows:

$$F(x) = \int_{-\infty}^{x} \frac{1}{\beta}\exp\Big(-\frac{t-\alpha}{\beta}\Big)\exp(-e^{-(x-\alpha)/\beta})\,\mathrm{d}t = \exp(-e^{-(x-\alpha)/\beta}).$$

Therefore, given u, a random draw x is obtained from:

$$x = F^{-1}(u) = \alpha - \beta\log(-\log u).$$

The Fortran program for the Gumbel random number generator with parameters α and β is given by `gumbelrnd(ix,iy,alpha,beta,rn)`.

`gumbelrnd(ix,iy,alpha,beta,rn)`

```
1:        subroutine gumbelrnd(ix,iy,alpha,beta,rn)
2:  c
3:  c  Use "gumbelrnd(ix,iy,alpha,beta,rn)"
4:  c  together with "urnd(ix,iy,rn)".
5:  c
6:  c  Input:
7:  c    ix, iy:  Seeds
8:  c    alpha:  Location Parameter
9:  c    beta:   Scale Parameter
10: c  Output:
11: c    rn: Gumbel Random Draw
12: c
13:       call urnd(ix,iy,rn1)
14:       rn=alpha-beta*log( -log(rn1) )
15:       return
16:       end
```

Note that `gumbelrnd(ix,iy,alpha,beta,rn)` requires `urnd(ix,iy,rn)` on p.83.

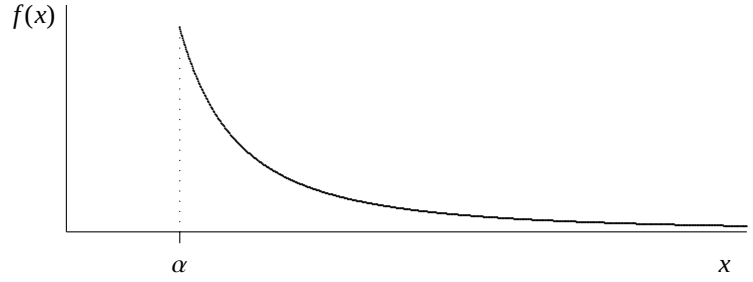
2.3.8 Pareto Distribution

The Pareto distribution is written as follows:

$$f(x) = \begin{cases} \beta\alpha^{\beta}x^{-\beta-1}, & \text{for } \alpha < x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

where $\alpha > 0$ and $\beta > 0$. The Pareto distribution is displayed in Figure 2.17.

Figure 2.17: Pareto Distribution



Mean and variance are given by:

$$\mathrm{E}(X) = \frac{\alpha\beta}{\beta - 1}, \qquad \text{for } \beta > 1,$$
$$\mathrm{V}(X) = \frac{\alpha^2\beta}{(\beta-1)^2(\beta-2)}, \qquad \text{for } \beta > 2,$$

$\phi(\theta)$ does not exist, because $\beta > p$ is required for $\mathrm{E}(X^p) < \infty$.

Integrating the density function with respect to x, the cumulative distribution is obtained as follows:

$$F(x) = \int_{\alpha}^{x} \beta\alpha^{\beta}t^{-\beta-1}\,\mathrm{d}t = 1 - \left(\frac{\alpha}{x}\right)^{\beta}.$$

Applying the inverse transform method, we have:

$$x = F^{-1}(u) = \alpha(1-u)^{-1/\beta}.$$

For $U \sim U(0,1)$, U is uniform when $1 - U$ is uniform. Therefore, we can obtain the following inverse transformation:

$$x = F^{-1}(1-u) = \alpha u^{-1/\beta},$$

which is slightly less computational.

Accordingly, `paretornd(ix,iy,alpha,beta,rn)` is the random number generator in this case.

`paretornd(ix,iy,alpha,beta,rn)`

```
1:       subroutine paretornd(ix,iy,alpha,beta,rn)
2: c
3: c  Use "paretornd(ix,iy,alpha,beta,rn)"
4: c  together with "urnd(ix,iy,rn)".
5: c
6: c  Input:
7: c    ix, iy:  Seeds
8: c    alpha, beta: Parameters
9: c  Output:
```

```
10: c     rn: Pareto Random Draw
11: c
12:        call urnd(ix,iy,rn1)
13:        rn=alpha*( rn1**(-1./beta) )
14:        return
15:        end
```

`paretornd(ix,iy,alpha,beta,rn)` is used together with `urnd(ix,iy,rn)` on p.83.

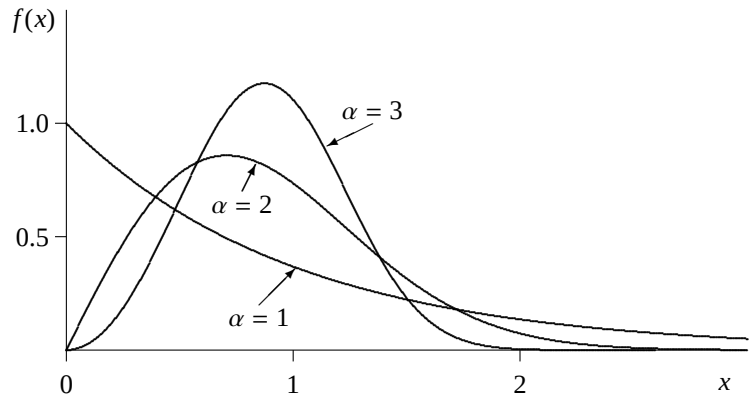
2.3.9 Weibull Distribution

The Weibull distribution with parameters α and β is represented as follows:

$$f(x) = \begin{cases} \frac{\alpha}{\beta}\left(\frac{x}{\beta}\right)^{\alpha-1}\exp\left(-\left(\frac{x}{\beta}\right)^{\alpha}\right), & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

where $\alpha > 0$ and $\beta > 0$ should be taken. α is called the shape parameter and β is the scale parameter. The Weibull distribution is in Figure 2.18, where given $\beta = 1$ the cases of $\alpha = 1, 2, 3$ are displayed.

Figure 2.18: Weibull Distribution: $\alpha = 1, 2, 3$ and $\beta = 1$



Mean and variance are obtained as follows:

$$\mathrm{E}(X) = \beta\Gamma(1 + \alpha^{-1}), \qquad \mathrm{V}(X) = \beta^2\left(\Gamma(1 + 2\alpha^{-1}) - \Gamma^2(1 + \alpha^{-1})\right).$$

Integrating $f(x)$ with respect to x, we have $F(x)$ as follows:

$$F(x) = \int_0^x \frac{\alpha}{\beta}\left(\frac{x}{\beta}\right)^{\alpha-1}\exp\left(-\left(\frac{t}{\beta}\right)^{\alpha}\right)\mathrm{d}t = 1 - \exp\left(-\left(\frac{x}{\beta}\right)^{\alpha}\right).$$

Thus, given a uniform random draw u, a random draw from $f(x)$ is generated as:

$$x = F^{-1}(u) = \beta(-\log(1 - u))^{1/\alpha}.$$

For $U \sim U(0,1)$, U is uniform when $1-U$ is uniform. Therefore, we can obtain the following inverse transformation:

$$x = F^{-1}(1-u) = \beta(-\log u)^{1/\alpha}.$$

which is slightly less computational.

The source code is given by `wrnd(ix,iy,alpha,beta,rn)`.

`wrnd(ix,iy,alpha,beta,rn)`

```
1:       subroutine wrnd(ix,iy,alpha,beta,rn)
2: C
3: C  Use "wrnd(ix,iy,alpha,beta,rn)"
4: C  together with "urnd(ix,iy,rn)".
5: C
6: C  Input:
7: C    ix, iy:  Seeds
8: C    alpha: Scale Parameter
9: C    beta:  Sape Parameter
10: C  Output:
11: C    rn: Weibull Random Draw
12: C
13:       call urnd(ix,iy,rn1)
14:       rn=beta*( (-log(rn1))**(1./alpha) )
15:       return
16:       end
```

`wrnd(ix,iy,alpha,beta,rn)` should be used together with `urnd(ix,iy,rn)` on p.83.

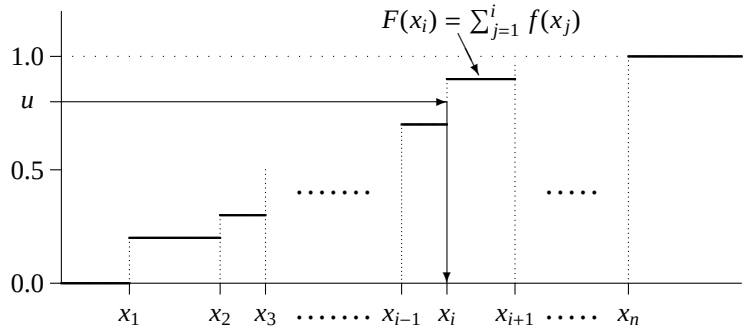
For $\alpha = 1$, the Weibull density reduces to the exponential density. When $\alpha = 2$ and $\beta = \sqrt{2\sigma^2}$, the Weibull distribution reduces to the **Rayleigh distribution**. The cases of $\alpha = 1, 2$ are shown in Figure 2.18.

2.4 Using $U(0,1)$: Discrete Type

In Sections 2.2 and 2.3, the random number generators from continuous distributions are discussed, i.e., the transformation of variables in Section 2.2 and the inverse transform method in Section 2.3 are utilized. Based on the uniform random draw between zero and one, in this section we deal with some discrete distributions and consider generating their random numbers.

As a representative random number generation method, we can consider utilizing the inverse transform method in the case of discrete random variables. Suppose that a discrete random variable X can take $x_1, x_2, \cdots, x_n$, where the probability which X takes x_i is given by $f(x_i)$, i.e., $P(X = x_i) = f(x_i)$. Generate a uniform random draw u, which is between zero and one. Consider the case where we have $F(x_{i-1}) \leq u < F(x_i)$, where $F(x_i) = P(X \leq x_i)$ and $F(x_0) = 0$. Then, the random draw of X is given by x_i. This relationship between x_i and u is described in Figure 2.19.

Figure 2.19: Inverse Transformation Method: Discrete Type



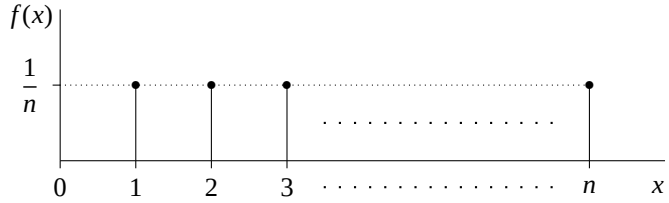
2.4.1 Rectangular Distribution (Discrete Uniform Distribution)

The rectangular distribution (or the discrete uniform distribution) is given by:

$$f(x) = \begin{cases} \dfrac{1}{n}, & \text{for } x = 1, 2, \cdots, n, \\ 0, & \text{otherwise,} \end{cases}$$

which implies that the discrete uniform random variable X can take $1, 2, \cdots, n$ with equal probability $1/n$. As a representative example of this distribution, we can take an experiment of casting a die, where $n = 6$ and $x = 1, 2, \cdots, 6$. The rectangular distribution (or the discrete uniform distribution) is displayed in Figure 2.20.

Figure 2.20: Rectangular Distribution (Discrete Uniform Distribution)



Mean, variance and the moment-generating function of the discrete uniform random variable X are represented as:

$$\mathrm{E}(X) = \frac{n+1}{2}, \qquad \mathrm{V}(X) = \frac{n^2-1}{12}, \qquad \phi(\theta) = \frac{e^{\theta}(1-e^{n\theta})}{n(1-e^{\theta})}.$$

In order to obtain E(X) and V(X) using $\phi(\theta)$, we utilize L'Hospital's theorem on p.123.

The cumulative distribution function of the rectangular distribution (or the discrete uniform distribution) is given by:

$$F(x) = \sum_{i=1}^{x} f(i) = \frac{i}{n}.$$

Utilizing the inverse transform method shown in Figure 2.19, when we have the following:

$$F(x-1) \le u < F(x) \quad \Longrightarrow \quad x - 1 \le nu < x,$$

the random variable X should take x, where u represents a uniform random number between zero and one. Since $x - 1$ is an integer, we have $x - 1 = [nu]$, where $[nu]$ indicates the maximum integer which is not greater than nu. Thus, when u is generated from $U(0,1)$, $x = [nu] + 1$ implies the random draw of X, where one of $1, 2, \cdots, n$ is chosen with equal probability $1/n$. Therefore, the Fortran program for the discrete uniform random number generator is written as `recrnd(ix,iy,n,rn)`.

`recrnd(ix,iy,n,rn)`

```
1:       subroutine recrnd(ix,iy,n,rn)
2: c
3: c  Use "recrnd(ix,iy,n,rn)"
4: c  together with "urnd(ix,iy,rn)".
5: c
6: c  Input:
7: c    ix, iy:  Seeds
8: c    n:  Domain of rn
9: c  Output:
10: c    rn: Discrete Uniform Random Draw
11: c
12:       call urnd(ix,iy,rn1)
13:       rn=int(float(n)*rn1)+1.
14:       return
15:       end
```

Note that `recrnd(ix,iy,n,rn)` should be used together with `urnd(ix,iy,rn)` on p.83.

2.4.2 Bernoulli Distribution

The Bernoulli distribution with parameter p is of the form:

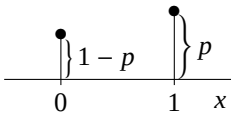
$$f(x) = \begin{cases} p^x(1-p)^{1-x}, & \text{for } x = 0, 1, \\ 0, & \text{otherwise,} \end{cases}$$

where $0 \le p \le 1$. A random experiment whose outcomes are classified into two categories, e.g., "success" and "failure", is called a **Bernoulli trial**. If a random variable X takes 1 when a Bernoulli trial results in success and 0 when it results in failure, then X has a Bernoulli distribution with parameter p, where $p = P(\text{success})$. The Bernoulli probability function is described in Figure 2.21.

Mean, variance and the moment-generating function of the Bernoulli random variable X are given by:

$$\mathrm{E}(X) = p, \qquad \mathrm{V}(X) = p(1-p), \qquad \phi(\theta) = 1 - p + pe^{\theta}.$$

Figure 2.21: Bernoulli Distribution



The random variable X takes $X = 1$ with probability p and $X = 0$ otherwise. Therefore, for a uniform random draw u between zero and one, we take $X = 1$ when $0 \le u < p$ and $X = 0$ when $p \le u \le 1$, or $X = 0$ when $0 \le u < 1 - p$ and $X = 1$ when $1 - p \le u \le 1$. Thus, the source code of the Bernoulli random number generator with parameter p is shown in `brnd(ix,iy,p,rn)`.

`brnd(ix,iy,p,rn)`

```
1:       subroutine brnd(ix,iy,p,rn)
2: c
3: c  Use "brnd(ix,iy,p,rn)"
4: c  together with "urnd(ix,iy,rn)".
5: c
6: c  Input:
7: c    ix, iy:  Seeds
8: c    p:  Probability of "Success"
9: c  Output:
10: c    rn: Bernoulli Random Draw
11: c
12:       call urnd(ix,iy,rn1)
13:       if( rn1.le.p ) rn=1.0
14:       if( rn1.gt.p ) rn=0.0
15:       return
16:       end
```

Note that `brnd(ix,iy,p,rn)` should be used together with `urnd(ix,iy,rn)` on p.83.

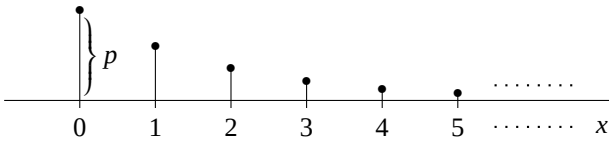
2.4.3 Geometric Distribution (Pascal Distribution)

The geometric distribution (or Pascal distribution) with parameter p is given by:

$$f(x) = \begin{cases} p(1-p)^x, & \text{for } x = 0, 1, 2, \cdots, \\ 0, & \text{otherwise,} \end{cases}$$

where $0 \le p \le 1$. As discussed in Section 2.4.2, p denotes the probability of success in one random experiment, i.e., $p = P(\text{success})$. The probability function $f(x)$ represents the probability which continues to fail x times from the first trial up to the xth trial and has the first success in the $(x + 1)$th trial.

Figure 2.22: Geometric Distribution (Pascal Distribution)



Mean, variance and the moment-generating function are as follows:

$$\mathrm{E}(X) = \frac{1-p}{p}, \qquad \mathrm{V}(X) = \frac{1-p}{p^2}, \qquad \phi(\theta) = \frac{p}{1-(1-p)e^{\theta}}.$$

The Fortran program for the Geometric random number generator with parameter p is given by `georand(ix,iy,p,rn)`.

`georand(ix,iy,p,rn)`

```
1:        subroutine georand(ix,iy,p,rn)
2:  c
3:  c  Use "georand(ix,iy,p,rn)"
4:  c  together with "urand(ix,iy,rn)".
5:  c
6:  c  Input:
7:  c    ix, iy:  Seeds
8:  c    p:  Probability of "Sucsess"
9:  c  Output:
10: c    rn: Geometric Random Draw
11: c
12:       rn=0.0
13:     1 call urand(ix,iy,rn1)
14:       if( rn1.le.p ) go to 2
15:       rn=rn+1
16:       go to 1
17:     2 return
18:       end
```

Note that `georand(ix,iy,p,rn)` should be used together with `urand(ix,iy,rn)` on p.83.

As for less computational computer algorithm, we consider the cumulative distribution function, where the inverse transform method is applied. From the definition, the cumulative distribution function of the geometric random variable is written as follows:

$$\begin{aligned} F(x) &= \sum_{i=0}^{x} f(i) = \sum_{i=0}^{x} pq^i = 1 - P(\text{first } x \text{ trials are all failures}) \\ &= 1-(1-p)^{x+1}. \end{aligned}$$

Let u be the uniform random draw between zero and one. Therefore, given u, x takes the positive integer satisfying:

$$F(x-1) \leq u < F(x),$$

i.e.,

$$1-(1-p)^x \leq u < 1-(1-p)^{x+1},$$

i.e.,

$$x \leq \frac{\log(1-u)}{\log(1-p)} < x+1.$$

That is, x should be the maximum integer less than $\log(1-u)/\log(1-p)$. Therefore, x is taken as:

$$x = \left[\frac{\log(1-u)}{\log(1-p)}\right].$$

Moreover, for $U \sim U(0,1)$, U is uniform when $1-U$ is uniform. Therefore, we can obtain the following:

$$x = \left[\frac{\log u}{\log(1-p)}\right].$$

See Ross (1997, pp.49 – 50). Thus, the random number generator in this case is represented as `geornd2(ix,iy,p,rn)`.

`geornd2(ix,iy,p,rn)`

```
1:        subroutine geornd2(ix,iy,p,rn)
2:  c
3:  c  Use "geornd2(ix,iy,p,rn)"
4:  c  together with "urnd(ix,iy,rn)".
5:  c
6:  c  Input:
7:  c    ix, iy:  Seeds
8:  c    p:  Probability of "Sucsess"
9:  c  Output:
10: c    rn: Geometric Random Draw
11: c
12:       call urnd(ix,iy,rn1)
13:       rn=int(log(rn1)/log(1.-p))
14:       return
15:       end
```

`geornd2(ix,iy,p,rn)` should be utilized together with `urnd(ix,iy,rn)` on p.83.

`geornd2(ix,iy,p,rn)` is much faster than `geornd(ix,iy,p,rn)`, because in order to obtain one geometric random draw the former utilizes one uniform random draw while the latter has to generate $x+1$ uniform random draws.

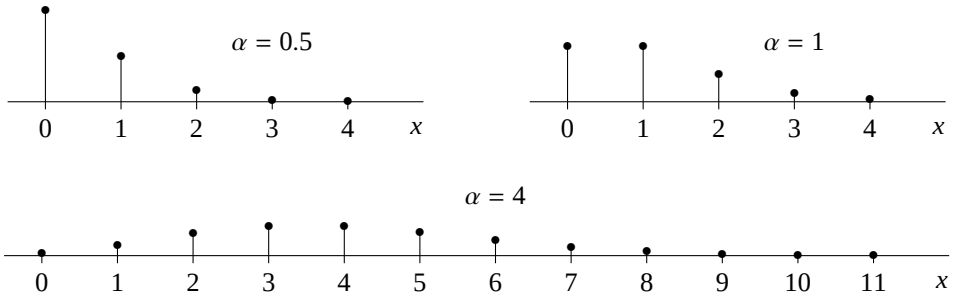
2.4.4 Poisson Distribution

The Poisson distribution with parameter α is given by:

$$f(x) = \begin{cases} \dfrac{e^{-\alpha}\alpha^x}{x!}, & \text{for } x = 0, 1, 2, \cdots, \\ 0, & \text{otherwise,} \end{cases}$$

where $\alpha > 0$ should be satisfied. The Poisson distribution provides a realistic count model for various random phenomena, e.g. the number of traffic accidents per week, the number of telephone calls per hour and so on.

Figure 2.23: Poisson Distribution



Mean, variance and the moment-generating function of the Poisson random variable with parameter α are:

$$\mathrm{E}(X) = \alpha, \qquad \mathrm{V}(X) = \alpha, \qquad \phi(\theta) = \exp(\alpha(e^{\theta} - 1)).$$

Suppose that Z_1, Z_2, $\cdots$, Z_x are mutually independently distributed. Z_i has an exponential distribution with parameter $1/\alpha$, i.e., $\mathrm{E}(Z_i) = 1/\alpha$. Then, the maximum integer of x which satisfies $Z_1 + Z_2 + \cdots + Z_x < 1$ has a Poisson distribution with parameter α. We can prove this fact as follows. Define $Y_n = \alpha \sum_{i=1}^{n} Z_i$. Then, it is known that Y_n has a gamma distribution with parameters n and 1. Therefore, the following equalities hold:

$$\begin{aligned} P(X = x) &= P(X \le x) - P(X \le x - 1) = P(Y_{x+1} \ge \alpha) - P(Y_x \ge \alpha) \\ &= \int_{\alpha}^{\infty} \frac{e^{-t}t^x}{x!}\,\mathrm{d}t - \int_{\alpha}^{\infty} \frac{e^{-t}t^{x-1}}{(x-1)!}\,\mathrm{d}t = \frac{e^{-\alpha}\alpha^x}{x!}. \end{aligned}$$

In other words, the Poisson random variable x is generated as follows:

$$\begin{aligned} X &= \max_x\{x; \sum_{i=1}^{x} Z_i \le 1\} = \min_x\{x; \sum_{i=1}^{x} Z_i > 1\} - 1 \\ &= \min_x\{x; \alpha \sum_{i=1}^{x} Z_i > \alpha\} - 1 = \min_x\{x; Y_x > \alpha\} - 1. \end{aligned}$$

See Ross (1997, p.64). Therefore, the source code for the Poisson random number generator with parameter α is shown in `pornd(ix,iy,alpha,rn)`.

`pornd(ix,iy,alpha,rn)`

```
1:        subroutine pornd(ix,iy,alpha,rn)
2:  c
3:  c  Use "pornd(ix,iy,alpha,rn)"
4:  c  together with "exprnd(ix,iy,alpha,rn)"
5:  c           and  "urnd(ix,iy,rn)".
6:  c
7:  c  Input:
8:  c    ix, iy:  Seeds
9:  c    alpha:  Parameter
10: c  Output:
11: c    rn: Poisson Random Draw
12: c
13:       rn=0.0
14:       sum=0.0
15:     1 call exprnd(ix,iy,1./alpha,rn1)
16:       sum=sum+rn1
17:       if( sum.ge.1.0 ) go to 2
18:       rn=rn+1.0
19:       go to 1
20:     2 return
21:       end
```

`pornd(ix,iy,alpha,rn)` should be used together with `urnd(ix,iy,rn)` on p.83 and `exprnd(ix,iy,alpha,rn)` on p.94.

As for an alternative random number generation method, we consider the recursive relationship between x and $x-1$. From the Poisson probability function, we can obtain the following recursion:

$$f(x) = \frac{\alpha}{x} f(x-1),$$

where the initial value is $f(0) = e^{-\alpha}$. Therefore, for a uniform random draw u between zero and one, the Poisson random draw is given by the x which satisfies the following:

$$P(X \leq x-1) \leq u < P(X \leq x),$$

where $P(X \leq x) = \sum_{i=0}^{x} f(i)$. See Ross (1997, p.50). This approach corresponds to the inverse transform method shown in Figure 2.19. Thus, the Fortran program is given by `pornd2(ix,iy,alpha,rn)`.

`pornd2(ix,iy,alpha,rn)`

```
1:        subroutine pornd2(ix,iy,alpha,rn)
2:  c
3:  c  Use "pornd2(ix,iy,alpha,rn)"
4:  c  together with "urnd(ix,iy,rn)".
```

```
5: C
6: C  Input:
7: C    ix, iy:  Seeds
8: C    alpha:  Parameter
9: C  Output:
10: C    rn: Poisson Random Draw
11: C
12:       call urnd(ix,iy,rn1)
13:       rn=0.0
14:       sum0=0.0
15:       sum =0.0
16:       pr=exp(-alpha)
17:     1 sum=sum+pr
18:       if( sum0.le.rn1.and.rn1.lt.sum ) go to 2
19:       rn=rn+1.0
20:       pr=pr*alpha/rn
21:       sum0=sum
22:       go to 1
23:     2 return
24:       end
```

`pornd2(ix,iy,alpha,rn)` should be used together with `urnd(ix,iy,rn)` on p.83.

Both `pornd(ix,iy,alpha,rn)` and `pornd2(ix,iy,alpha,rn)` have computationally disadvantage, especially for large α. Ahrens and Dieter (1980) and Schmeiser and Kachitvichyanukul (1990) give efficient methods whose computation CPU times do not depend on mean α.

2.4.5 Binomial Distribution: $B(n, p)$

The binomial distribution is represented as:

$$f(x) = \begin{cases} \binom{n}{x} p^x (1-p)^{n-x} = \dfrac{n!}{x!(n-x)!} p^x (1-p)^{n-x}, & \text{for } x = 0, 1, 2, \cdots, n, \\ 0, & \text{otherwise,} \end{cases}$$

where $0 \le p \le 1$. n should be a positive integer. As for the notation on combination, note that the following equalities hold:

$$\binom{n}{x} = \frac{n!}{x!(n-x)!} = {}_nC_x,$$

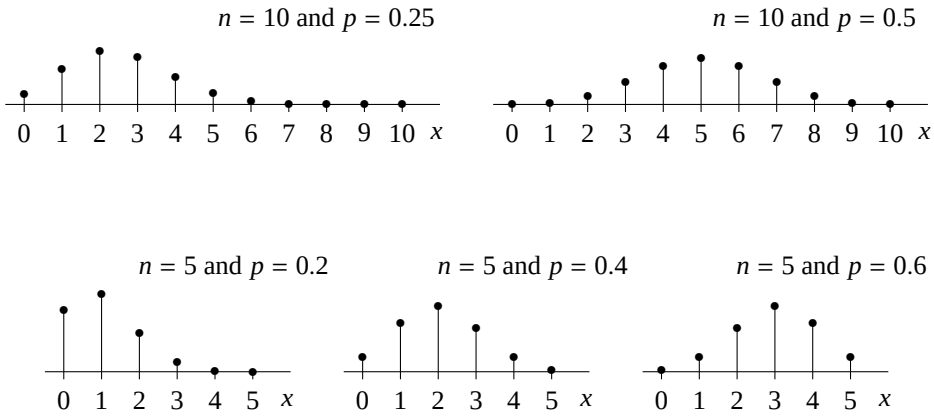
where $n! = n(n-1)(n-2)\cdots 1$. The binomial probability functions are displayed in Figure 2.24, where five cases are taken for n and p, i.e., $(n, p) = (10, 0.25), (10, 0.5), (5, 0.2), (5, 0.4), (5, 0.6)$.

Mean, variance and the moment-generating function of the binomial random variable X are given by:

$$\mathrm{E}(X) = np, \qquad \mathrm{V}(X) = np(1-p), \qquad \phi(\theta) = (1 - p + pe^{\theta})^n,$$

which are derived in Example 1.5 (p.12).

Figure 2.24: Binomial Distribution



The binomial random variable is obtained from Bernoulli random variables. We perform n experiments and x successes, where the probability of success per experiment is denoted by p. Then, the number of successes has a binomial distribution. Therefore, suppose that Y_1, Y_2, $\cdots$, Y_n are mutually independently distributed with Bernoulli random variables. Then, $X = \sum_{i=1}^{n} Y_i$ has a binomial distribution with parameters n and p. This fact is proved as follows. Let $\phi_x(\theta)$ and $\phi_i(\theta)$ be the moment-generating functions of X and Y_i, respectively. Using $\phi_i(\theta)$, $\phi_x(\theta)$ is derived as:

$$\phi_x(\theta) = \mathrm{E}(e^{\theta X}) = \mathrm{E}(e^{\theta \sum_{i=1}^{n} Y_i}) = \prod_{i=1}^{n} \mathrm{E}(e^{\theta Y_i}) = \prod_{i=1}^{n} \phi_i(\theta)$$

$$= \prod_{i=1}^{n} (1 - p + pe^{\theta}) = (1 - p + pe^{\theta})^n,$$

which is equivalent to the moment-generating function of the binomial random variable with n and p. Thus, the sum of the Bernoulli trials yields the binomial random variable. The source code is shown in `birnd(ix,iy,n,p,rn)`.

`birnd(ix,iy,n,p,rn)`

```
1:        subroutine birnd(ix,iy,n,p,rn)
2:  C
3:  C  Use "birnd(ix,iy,n,p,rn)"
4:  C  together with "brnd(ix,iy,p,rn)"
5:  C           and  "urnd(ix,iy,rn)".
6:  C
7:  C  Input:
8:  C    ix, iy:  Seeds
9:  C    n:  Number of Trials
10: C    p:  Probability of "Success"
11: C  Output:
```

```
12: c     rn: Binomial Random Draw
13: c
14:        rn=0.0
15:         do 1 i=1,n
16:        call brnd(ix,iy,p,rn1)
17:      1 rn=rn+rn1
18:        return
19:        end
```

birnd(ix,iy,n,p,rn) should be utilized together with urnd(ix,iy,rn) on p.83 and brnd(ix,iy,p,rn) on p.138.

As for an alternative random number generation method, consider the recursive relationship between x and $x-1$. From the binomial probability function, we can derive the following recursion:

$$f(x) = \frac{n-x+1}{x}\frac{p}{1-p}f(x-1),$$

where the initial value is given by $f(x) = (1-p)^n$. Let u be a uniform random draw between zero and one. We obtain the integer x which satisfies the following inequality:

$$P(X \leq x-1) \leq u < P(X \leq x),$$

where $P(X \leq x) = \sum_{i=0}^{x} f(i)$. See Ross (1997, p.52). This approach corresponds to the inverse transform method shown in Figure 2.19. Thus, the Fortran program is given by birnd2(ix,iy,n,p,rn).

birnd2(ix,iy,n,p,rn)

```
1:        subroutine birnd2(ix,iy,n,p,rn)
2: c
3: c  Use "birnd2(ix,iy,n,p,rn)"
4: c  together with "urnd(ix,iy,rn)".
5: c
6: c  Input:
7: c    ix, iy:  Seeds
8: c    n:  Number of Trials
9: c    p:  Probability of "Success"
10: c  Output:
11: c    rn: Binomial Random Draw
12: c
13:       call urnd(ix,iy,rn1)
14:       rn=0.0
15:       sum0=0.0
16:       sum =0.0
17:       pr=(1.-p)**n
18:     1 sum=sum+pr
19:       if( sum0.le.rn1.and.rn1.lt.sum ) go to 2
20:       rn=rn+1.0
21:       pr=pr*( (n-rn+1.)/rn )*( p/(1.-p) )
22:       sum0=sum
23:       go to 1
```

```
24:      2 return
25:        end
```

Note that `birnd2(ix,iy,n,p,rn)` should be used with `urnd(ix,iy,rn)` on p.83. Kachitvichyanukul and Schmeiser (1988) report that the inverse transform approach performs faster than the Bernoulli approach.

As another random number generator, the central limit theorem is utilized. $X = \sum_{i=1}^{n} Y_i$, where Y_i has a Bernoulli distribution with parameter p. $\mathrm{E}(Y_i) = p$ and $\mathrm{V}(Y_i) = p(1-p)$. Therefore, $\mathrm{E}(X/n) = p$ and $\mathrm{V}(X/n) = p(1-p)/n$. Then, the central limit theorem indicates that as $n \longrightarrow \infty$ we have:

$$\frac{X/n - p}{\sqrt{p(1-p)/n}} \longrightarrow N(0,1).$$

That is, when n is large, X is approximately normally distributed with mean np and variance $np(1-p)$. Suppose that $Z \sim N(0,1)$. We approximately have $X = np + Z\sqrt{np(1-p)}$ for large n. Since x takes an integer, x should be the integer which is close to $np + Z\sqrt{np(1-p)}$. Therefore, $X = [np + Z\sqrt{np(1-p)} + 0.5]$ is approximately taken as a binomial random variable. Thus, the source code is given by `birnd3(ix,iy,n,p,rn)`.

`birnd3(ix,iy,n,p,rn)`

```
1:        subroutine birnd3(ix,iy,n,p,rn)
2:  c
3:  c   Use "birnd3(ix,iy,n,p,rn)"
4:  c   together with "snrnd(ix,iy,rn)"
5:  c             and  "urnd(ix,iy,rn)".
6:  c
7:  c   Input:
8:  c     ix, iy:  Seeds
9:  c     n:  Number of Trials
10: c     p:  Probability of "Success"
11: c   Output:
12: c     rn: Binomial Random Draw
13: c
14:       call snrnd(ix,iy,rn1)
15:       rn=int( n*p+sqrt(n*p*(1.-p))*rn1+0.5 )
16:       return
17:       end
```

Clearly, `birnd3(ix,iy,n,p,rn)` should be used with `urnd(ix,iy,rn)` on p.83 and `snrnd(ix,iy,rn)` on p.83. Only when $n\min(p, 1-p) > 10$, the algorithm above should be used. Otherwise, the approximation of the binomial random variable becomes very poor.

In any case, it is expected that `birnd3(ix,iy,n,p,rn)` shows the worst performance of the three. As another candidate of the random number generator, we

may utilize the composition method shown in Section 3.1, which is discussed later in Section 3.1.2.

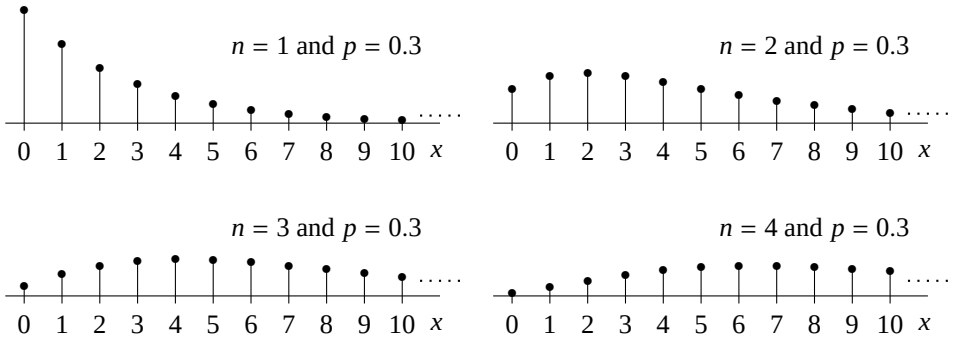
2.4.6 Negative Binomial Distribution

The negative binomial distribution with n and p is written as follows:

$$f(x) = \begin{cases} \binom{x+n-1}{x} p^n (1-p)^x = \dfrac{(x+n-1)!}{x!(n-1)!} p^n (1-p)^x, & \text{for } x = 0, 1, 2, \cdots, \\ 0, & \text{otherwise,} \end{cases}$$

where $0 \leq p \leq 1$. n should be a positive integer. The negative binomial distributions with $(n, p) = (1, 0.3), (2, 0.3), (3, 0.3), (4, 0.3)$ are drawn in Figures 2.25.

Figure 2.25: Negative Binomial Distribution



Mean, variance and the moment-generating function of the negative binomial random variable are given by:

$$\mathrm{E}(X) = \frac{n(1-p)}{p}, \qquad \mathrm{V}(X) = \frac{n(1-p)}{p^2}, \qquad \phi(\theta) = \Big(\frac{p}{1-(1-p)e^{\theta}}\Big)^n.$$

The negative binomial distribution is related to the geometric distribution (or the Pascal distribution) shown in Section 2.4.3. From the moment-generating function, the case of $n = 1$ corresponds to the geometric distribution. Let $Y_1, Y_2, \cdots, Y_n$ be mutually independently and identically distributed geometric random variables with parameter p. Suppose that the moment-generating function of Y_i is given by $\phi_i(\theta) = p/(1-(1-p)e^{\theta})$. Define $X = \sum_{i=1}^n Y_i$. Then, we derive the moment-generating function of X, denoted by $\phi_x(\theta)$.

$$\begin{aligned} \phi_x(\theta) &= \mathrm{E}(e^{\theta X}) = \mathrm{E}(e^{\theta \sum_{i=1}^n Y_i}) = \prod_{i=1}^n \mathrm{E}(e^{\theta Y_i}) = \prod_{i=1}^n \phi_i(\theta) = \prod_{i=1}^n \frac{p}{1-(1-p)e^{\theta}} \\ &= \Big(\frac{p}{1-(1-p)e^{\theta}}\Big)^n, \end{aligned}$$

which is equivalent to the moment-generating function of the negative binomial distribution, i.e., $\phi(\theta)$. Therefore, $X = \sum_{i=1}^{n} Y_i$ is distributed as a negative binomial random variable. Thus, a sum of n geometric random variables reduces to a negative binomial random variable. The negative binomial random variable X is interpreted as the number of failures to obtain n successes, where the probability of success per random experiment is given by p, i.e., $p = P(\text{success})$. The source code is presented as `nbirnd(ix,iy,n,p,rn)`.

`nbirnd(ix,iy,n,p,rn)`

```
1:       subroutine nbirnd(ix,iy,n,p,rn)
2: c
3: c  Use "nbirnd(ix,iy,p,rn)"
4: c  together with "geornd2(ix,iy,p,rn)"
5: c            and  "urnd(ix,iy,rn)".
6: c
7: c  Input:
8: c    ix, iy:  Seeds
9: c    n:  Number of "Success"
10: c    p:  Probability of "Success"
11: c  Output:
12: c    rn: Negative Binomial Random Draw
13: c
14:       rn=0.0
15:        do 1 i=1,n
16:       call geornd2(ix,iy,p,rn1)
17:     1 rn=rn+rn1
18:       return
19:       end
```

`nbirnd(ix,iy,n,p,rn)` should be used together with `geornd2(ix,iy,p,rn)` on p.140 and `urnd(ix,iy,rn)` on p.83. Since `geornd2(ix,iy,p,rn)` is much faster than `geornd(ix,iy,p,rn)`, we use `geornd2`, rather than `geornd`.

An alternative generator is obtained by utilizing the central limit theorem discussed in Section 1.6.3. As shown above, $X = \sum_{i=1}^{n} Y_i$ is distributed as the negative binomial random variable, where $\mathrm{E}(Y_i) = (1-p)/p$ and $\mathrm{V}(Y_i) = (1-p)/p^2$. Therefore, by the central limit theorem, we can obtain the fact that $\dfrac{X - n(1-p)/p}{\sqrt{n(1-p)/p^2}}$ approaches the standard normal distribution as n goes to infinity. Thus, we may take $[n(1-p)/p + Z\sqrt{n(1-p)/p^2} + 0.5]$ as the negative binomial random draw, where $Z \sim N(0,1)$. However, since this generator is not too precise for small n, we do not consider this generator in this section.

As for another random number generation method, we consider the recursive relationship between x and $x-1$. From the negative binomial probability function, we can obtain the following recursion:

$$f(x) = \frac{n+x-1}{x}(1-p)f(x-1),$$

where the initial value is $f(0) = p^n$. Therefore, for a uniform random draw u between zero and one, the Poisson random draw is given by the x which satisfies the following:

$$P(X \leq x-1) \leq u < P(X \leq x),$$

where $P(X \leq x) = \sum_{i=0}^{x} f(i)$. This approach corresponds to the inverse transform method displayed in Figure 2.19. Thus, the Fortran 77 program based on the inverse transform method is given by `nbirnd2(ix,iy,alpha,rn)`.

`nbirnd2(ix,iy,n,p,rn)`

```
1:       subroutine nbirnd2(ix,iy,n,p,rn)
2: c
3: c  Use "nbirnd2(ix,iy,n,p,rn)"
4: c  together with "urnd(ix,iy,rn)".
5: c
6: c  Input:
7: c    ix, iy:  Seeds
8: c    n:  Number of "Success"
9: c    p:  Probability of "Success"
10: c  Output:
11: c    rn: Negative Binomial Random Draw
12: c
13:       call urnd(ix,iy,rn1)
14:       rn=0.0
15:       sum0=0.0
16:       sum =0.0
17:       pr=p**n
18:     1 sum=sum+pr
19:       if( sum0.le.rn1.and.rn1.lt.sum ) go to 2
20:       rn=rn+1.0
21:       pr=pr*( (n+rn-1.)/rn )*(1.-p)
22:       sum0=sum
23:       go to 1
24:     2 return
25:       end
```

`nbirnd2(ix,iy,n,p,rn)` should be used together with `urnd(ix,iy,rn)` on p.83.

2.4.7 Hypergeometric Distribution

The probability function of the hypergeometric distribution with m, n and k is written as follows:

$$f(x) = \begin{cases} \dfrac{\binom{n}{x}\binom{m}{k-x}}{\binom{n+m}{k}} = \dfrac{\dfrac{n!}{x!(n-x)!}\dfrac{m!}{(k-x)!(m-k+x)!}}{\dfrac{(n+m)!}{k!(n+m-k)!}}, & \text{for } x = 0, 1, 2, \cdots, k, \\ 0, & \text{otherwise,} \end{cases}$$

where m, n and k are the integers which satisfies $0 \leq m$, $0 \leq n$ and $0 < k \leq n + m$.

Mean and variance of the hypergeometric random variable are obtained as:

$$\mathrm{E}(X) = k\frac{n}{n+m}, \qquad \mathrm{V}(X) = \frac{n+m-k}{n+m-1}k\frac{n}{n+m}\frac{m}{n+m}.$$

The hypergeometric random variable is interpreted as follows. Consider an urn containing $n+m$ balls, of which n are red and m are white. Choose k balls out of $n+m$ without replacement. We have X red balls out of k. Then, the number of red balls (i.e., X) has a hypergeometric distribution.

hgeornd(ix,iy,n,m,k,rn)

```
1:        subroutine hgeornd(ix,iy,n,m,k,rn)
2:  c
3:  c  Use "hgeornd(ix,iy,n,m,k,rn)"
4:  c  together with "urnd(ix,iy,rn)".
5:  c
6:  c  Input:
7:  c    ix, iy:  Seeds
8:  c    n:  Number of "Sucsess"
9:  c    m:  Number of "Fail"
10: c    k:  Number of Experiments
11: c  Output:
12: c    rn: Number of "Sucsess" out of Experiments
13: c          ,i.e.,Hypergeometric Random Draw
14: c
15:       n1=n
16:       m1=m
17:       rn=0.0
18:        do 1 i=1,k
19:       p=float(n1)/float(n1+m1)
20:       call urnd(ix,iy,rn1)
21:          if( rn1.lt.p ) then
22:       rn=rn+1.0
23:       n1=n1-1
24:          else
25:       m1=m1-1
26:          endif
27:     1 continue
28:       return
29:       end
```

hgeornd(ix,iy,n,m,k,rn) should be used with urnd(ix,iy,rn) on p.83. Note that a red ball implies “Success” in the above source code.

As for another random number generation method, we consider the recursive relationship between x and $x-1$. From the hypergeometric probability function, we can obtain the following recursion:

$$f(x) = \frac{n-x+1}{x}\frac{k-x+1}{m-k+x}f(x-1),$$

where the initial value is given by $\binom{m}{k} / \binom{n+m}{k}$. Therefore, for a uniform random draw u between zero and one, the hypergeometric random draw is given by the x which satisfies the following:

$$P(X \leq x-1) \leq u < P(X \leq x),$$

where $P(X \leq x) = \sum_{i=0}^{x} f(i)$. This approach corresponds to the inverse transform method shown in Figure 2.19. The Fortran 77 source code for the algorithm above is `hgeornd2(ix,iy,n,m,k,rn)`.

`hgeornd2(ix,iy,n,m,k,rn)`

```
      subroutine hgeornd(ix,iy,n,m,k,rn)
c
c  Use "hgeornd2(ix,iy,n,m,k,rn)"
c  together with "urnd(ix,iy,rn)".
c
c  Input:
c    ix, iy:  Seeds
c    n:  Number of "Sucsess"
c    m:  Number of "Fail"
c    k:  Number of Experiments
c  Output:
c    rn: Number of "Sucsess" out of Experiments
c         ,i.e.,Hypergeometric Random Draw
c
       call urnd(ix,iy,rn1)
       rn=0.0
       sum0=0.0
       sum =0.0
       pr=1.0
         do 1 i=1,k
     1 pr=pr*( float(m-i+1)/float(n+m-i+1) )
     2 sum=sum+pr
       if( sum0.le.rn1.and.rn1.lt.sum ) go to 3
       rn=rn+1.0
       pr=pr*( (n-rn+1.)/rn )*( (k-rn+1.)/(m-k+rn) )
       sum0=sum
         go to 2
     3 return
       end
```

`hgeornd2(ix,iy,n,m,k,rn)` is also used with `urnd(ix,iy,rn)` on p.83.

Furthermore, Kachitvichyanukul and Schmeiser (1985) give us an algorithm based on rejection sampling (Section 3.2 for rejection sampling) and Stadlober (1990) describes an algorithm based on a ratio-of-uniforms method (Section 3.5 for the ratio-of-uniforms method). Both algorithms are faster than `hgeornd(ix,iy,n,m,k,rn)` and `hgeornd2(ix,iy,n,m,k,rn)` for large n and k.

2.5 Multivariate Distribution

If $X_1, X_2, \cdots, X_k$ are mutually independent, the joint distribution of $X_1, X_2, \cdots, X_k$ is given by:

$$f(x_1, x_2, \cdots, x_k) = \prod_{i=1}^{k} f(x_i),$$

where $f(x_1, x_2, \cdots, x_k)$ denotes the joint density function of $X_1, X_2, \cdots, X_k$ and $f(x_i)$ represents the marginal density of X_i. To generate random draws of $X_1, X_2, \cdots, X_k$ from the multivariate distribution function $f(x_1, x_2, \cdots, x_k)$, we may generate random draws of X_i from $f(x_i)$ and repeat for $i = 1, 2, \cdots, k$. However, $X_1, X_2, \cdots, X_k$ are dependent with each other, we need to generate the random draws, taking into account correlation between X_i and X_j for $i \neq j$. In this section, random number generators in multivariate cases are discussed.

2.5.1 Multivariate Normal Distribution: $N(\mu, \Sigma)$

The density function of X with parameters μ and Σ, denoted by $f(x)$, is given by:

$$f(x) = \frac{1}{(2\pi)^{k/2}} |\Sigma|^{-\frac{1}{2}} \exp\Big(-\frac{1}{2}(x-\mu)'\Sigma^{-1}(x-\mu)\Big),$$

for $-\infty < x_i < \infty$, $i = 1, 2, \cdots, k$, where x denotes a $k \times 1$ vector, i.e., $x = (x_1, x_2, \cdots, x_k)'$. μ is a $k \times 1$ vector and Σ is a $k \times k$ matrix. The multivariate normal distribution has a single mode at $x = \mu$ and it is symmetric about $x = \mu$. Σ should be a positive definite matrix. See p.74 for the definition of the positive definite matrix.

Mean, variance and the moment-generating function of the multivariate normal random variable X are represented as:

$$\mathrm{E}(X) = \mu, \qquad \mathrm{V}(X) = \Sigma, \qquad \phi(\theta) = \exp\Big(\theta'\mu + \frac{1}{2}\theta'\Sigma\theta\Big),$$

where $\theta = (\theta_1, \theta_2, \cdots, \theta_k)'$. Therefore, the two parameters included in the normal distribution, i.e., μ and Σ, represent mean and variance. Accordingly, the parameters μ and Σ represent mean and variance, which is denoted by $N(\mu, \Sigma)$. When $k = 1$, the multivariate normal distribution reduces to the univariate normal distribution shown in Section 2.2.2.

The multivariate normal random draws with mean μ and variance Σ are generated as follows. Suppose that $Z_1, Z_2, \cdots, Z_k$ are mutually independently distributed as standard normal random variables. Because Σ is a positive definite matrix, there exists a $k \times k$ matrix of P which satisfies $\Sigma = PP'$, where P is a lower-triangular matrix. Let $X = \mu + PZ$, where $Z = (Z_1, Z_2, \cdots, Z_k)'$. Then, X has a normal distribution with mean μ and variance Σ. Note that we have $\mathrm{V}(X) = P\mathrm{E}(ZZ')P' = PP' = \Sigma$, because $\mathrm{E}(ZZ') = I_k$, where I_k denotes a $k \times k$ identity matrix. Thus, the k independent standard normal random numbers yield the k-variate normal random draw with mean μ

and variance Σ. The Fortran program for this random number generator is given by mnrnd(ix,iy,ave,var,k,rn), where ave and var correspond to μ and Σ, respectively.

mnrnd(ix,iy,ave,var,k,rn)

```
1:        subroutine mnrnd(ix,iy,ave,var,k,rn)
2:        dimension ave(10),var(10,10),rn(10)
3:        dimension p(10,10),z(10)
4:  c
5:  c  Use "mnrnd(ix,iy,ave,var,k,rn)"
6:  c  together with "mvar(k,var,p)",
7:  c                "snrnd(ix,iy,rn)"
8:  c            and  "urnd(ix,iy,rn)".
9:  c
10: c Input:
11: c   k:        k-variate random draw
12: c   ave(i):   vector of mean
13: c   var(i,j): matrix of variance
14: c Output:
15: c   rn(i):  k-variate normal random draw
16: c           with mean ave(i) and variance var(i,j)
17: c
18:       call mvar(k,var,p)
19:         do 1 i=1,k
20:       call snrnd(ix,iy,rn1)
21:     1 z(i)=rn1
22:         do 2 i=1,k
23:       rn(i)=ave(i)
24:          do 2 j=1,k
25:     2 rn(i)=rn(i)+p(i,j)*z(j)
26:       return
27:       end
```

mnrnd(ix,iy,ave,var,k,rn) have to be used together with urnd(ix,iy,rn) on p.83, snrnd(ix,iy,rn) on p.85 and mvar(k,var,p) which is discussed in the next paragraph.

mvar(k,var,p) represents the source code to obtain the lower-triangular matrix P which satisfies $\Sigma = PP'$, where Σ should be a positive definite and symmetric matrix (this condition is satisfied because Σ is a variance-covariance matrix). Given Σ, P is derived from the following recursion:

$$p_{ij} = \frac{\sigma_{ij} - \sum_{m=1}^{j-1} p_{im}p_{jm}}{\sqrt{\sigma_{jj} - \sum_{m=1}^{j-1} p_{jm}^2}},$$

for $1 \leq j \leq i \leq k$, where $\sum_{m=1}^{0} p_{im}p_{jm} = 0$. p_{ij} and σ_{ij} denote (i, j)th elements of P and Σ, respectively. See Fishman (1996, p.223) and Rubinstein and Melamed (1998, p.51) for the above recursion. In mvar(k,var,p), var represents Σ and p denotes P.

mvar(k,var,p)

```
1:       subroutine mvar(k,var,p)
2:       dimension var(10,10),p(10,10)
3: c
4: c Input:
5: c   k  :  Dimension of Matrix
6: c   var:  Original Matrix
7: c         (Positive Definite and Symmetric Matrix)
8: c Output:
9: c   p  :  Lower-Triangular Matrix satisfying var=pp'
10: c
11:        do 1 j=1,k
12:         do 1 i=1,j-1
13:     1 p(i,j)=0.
14:         do 2 j=1,k
15:       pjj=var(j,j)
16:           do 3 m=1,j-1
17:     3 pjj=pjj-p(j,m)*p(j,m)
18:           do 2 i=j,k
19:       p(i,j)=var(i,j)/sqrt(pjj)
20:           do 2 m=1,j-1
21:     2 p(i,j)=p(i,j)-p(i,m)*p(j,m)/sqrt(pjj)
22:       return
23:       end
```

In both mnrnd(ix,iy,ave,var,k,rn) and mvar(k,var,p), $k \leq 10$ is set. However, the dimension of all the vectors and matrices in the 2nd and 3rd lines of mnrnd(ix,iy,ave,var,k,rn) and the 2nd line of mvar(k,var,p) can take any integer. In order to simplify the arguments in the subroutines, we take a specific integer (i.e., 10 in the above source codes may be changed to the other integer) for the vectors and matrices.

We have some computational difficulty in mnrnd(ix,iy,ave,var,k,rn) because of mvar(k,var,p). In practice, we have the case where pjj in Line 19 of mvar(k,var,p) is very close to zero for the positive definite matrix Σ and accordingly we have an overflow error message in computation. To avoid this situation, we may consider obtaining P based on the eigenvectors, where P satisfies $\Sigma = PP'$ but it is not a triangular matrix. A positive definite matrix Σ can be factored into $\Sigma = C\Lambda C'$, where the columns of C are the characteristic vectors of Σ and the characteristic roots of Σ are arrayed in the diagonal matrix Λ. Let $\Lambda^{1/2}$ be the diagonal matrix with the ith diagonal element $\lambda_i^{1/2}$ and let $P = C\Lambda^{1/2}$. Then, we have $\Sigma = PP'$. Thus, we can obtain P from C and Λ. By taking this procedure, we can reduce the overflow on pjj in mvar(k,var,p). The source code is given by mnrnd2(ix,iy,ave,var,k,rn).

mnrnd2(ix,iy,ave,var,k,rn)

```
1:       subroutine mnrnd2(ix,iy,ave,var,k,rn)
2:       dimension ave(10),var(10,10),rn(10),p(10,10)
3:       dimension v(10,10),z(10),a(10,10),d(10,10)
4: c
```

```
5: c  Use "mnrnd2(ix,iy,ave,var,k,rn)"
6: c  together with "eigen(var,k,v,d)",
7: c                "snrnd(ix,iy,rn)"
8: c           and  "urnd(ix,iy,rn)".
9: c
10: c Input:
11: c   k:        k-variate random draw
12: c   ave(i):   vector of mean
13: c   var(i,j): matrix of variance
14: c Output:
15: c   rn(i):  k-variate normal random draw
16: c           with mean ave(i) and variance var(i,j)
17: c
18:       call eigen(var,k,v,d)
19:         do 1 i=1,k
20:          do 1 j=1,k
21:            if( i.eq.j) then
22:              if( d(i,j).gt.0.0 ) then
23:       a(i,j)=sqrt( d(i,j) )
24:              else
25:       a(i,j)=0.
26:              endif
27:            else
28:       a(i,j)=0.
29:            endif
30:     1 continue
31:        do 2 i=1,k
32:         do 2 j=1,k
33:       p(i,j)=0.
34:           do 2 m=1,k
35:     2 p(i,j)=p(i,j)+v(i,m)*a(m,j)
36:         do 3 i=1,k
37:     3 call snrnd(ix,iy,z(i))
38:         do 4 i=1,k
39:       rn(i)=ave(i)
40:           do 4 j=1,k
41:     4 rn(i)=rn(i)+p(i,j)*z(j)
42:       return
43:       end
```

Thus, the subroutine `mnrnd2(ix,iy,ave,var,k,rn)` utilizes `eigen(var,k,v,d)`, `snrnd(ix,iy,rn)` and `urnd(ix,iy,rn)`. `eigen(var,k,v,d)` is the subroutine which obtains `v(i,j)` and `d(i,j)` from `var(i,j)`. Note that `var(i,j)`, `v(i,j)` and `d(i,j)` correspond to Σ, C and Λ, respectively. Moreover, in Lines 33 – 35 of `mnrnd2(ix,iy,ave,var,k,rn)`, `p(i,j)` is equivalent to $P = C\Lambda^{1/2}$. To avoid computational difficulty such as overflow and underflow, `a(i,j)` is introduced in Lines 19 – 30 of `mnrnd2(ix,iy,ave,var,k,rn)`.

Using the Jacobi method, `eigen(x,k,v,d)` is shown as follows.

`eigen(x,k,v,d)`

```
1:       subroutine eigen(x,k,v,d)
2:       parameter (nmax=500)
3:       dimension x(10,10),d(10,10),v(10,10),a(10,10)
4:       dimension b(nmax),z(nmax)
```

```
5: C
6: C   Input:
7: C       x(i,j):  Original Matrix (Symmetric Matrix)
8: C       k :      Dimension of Matrix
9: C   Output:
10: C      d(i,j):  Eigen Values in Diagonal Element
11: C      v(i,j):  Orthogonal Matrix
12: C               (Eigen Vector in Each Column)
13: C
14:        do 1 ip=1,k
15:         do 1 iq=1,k
16:           if( ip.eq.iq) then
17:       v(ip,iq)=1.
18:           else
19:       v(ip,iq)=0.
20:           endif
21:       a(ip,iq)=x(ip,iq)
22:     1 d(ip,iq)=0.0
23:        do 2 ip=1,k
24:       b(ip)=a(ip,ip)
25:       d(ip,ip)=b(ip)
26:     2 z(ip)=0.
27:        do 3 i=1,50
28:       sm=0.
29:         do 4 ip=1,k-1
30:           do 4 iq=ip+1,k
31:     4 sm=sm+abs(a(ip,iq))
32:       if(sm.eq.0.) go to 5
33:            if(i.lt.4) then
34:       tresh=0.2*sm/k**2
35:            else
36:       tresh=0.
37:            endif
38:         do 6 ip=1,k-1
39:           do 6 iq=ip+1,k
40:       g=100.*abs(a(ip,iq))
41:       if((i.gt.4)
42:      & .and.(abs(d(ip,ip))+g.eq.abs(d(ip,ip)))
43:      & .and.(abs(d(iq,iq))+g.eq.abs(d(iq,iq)))) then
44:       a(ip,iq)=0.
45:            else if(abs(a(ip,iq)).gt.tresh)then
46:       h=d(iq,iq)-d(ip,ip)
47:               if(abs(h)+g.eq.abs(h))then
48:       t=a(ip,iq)/h
49:               else
50:       theta=0.5*h/a(ip,iq)
51:       t=1./(abs(theta)+sqrt(1.+theta**2))
52:                 if(theta.lt.0.)t=-t
53:               endif
54:       c=1./sqrt(1+t**2)
55:       s=t*c
56:       tau=s/(1.+c)
57:       h=t*a(ip,iq)
58:       z(ip)=z(ip)-h
59:       z(iq)=z(iq)+h
60:       d(ip,ip)=d(ip,ip)-h
61:       d(iq,iq)=d(iq,iq)+h
62:       a(ip,iq)=0.
63:               do 7 j=1,ip-1
64:       g=a(j,ip)
65:       h=a(j,iq)
66:       a(j,ip)=g-s*(h+g*tau)
67:     7 a(j,iq)=h+s*(g-h*tau)
```

```
68:                do 8 j=ip+1,iq-1
69:          g=a(ip,j)
70:          h=a(j,iq)
71:          a(ip,j)=g-s*(h+g*tau)
72:        8 a(j,iq)=h+s*(g-h*tau)
73:                do 9 j=iq+1,k
74:          g=a(ip,j)
75:          h=a(iq,j)
76:          a(ip,j)=g-s*(h+g*tau)
77:        9 a(iq,j)=h+s*(g-h*tau)
78:                do 10 j=1,k
79:          g=v(j,ip)
80:          h=v(j,iq)
81:          v(j,ip)=g-s*(h+g*tau)
82:       10 v(j,iq)=h+s*(g-h*tau)
83:             endif
84:        6   continue
85:            do 11 ip=1,k
86:          b(ip)=b(ip)+z(ip)
87:          d(ip,ip)=b(ip)
88:       11 z(ip)=0.
89:        3 continue
90:          pause 'too many iterations in Jacobi'
91:        5 return
92:          end
```

`x(i,j)` has to be a symmetric matrix. The columns of `v(i,j)` represent the characteristic vectors of `x(i,j)`. The characteristic roots of `x(i,j)` are arrayed in the diagonal matrix `d(i,j)`. See Press, Teukolsky, Vetterling and Flannery (1992b, pp.456 – 462) for `eigen(x,k,v,d)`.

Thus, because `mnrnd2(ix,iy,ave,var,k,rn)` can avoid computational difficulty, `mnrnd2` is recommended, rather than `mnrnd`.

2.5.2 Multivariate *t* Distribution

The k-variate t distribution with m degrees of freedom is written as:

$$f(x) = \frac{m^{m/2}\Gamma(\frac{m+k}{2})|\Omega|^{-1/2}}{\pi^{k/2}\Gamma(\frac{m}{2})}\Big(m + (x-\mu)'\Omega^{-1}(x-\mu)\Big)^{-(k+m)/2},$$

for $-\infty < x_i < \infty$, $i = 1, 2, \cdots, k$, where $x = (x_1, x_2, \cdots, x_k)'$. μ denotes a location parameter while Ω represents a scale parameter. Like the multivariate normal distribution, the multivariate t distribution has a single mode at $x = \mu$ and it is symmetric about $x = \mu$.

Mean and variance are given by:

$$\mathrm{E}(X) = \mu, \quad \text{for } k > 1,$$
$$\mathrm{V}(X) = \frac{k}{k-2}\Omega, \quad \text{for } k > 2.$$

When $\mu = 0$, $\Omega = I_k$ and $k = 1$, the multivariate t distribution reduces to the univariate t distribution, $t(m)$, discussed in Section 2.2.10.

The multivariate t distribution with m degrees of freedom can be derived from Bayesian procedure, which is as follows. Let X be a vector of normal random variable with mean μ and variance $\sigma^2\Omega$. In Bayesian statistics, σ is assumed to be a random variable, where the conditional distribution of X given σ is assumed to be normal. From Section 2.5.1, the multivariate normal density function of X, $f_x(x|\sigma)$, is given by:

$$f_x(x|\sigma) = \frac{1}{(2\pi\sigma^2)^{k/2}}|\Omega|^{-1/2}\exp\Big(-\frac{1}{2\sigma^2}(x-\mu)'\Omega^{-1}(x-\mu)\Big).$$

Suppose that the density function of σ, $f_\sigma(\sigma)$, is:

$$f_\sigma(\sigma) = \frac{2}{\Gamma(\frac{m}{2})}\Big(\frac{m}{2}\Big)^{m/2}\frac{1}{\sigma^{m+1}}\exp\Big(-\frac{m}{2\sigma^2}\Big),$$

which is an inverse gamma distribution with parameters $m > 0$ and $s = 1$ in equation (2.1) on p.98. Multiplying $f_x(x|\sigma)$ and $f_\sigma(\sigma)$, the joint density of X and σ, $f_{x\sigma}(x,\sigma)$, is obtained as follows:

$$\begin{aligned} f_{x\sigma}(x,\sigma) &= f_x(x|\sigma)\,f_\sigma(\sigma) \\ &= \frac{1}{(2\pi\sigma^2)^{k/2}}|\Omega|^{-1/2}\exp\Big(-\frac{1}{2\sigma^2}(x-\mu)'\Omega^{-1}(x-\mu)\Big) \\ &\quad\times\frac{2}{\Gamma(\frac{m}{2})}\Big(\frac{m}{2}\Big)^{m/2}\frac{1}{\sigma^{m+1}}\exp\Big(-\frac{m}{2\sigma^2}\Big) \\ &\propto \frac{1}{\sigma^{m+k+1}}\exp\left(-\frac{1}{2\sigma^2}\Big(m+(x-\mu)'\Omega^{-1}(x-\mu)\Big)\right), \end{aligned}$$

where "$A \propto B$" implies "A is proportional to B". Integrating $f_{x\sigma}(x,\sigma)$ with respect to σ, we can obtain the multivariate t distribution. That is, the multivariate t is given by:

$$f(x) = \int_0^\infty f_{x\sigma}(x,\sigma)\,\mathrm{d}\sigma \propto \Big(m+(x-\mu)'\Omega^{-1}(x-\mu)\Big)^{-(k+m)/2}.$$

Thus, the multivariate t distribution is derived in a Bayesian framework, where the unknown parameter σ is taken as a random variable. See Zellner (1971, pp.383 – 389) for the multivariate t distribution.

Therefore, in order to generate a multivariate t random draw with μ, Ω and k, first, σ is generated from the inverse gamma distribution $f_\sigma(\sigma)$, and second, given a random draw of σ, x is generated from the multivariate normal distribution with mean μ and variance $\sigma^2\Omega$. The generated x is regarded as a multivariate t random draw. The Fortran 77 source code for the multivariate t random number generator is given by `mtrnd(ix,iy,ave,var,k,m,rn)`. Note that `ave(i)`, `var(i,j)`, `k` and `m` correspond to μ, Ω, k and m, respectively.

mtrnd(ix,iy,ave,var,k,m,rn)

```
      subroutine mtrnd(ix,iy,ave,var,k,m,rn)
      dimension ave(10),var(10,10),rn(10)
      dimension var0(10,10)
c
c  Use "mtrnd(ix,iy,ave,var,k,m,rn)"
c  together with "mnrnd2(ix,iy,ave,var,k,rn)",
c                "eigen(x,k,v,d)",
c                "igammarnd(ix,iy,alpha,beta,rn)",
c                "exprnd(ix,iy,beta,rn)",
c                "snrnd(ix,iy,rn)"
c           and  "urnd(ix,iy,rn)".
c
c Input:
c   k:        k-variate random draw
c   ave(i):   vector of mean
c   var(i,j): matrix of variance
c   m:        Degree of Freedom
c Output:
c   rn(i):  k-variate multivariate t random draw
c           with m degrees of freedom,
c           mean ave(i) and variance var(i,j)
c
      call igammarnd(ix,iy,m,1.0,rn1)
       do 1 i=1,k
        do 1 j=1,k
    1 var0(i,j)=rn1*var(i,j)
      call mnrnd2(ix,iy,ave,var0,k,rn)
      return
      end
```

As it is seen from the source code above, `mtrnd(ix,iy,ave,var,k,m,rn)` should be utilized with the subroutines `urnd(ix,iy,rn)` on p.83, `snrnd(ix,iy,rn)` on p.85, `exprnd(ix,iy,beta,rn)` on p.94, `igammarnd(ix,iy,alpha,beta,rn)` on p.97, `mnrnd2(ix,iy,ave,var,k,rn)` on p.154 and `eigen(x,k,v,d)` on p.155. `igammarnd(ix,iy,alpha,beta,rn)` in the subroutine `mtrnd` may be replaced by more efficient inverse gamma random number generator, which will be discussed in Section 3.5.2 (p.213), where the gamma random number generator represented by `gammarnd8(ix,iy,alpha,beta,rn)` is utilized to obtain the inverse gamma random number generator.

2.5.3 Wishart Distribution: $W(n, \Sigma)$

The Wishart distribution with parameters n and Σ is represented by:

$$f(\Omega) = \frac{1}{2^{\frac{nk}{2}} \pi^{\frac{k(k-1)}{4}} \prod_{i=1}^{k} \Gamma(\frac{n-i+1}{2})} |\Sigma|^{-\frac{n}{2}} |\Omega|^{\frac{n-k-1}{2}} \exp\Big(-\mathrm{tr}(\frac{1}{2}\Sigma^{-1}\Omega)\Big),$$

which is simply written as $\Omega \sim W(n, \Sigma)$. Σ denotes a $k \times k$ symmetric matrix. Ω is a $k \times k$ symmetric matrix, which is a random variable. Therefore, Ω has $k(k+1)/2$

distinct elements.

Let us denote the (i, j)th element of Ω by ω_{ij} and the (i, j)th element of Σ by σ_{ij}. Mean, variance and covariance of Ω are given by:

$$\begin{aligned}
&\mathrm{E}(\omega_{ij}) = n\sigma_{ij}, \qquad \mathrm{V}(\omega_{ij}) = n(\sigma_{ij}^2 + \sigma_{ii}\sigma_{jj}), \\
&\mathrm{Cov}(\omega_{ij}, \omega_{ml}) = n(\sigma_{im}\sigma_{jl} + \sigma_{il}\sigma_{jm}).
\end{aligned}$$

Note that $\sigma_{ij} = \sigma_{ji}$ because Σ is symmetric. The Wisher distribution has some features, which are shown as follows:

1. When $k = 1$ and $\Sigma = I_k$, the Wishart distribution $W(n, \Sigma)$ reduces to the chi-square distribution $\chi^2(n)$. Moreover, for $\Omega \sim W(n, \Sigma)$, we have $\omega_{ii}/\sigma_{ii} \sim \chi^2(n)$ for $i = 1, 2, \cdots, k$.

2. Let X_i be a k-variate random variable for $i = 1, 2, \cdots, n$. If X_1, X_2, $\cdots$, X_n are mutually independently distributed as $X_i \sim N(0, \Sigma)$ for $i = 1, 2, \cdots, n$, then $\Omega = \sum_{i=1}^n X_i X_i'$ has a Wishart distribution with n and Σ, i.e., $\Omega \sim W(n, \Sigma)$.

3. Suppose that Ω and Σ are partitioned as follows:

$$\begin{aligned}
\Omega &= \begin{pmatrix} \Omega_{11} & \Omega_{12} \\ \Omega_{21} & \Omega_{22} \end{pmatrix}, \qquad \Omega^{-1} = \begin{pmatrix} \Omega^{11} & \Omega^{12} \\ \Omega^{21} & \Omega^{22} \end{pmatrix}, \\
\Sigma &= \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}, \qquad \Sigma^{-1} = \begin{pmatrix} \Sigma^{11} & \Sigma^{12} \\ \Sigma^{21} & \Sigma^{22} \end{pmatrix},
\end{aligned}$$

   where Ω_{11} and Σ_{11} denote $k_1 \times k_1$ matrices, while Ω_{22} and Σ_{22} represent $k_2 \times k_2$ matrices, where $k = k_1 + k_2$. Further, define $\Omega_{11.2}$ and $\Sigma_{11.2}$ as follows:

$$\begin{aligned}
\Omega_{11.2} &= \Omega_{11} - \Omega_{12}\Omega_{22}^{-1}\Omega_{21}, \qquad \Omega_{11.2}^{-1} = \Omega^{11}, \\
\Sigma_{11.2} &= \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}, \qquad \Sigma_{11.2}^{-1} = \Sigma^{11}.
\end{aligned}$$

   Then, $\Omega_{11} \sim W(n, \Sigma_{11})$ and $\Omega_{11.2} \sim W(n - k_2, \Sigma_{11.2})$ hold.

4. Furthermore, suppose that $X_1, X_2, \cdots, X_n$ are mutually independently distributed as $X_i \sim N(\mu, \Sigma)$ for $i = 1, 2, \cdots, n$. Define $\overline{X} = (1/n)\sum_{i=1}^n X_i$ and $S = \sum_{i=1}^n (X_i - \overline{X})(X_i - \overline{X})'/(n-1)$. Then, $(n-1)S$ has a Wishart distribution with $n-1$ and Σ, i.e., $(n-1)S \sim W(n-1, \Sigma)$, and S is independent of $\overline{X}$.

5. For $\Omega \sim W(n, \Sigma)$, it is known that $|\Omega|$ has the following distribution:

$$\frac{|\Omega|}{|\Sigma|} \sim \chi^2(n) \times \chi^2(n-1) \times \cdots \times \chi^2(n-k+1),$$

   where $|\Omega|/|\Sigma|$ is distributed as a product of k independent chi-square random variables, $\chi^2(n), \chi^2(n-1), \cdots, \chi^2(n-k+1)$.

6. As for S, we have $(n-1)S \sim W(n-1,\Sigma)$, which is discussed above. Therefore, the distribution of $|S|$ is given by:

$$\frac{(n-1)^k|S|}{|\Sigma|} \sim \chi^2(n-1)\times\chi^2(n-2)\times\cdots\times\chi^2(n-k),$$

which is also represented by a product of k independent chi-square random variables, i.e., $\chi^2(n-1), \chi^2(n-2), \cdots, \chi^2(n-k)$.

7. We can introduce another feature of the Wishart random variable. When $V = \sum_{i=1}^{n} Z_i Z_i'$ for $Z_i \sim N(0, I_k)$, $i = 1, 2, \cdots, n$, V is known to be simulated as:

$$V_{ii} = x_i + \sum_{m<i} \epsilon_{mi}^2,$$
$$V_{ij} = V_{ji} = \epsilon_{ij}\sqrt{x_i} + \sum_{m<i} \epsilon_{mi}\epsilon_{mj}, \qquad \text{for } i < j,$$

which is called Bartlett's decomposition. V_{ij} denotes the (i, j)th element of V, $x_i \sim \chi^2(n+1-i)$ for $i = 1, 2, \cdots, k$ and $\epsilon_{ij} \sim N(0,1)$ for $1 \le i \le j \le m$. The relationship between Ω and V is given by $\Omega = PVP'$, where P is the lower triangular matrix which satisfies $\Sigma = PP'$.

Thus, the Wishart distribution $W(n,\Sigma)$ has a lot of properties. See Gentle (1998), Odell and Feiveson (1966), Ripley (1987) and Smith and Hocking (1972) for the discussion above. Here, as it is shown above, utilizing $\Omega = \sum_{i=1}^{n} X_i X_i' \sim W(n,\Sigma)$ for $X_i \sim N(0,\Sigma)$, the Fortran source code for the Wishart random number generator with n and Σ is presented as `wishartrnd(ix,iy,n,k,var,rn)`, where `n`, `k` and `var` implies n, k and Σ, respectively. The random draw of Ω is given by `rn`.

`wishartrnd(ix,iy,n,k,var,rn)`

```
1:       subroutine wishartrnd(ix,iy,n,k,var,rn)
2:       dimension var(10,10),rn(10,10),z(10)
3: c
4: c  Use "wishartrnd(ix,iy,n,k,var,rn)"
5: c  together with "mnrnd2(ix,iy,ave,var,k,rn)"
6: c                "eigen(x,k,v,d)"
7: c                "snrnd(ix,iy,rn)"
8: c           and  "urnd(ix,iy,rn)",
9: c
10: c Input:
11: c   k:        Dimension of var(i,j)
12: c   n:        Parameter
13: c   var(i,j): Parameter
14: c Output:
15: c   rn(i,j): (kxk)-variate Wishart random draw
16: c             with n, k and var(i,j)
17: c
18:        do 1 i=1,k
19:       ave(i)=0.0
```

```
      do 1 j=1,k
    1 rn(i,j)=0.0
      do 2 m=1,n
      call mnrnd2(ix,iy,ave,var,k,z)
      do 2 i=1,k
      do 2 i=1,k
    2 rn(i,j)=rn(i,j)+z(i)*z(j)
      return
      end
```

As is is seen from this source code, note that `wishartrnd(ix,iy,n,k,var,rn)` should be utilized together with `urnd(ix,iy,rn)` on p.83, `snrnd(ix,iy,rn)` on p.85, `mnrnd2(ix,iy,ave,var,k,rn)` on p.154 and `eigen(x,k,v,d)` on p.155.

The above source code is very simple, but it gives us an inefficient algorithm as n increases. Therefore, the random number generator based on the Bartlett decomposition is recommended.

2.5.4 Dirichlet Distribution

The Dirichlet distribution with parameters $\alpha_1, \alpha_2, \cdots, \alpha_{k+1}$ is represented as follows:

$$f(x_1, x_2, \cdots, x_k) = \begin{cases} c x_1^{\alpha_1 - 1} x_2^{\alpha_2 - 1} \cdots x_k^{\alpha_k - 1} (1 - x_1 - x_2 - \cdots - x_k)^{\alpha_{k+1} - 1}, \\ \qquad \text{for } 0 < x_i,\ i = 1, 2, \cdots, k, \text{ and } x_1 + x_2 + \cdots + x_k < 1, \\ 0, \qquad \text{otherwise,} \end{cases}$$

where c is given by:

$$c = \frac{\Gamma(\alpha_1 + \alpha_2 + \cdots + \alpha_{k+1})}{\Gamma(\alpha_1)\Gamma(\alpha_2) \cdots \Gamma(\alpha_{k+1})}.$$

When $k = 1$, the Dirichlet distribution reduces to the beta distribution with parameters α_1 and α_2.

Mean, variance and covariance of $X_1, X_2, \cdots, X_k$ are given by:

$$\mathrm{E}(X_i) = \frac{\alpha_i}{\sum_{m=1}^{k+1} \alpha_m},$$

$$\mathrm{V}(X_i) = \frac{\alpha_i(-\alpha_i + \sum_{m=1}^{k+1} \alpha_m)}{(\sum_{m=1}^{k+1} \alpha_m)^2 (1 + \sum_{m=1}^{k+1} \alpha_m)},$$

$$\mathrm{Cov}(X_i, X_j) = -\sqrt{\frac{\alpha_i \alpha_j}{(-\alpha_i + \sum_{m=1}^{k+1} \alpha_m)(-\alpha_j + \sum_{m=1}^{k+1} \alpha_m)}}.$$

See Kotz, Balakrishman and Johnson (2000e, pp.485 – 491) for the Dirichlet distribution.

The Dirichlet distribution with parameters $\alpha_1, \alpha_2, \cdots, \alpha_{k+1}$ is derived as follows. Let $Z_1, Z_2, \cdots, Z_{k+1}$ be mutually independent random variables. Z_i has a gamma distribution with parameters $\alpha = \alpha_i$ and $\beta = 1$ in Section 2.2.5. Define:

$$X_i = \frac{Z_i}{Z_1 + Z_2 + \cdots + Z_{k+1}},$$

for $0 < X_i < 1$, $i = 1, 2, \cdots, k$. Then, the joint density of $X_1, X_2, \cdots, X_k$ is given by the Dirichlet distribution with parameters $\alpha_1, \alpha_2, \cdots, \alpha_{k+1}$. This fact is proved as follows. Since Z_i has a gamma distribution with parameters $\alpha = \alpha_i$ and $\beta = 1$, the density function of Z_i, denoted by $f_i(z_i)$, is given by:

$$f_i(z_i) = \frac{1}{\Gamma(\alpha_i)} z_i^{\alpha_i - 1} e^{-z_i}.$$

Therefore, since $Z_1, Z_2, \cdots, Z_{k+1}$ are assumed to be mutually independent, we can construct the joint density of $Z_1, Z_2, \cdots, Z_{k+1}$ as:

$$\begin{aligned} f_z(z_1, z_2, \cdots, z_{k+1}) &= f_1(z_1) f_2(z_2) \cdots f_{k+1}(z_{k+1}) \\ &= \prod_{i=1}^{k+1} \frac{1}{\Gamma(\alpha_i)} z_i^{\alpha_i - 1} e^{-z_i}, \end{aligned}$$

for $0 < z_i < \infty$, $i = 1, 2, \cdots, k+1$. In addition to $X_i = Z_i/(Z_1 + Z_2 + \cdots + Z_{k+1})$ for $i = 1, 2, \cdots, k$, we define:

$$X_{k+1} = Z_1 + Z_2 + \cdots + Z_{k+1},$$

for $0 < X_{k+1} < \infty$. Then, by the inverse transformation, we obtain:

$$\begin{aligned} z_i &= x_i x_{k+1}, \qquad i = 1, 2, \cdots, k, \\ z_{k+1} &= x_{k+1}(1 - x_1 - x_2 - \cdots - x_k). \end{aligned}$$

Therefore, the Jacobian is represented as:

$$\begin{aligned} J &= \begin{vmatrix} \dfrac{\partial z_1}{\partial x_1} & \dfrac{\partial z_1}{\partial x_2} & \cdots & \dfrac{\partial z_1}{\partial x_{k+1}} \\ \dfrac{\partial z_2}{\partial x_1} & \dfrac{\partial z_2}{\partial x_2} & \cdots & \dfrac{\partial z_2}{\partial x_{k+1}} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial z_{k+1}}{\partial x_1} & \dfrac{\partial z_{k+1}}{\partial x_2} & \cdots & \dfrac{\partial z_{k+1}}{\partial x_{k+1}} \end{vmatrix} \\ &= \begin{vmatrix} x_{k+1} & 0 & \cdots & 0 & x_1 \\ 0 & x_{k+1} & \cdots & 0 & x_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & x_{k+1} & x_k \\ -x_{k+1} & -x_{k+1} & \cdots & -x_{k+1} & (1 - x_1 - \cdots - x_k) \end{vmatrix} \\ &= x_{k+1}^k. \end{aligned}$$

Hence the joint density of X_1, X_2, $\cdots$, X_{k+1}, denoted by $f_x(x_1, x_2, \cdots, x_{k+1})$, is given by:

$$\begin{aligned}
&f_x(x_1, x_2, \cdots, x_{k+1}) \\
&\quad = |J| f_z(x_1 x_{k+1}, x_2 x_{k+1}, \cdots, x_k x_{k+1}, x_{k+1}(1 - x_1 - x_2 - \cdots - x_k)) \\
&\quad = x_{k+1}^k \frac{1}{\Gamma(\alpha_{k+1})} \Big(x_{k+1}(1 - x_1 - x_2 - \cdots - x_k)\Big)^{\alpha_{k+1}-1} e^{-x_{k+1}(1-x_1-x_2-\cdots-x_k)} \\
&\qquad\quad \times \prod_{i=1}^{k} \frac{1}{\Gamma(\alpha_i)} (x_i x_{k+1})^{\alpha_i - 1} e^{-x_i x_{k+1}} \\
&\quad = \frac{x_1^{\alpha_1 - 1} \cdots x_k^{\alpha_k - 1} (1 - x_1 - x_2 - \cdots - x_k)^{\alpha_{k+1}-1}}{\Gamma(\alpha_1) \cdots \Gamma(\alpha_k) \Gamma(\alpha_{k+1})} x_{k+1}^{\alpha_1 + \cdots + \alpha_k - 1} e^{-x_{k+1}}.
\end{aligned}$$

Integrating $f_x(x_1, \cdots, x_k, x_{k+1})$ with respect to x_{k+1}, the Dirichlet distribution $f(x_1, x_2, \cdots, x_k)$ is obtained as follows:

$$\begin{aligned}
&f(x_1, x_2, \cdots, x_k) \\
&\quad = \int_0^\infty f_x(x_1, x_2, \cdots, x_{k+1}) \, \mathrm{d}x_{k+1} \\
&\qquad = \frac{\Gamma(\alpha_1 + \cdots + \alpha_k)}{\Gamma(\alpha_1) \cdots \Gamma(\alpha_k) \Gamma(\alpha_{k+1})} x_1^{\alpha_1 - 1} \cdots x_k^{\alpha_k - 1} (1 - x_1 - x_2 - \cdots - x_k)^{\alpha_{k+1}-1} \\
&\qquad\quad \times \int_0^\infty \frac{1}{\Gamma(\alpha_1 + \cdots + \alpha_k)} x_{k+1}^{\alpha_1 + \cdots + \alpha_k - 1} e^{-x_{k+1}} \, \mathrm{d}x_{k+1} \\
&\quad = \frac{\Gamma(\alpha_1 + \cdots + \alpha_k)}{\Gamma(\alpha_1) \cdots \Gamma(\alpha_k) \Gamma(\alpha_{k+1})} x_1^{\alpha_1 - 1} \cdots x_k^{\alpha_k - 1} (1 - x_1 - x_2 - \cdots - x_k)^{\alpha_{k+1}-1},
\end{aligned}$$

where the integration above is equivalent to the gamma distribution with parameters $\alpha = \sum_{i=1}^k \alpha_i$ and $\beta = 1$. Thus, $k + 1$ gamma random draws yield a Dirichlet random draw, which source code is shown in `dirichletrnd(ix,iy,alpha,k,rn)`, where `alpha(i)` denotes a vector of parameters α_1, α_2, $\cdots$, α_{k+1}.

`dirichletrnd(ix,iy,alpha,k,rn)`

```
      subroutine dirichletrnd(ix,iy,alpha,k,rn)
      dimension alpha(11),rn(10)
      dimension z(11)
c
c  Use "dirichletrnd(ix,iy,alpha,k,rn)"
c  together with "gammarnd(ix,iy,alpha,beta,rn)",
c                "exprnd(ix,iy,alpha,rn)"
c           and  "urnd(ix,iy,rn)".
c
c Input:
c   k:         k-variate random draw
c   alpha(i): (k+1) vector of Parameters
c Output:
c   rn(i):  k-variate Dirichlet random draw
```

```
15: c            with parameters r(i), i=1,...,k+1.
16: c
17:       sum=0.0
18:        do 1 i=1,k+1
19:       call gammarnd(ix,iy,alpha(i),1.0,z(i))
20:     1 sum=sum+z(i)
21:        do 2 i=1,k
22:     2 rn(i)=z(i)/sum
23:       return
24:       end
```

Finally, note that `dirichletrnd(ix,iy,alpha,k,rn)` should be utilized simultaneously together with `urnd(ix,iy,rn)` on p.83, `exprnd(ix,iy,alpha,rn)` on p.94 and `gammarnd(ix,iy,alpha,beta,rn)` on p.96.

`gammarnd(ix,iy,alpha,beta,rn)` in the subroutine `dirichletrnd` may be replaced by more efficient gamma random number generator, which will be discussed in Section 3.5.2 (p.213), where `gammarnd8(ix,iy,alpha,beta,rn)` is introduced for the gamma random number generator.

2.5.5 Multinomial Distribution

In Sections 2.5.1 – 2.5.4, continuous multivariate distributions have been introduced. In this section we introduce a representative discrete multivariate distribution, which is the multinomial distribution shown as follows:

$$f(x_1, x_2, \cdots, x_{k-1}) = \begin{cases} \dfrac{n!}{x_1! \cdots x_{k-1}! x_k!} p_1^{x_1} \cdots p_{k-1}^{x_{k-1}} p_k^{x_k}, \\ \qquad\qquad \text{for } x_i = 0, 1, 2, \cdots, n \text{ and } \sum_{i=1}^{n} x_i = n, \\ 0, \qquad\qquad \text{otherwise,} \end{cases}$$

where $\sum_{i=1}^{k} p_i = 1$. When $k = 2$, the multinomial distribution shown above reduces to the binomial distribution with parameters n and p_1. When $k = 3$, the multinomial distribution is called the trinomial distribution with n, p_1 and p_2 (note that $p_3 = 1 - p_1 - p_2$).

Mean, variance, covariance and the moment-generating function are given by:

$$\mathrm{E}(X_i) = np_i, \qquad \mathrm{V}(X_i) = np_i(1 - p_i), \qquad \mathrm{Cov}(X_i, X_j) = -np_i p_j,$$
$$\phi(\theta_1, \theta_2, \cdots, \theta_{k-1}) = (p_1 e^{\theta_1} + \cdots + p_{k-1} e^{\theta_{k-1}} + p_k)^n.$$

The multinomial random draws are obtained in the following way. Suppose that we have mutually exclusive events $A_1, A_2, \cdots, A_k$. Let p_i be the probability which the event A_i occurs in one trial, i.e., $p_i = P(A_i)$, where $\sum_{i=1}^{k} p_i = 1$. We perform n independent trials. Let X_i be the number which the event A_i occurs in the n trials, where $\sum_{i=1}^{k} X_i = n$. The joint density of $X_1, X_2, \cdots, X_{k-1}$ is the multinomial distribution. Therefore, the Fortran source code of the multinomial random number generator with parameters n, $p_1, p_2, \cdots, p_{k-1}$ is given by `multirnd(ix,iy,n,k,p,rn)`.

multirnd(ix,iy,n,k,p,rn)

```
1:        subroutine multirnd(ix,iy,n,k,p,rn)
2:        dimension p(100),rn(100),pr(0:101)
3:  c
4:  c  Use "multirnd(ix,iy,n,p,rn)"
5:  c  together with "urnd(ix,iy,rn)".
6:  c
7:  c  Input:
8:  c    ix, iy:  Seeds
9:  c    n:  Number of Experimemts
10: c    k:  Number of Events
11: c    p(i):  Probability which the i-th event occurs
12: c  Output:
13: c    rn(i): Multinomial Random Draw
14: c
15:       pr(0)=0.0
16:        do 1 i=1,k
17:       rn(i)=0.0
18:     1 pr(i)=pr(i-1)+p(i)
19:        do 2 i=1,n
20:       call urnd(ix,iy,r1)
21:        do 2 j=1,k
22:     2 if(pr(j-1).le.r1.and.r1.lt.pr(j))
23:      &   rn(i)=rn(i)+1.
24:       return
25:       end
```

multirnd(ix,iy,n,k,p,rn) should be used together with urnd(ix,iy,rn) on p.83.

The alternative generator is based on a sequential procedure. The marginal distribution of X_i, denoted by $f_i(x_i)$, is a binomial distribution with parameters n and p_i, i.e.,

$$\begin{aligned} f_i(x_i) &= \sum_{x_1} \cdots \sum_{x_{i-1}} \sum_{x_{i+1}} \cdots \sum_{x_{k-1}} f(x_1, x_2, \cdots, x_{k-1}) \\ &= \frac{n!}{x_i!(n-x_i)!} p_i^{x_i} (1-p_i)^{n-p_i}. \end{aligned}$$

The conditional distribution of X_i given $X_1, X_2, \cdots, X_{i-1}$, denoted by $f_{i|i-1}(x_i|x_1, x_2, \cdots, x_{i-1})$, is represented as:

$$f_{i|i-1}(x_i|x_1, x_2, \cdots, x_{i-1}) = \binom{n - \sum_{i=1}^{j-1} x_i}{x_j} q_j^{x_j} (1-q_j)^{n-\sum_{i=1}^{j} x_i},$$

where

$$q_j = \frac{p_j}{1 - \sum_{i=1}^{j-1} p_i},$$

for $2 \le j \le k$. The joint density of $f(x_1, x_2, \cdots, x_{k-1})$ is rewritten as:

$$f(x_1, x_2, \cdots, x_{k-1})$$

$$
\begin{aligned}
&= f_{k-1|k-2}(x_{k-1}|x_1, x_2, \cdots, x_{k-2}) f(x_1, x_2, \cdots, x_{k-2}) \\
&= f_{k-1|k-2}(x_{k-1}|x_1, x_2, \cdots, x_{k-2}) f_{k-2|k-3}(x_{k-2}|x_1, x_2, \cdots, x_{k-3}) f(x_1, x_2, \cdots, x_{k-3}) \\
&\quad \vdots \\
&= f(x_1) \prod_{i=2}^{k-1} f_{i|i-1}(x_i|x_1, x_2, \cdots, x_{i-1}).
\end{aligned}
$$

Accordingly, x_1 is generated from a binomial distribution with parameters n and p_1, x_j is generated from a binomial distribution with parameters $n - \sum_{i=1}^{j-1} x_i$ and $p_j/(1 - \sum_{i=1}^{j-1} p_i)$, and repeat for $j = 2, 3, \cdots, k-1$. Thus, the multinomial random draws x_1, x_2, $\cdots$, x_{k-1} are sequentially generated. See Fishman (1996) for the sequential approach. However, here we do not show the source code based on the sequential procedure.

References

Ahrens, J.H. and Dieter, U., 1980, "Sampling from Binomial and Poisson Distributions: A Method with Bounded Computation Times," *Computing*, Vol.25, pp.193 – 208.

Ahrens, J.H. and Dieter, U., 1988, "Efficient, Table-Free Sampling Methods for the Exponential, Cauchy and Normal Distributions," *Communications of the ACM*, Vol.31, pp.1330 – 1337.

Bays, C. and Durham, S.D., 1976, "Improving a Poor Random Number Generator," *ACM Transactions on Mathematical Software*, Vol.2, pp.59 – 64.

Box, G.E.P. and Muller, M.E., 1958, "A Note on the Generation of Random Normal Deviates," *Annals of Mathematical Statistics*, Vol.29, No.2, pp.610 – 611.

Cheng, R.C.H., 1998, "Random Variate Generation," in *Handbook of Simulation*, Chap.5, edited by Banks, J., pp.139 – 172, John Wiley & Sons.

De Matteis, A. and Pagnutti, S., 1993, "Long-Range Correlation Analysis of the Wichmann-Hill Random Number Generator," *Statistics and Computing*, Vol.3, pp.67 – 70.

Fishman, G.S., 1996, *Monte Carlo: Concepts, Algorithms, and Applications*, Springer-Verlag.

Gentle, J.E., 1998, *Random Number Generation and Monte Carlo Methods*, Springer-Verlag.

Hastings, C., 1955, *Approximations for Digital Computers*, Princeton University Press.

Hill, I.D and Pike, A.C., 1967, "Algorithm 2999: Chi-Squared Integral," *Communications of the ACM*, Vol.10, pp.243 – 244.

Hogg, R.V. and Craig, A.T., 1995, *Introduction to Mathematical Statistics* (Fifth Edition), Prentice Hall.

Johnson, N.L. and Kotz, S., 1970a, *Continuous Univariate Distributions*, Vol.1, John Wiley & Sons.

Johnson, N.L. and Kotz, S., 1970b, *Continuous Univariate Distributions*, Vol.2, John Wiley & Sons.

Kachitvichyanukul, V. and Schmeiser, B., 1985, "Computer Generation of Hypergeometric Random Variates," *Journal of Statistical Computation and Simulation*, Vol.22, pp.127 – 145.

Kennedy, Jr. W.J. and Gentle, J.E., 1980, *Statistical Computing* (Statistics: Textbooks and Monographs, Vol.33), Marcel Dekker.

Knuth, D.E., 1981, *The Art of Computer Programming, Vol.2: Seminumerical Algorithms* (Second Edition), Addison-Wesley, Reading, MA.

Kotz, S. and Johnson, N.L., 1982, *Encyclopedia of Statistical Sciences*, Vol.2, pp.188 – 193, John Wiley & Sons.

Kotz, S., Balakrishman, N. and Johnson, N.L., 2000a, *Univariate Discrete Distributions* (Second Edition), John Wiley & Sons.

Kotz, S., Balakrishman, N. and Johnson, N.L., 2000b, *Continuous Univariate Distributions, Vol.1* (Second Edition), John Wiley & Sons.

Kotz, S., Balakrishman, N. and Johnson, N.L., 2000c, *Continuous Univariate Distributions, Vol.2* (Second Edition), John Wiley & Sons.

Kotz, S., Balakrishman, N. and Johnson, N.L., 2000d, *Discrete Multivariate Distributions* (Second Edition), John Wiley & Sons.

Kotz, S., Balakrishman, N. and Johnson, N.L., 2000e, *Continuous Multivariate Distributions, Vol.1* (Second Edition), John Wiley & Sons.

Law, A.M. and Kelton, W.D., 2000, *Simulation Modeling and Analysis* (Third Edition), McGraw-Hill Higher Education.

L'Ecuyer, P., 1988, "Efficient and Portable Combined Random Number Generators," *Communications of the ACM*, Vol.31, No.6, pp.742 – 749.

L'Ecuyer, P., 1990, "Random Numbers for Simulation," *Communications of the ACM*, Vol.33, No.10, pp.85 – 97.

L'Ecuyer, P., 1998, "Random Number Generation," in *Handbook of Simulation*, Chap. 4, edited by Banks, J., pp.93 – 137, John Wiley & Sons.

Marsaglia, G., 1964, "Generating a Variable from the Tail of the Normal Distribution," *Technometrics*, Vol.6, pp.101 – 102.

Marsaglia, G., MacLaren, M.D. and Bray, T.A., 1964, "A Fast Method for Generating Normal Random Variables," *Communications of the ACM*, Vol.7, pp.4 – 10.

Marsaglia, G. and Zaman, A., 1994, "Rapid Evaluation of the Inverse of the Normal Distribution Function," *Statistics and Probability Letters*, Vol.19, No.2, pp.259 – 266.

Niederreiter, H., 1992, *Random Number Generation and Quasi-Monte Carlo Methods* (CBMS-NFS Regional Conference Series in Applied Mathematics 63), Society for Industrial and Applied Mathematics.

Odeh, R.E. and Evans, J.O., 1974, "Algorithm AS 70: The Percentage Points of the Normal Distribution," *Applied Statistics*, Vol.23, No.1, pp.96 – 97.

Odell, P.L. and Feiveson, A.H., 1966, "A Numerical Procedure to Generate a Simple Covariance Matrix," *Journal of the American Statistical Association*, Vol.61, No.313, pp.199 – 203.

Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P., 1992a, *Numerical Recipes in C: The Art of Scientific Computing* (Second Edition), Cambridge University Press.

Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P., 1992b, *Numerical Recipes in Fortran: The Art of Scientific Computing* (Second Edition), Cambridge University Press.

Ripley, B.D., 1987, *Stochastic Simulation*, John Wiley & Sons.

Robert, C.P. and Casella, G., 1999, *Monte Carlo Statistical Methods*, Springer-Verlag.

Ross, S.M., 1997, *Simulation* (Second Edition), Academic Press.

Rubinstein, R.Y., 1981, *Simulation and the Monte Carlo Method*, John Wiley & Sons.

Rubinstein, R.Y. and Melamed, B., 1998, *Modern Simulation and Modeling*, John Wiley & Sons.

Schmeiser, B. and Kachitvichyanukul, V., 1990, "Noninverse Correlation Induction: Guidelines for Algorithm Development," *Journal of Computational and Applied Mathematics*, Vol.31, pp.173 – 180.

Shibata, Y., 1981, *Normal Distribution* (in Japanese), Tokyo University Press.

Smith, W.B and Hocking, R.R., 1972, "Algorithm AS53: Wishart Variate Generator," *Applied Statistics*, Vol.21, No.3, pp.341 – 345.

Stadlober, E., 1990, "The Ratio of Uniforms Approach for Generating Discrete Random Variates," *Journal of Computational and Applied Mathematics*, Vol.31, pp.181 – 189.

Takeuchi, K., 1989, *Dictionary of Statistics* (in Japanese), Toyo-Keizai.

Thompson, J.R., 2000, *Simulation: A Modeler's Approach*, Jhon Wiley & Sons.

Wichmann, B.A. and Hill, I.D., 1982, "Algorithm AS183: An Efficient and Portable Pseudo-random Number Generator," *Applied Statistics*, Vol.31, No.2, pp.188 – 190.

Wichmann, B.A. and Hill, I.D., 1984, "Correction of Algorithm AS183: An Efficient and Portable Pseudo-random Number Generator," *Applied Statistics*, Vol.33, No.2, p.123.

Zellner, A., 1971, *An Introduction to Bayesian Inference in Econometrics*, John Wiley & Sons.

Chapter 3

Random Number Generation II

In Chapter 2, we have seen various random number generators based on the uniform random draws between zero and one. In this chapter, we consider the random number generation methods in the case where it is not easy to generate a random draw from the target density function. Note that the density function from which we want to generate random draws is called the **target density function**. We discuss the composition method in Section 3.1, the rejection sampling method in Section 3.2, the importance resampling method in Section 3.3, the Metropolis-Hastings algorithm in Section 3.4 and the ratio-of-uniforms method in Section 3.5. All these methods indicate the random number generation methods when it is intractable to generate random draws from the target density. In Section 3.6, as a multivariate case we introduce the Gibbs sampler, which is the random number generator which yields unconditional random variables from two conditional distributions. In Section 3.7, the random number generators discussed in Sections 3.1 – 3.6 are compared through some Monte Carlo studies.

3.1 Composition Method

As for one of the random number generation methods which enable us to generate random draws from any distribution, first we discuss the **composition method** in this section.

Suppose that $f(x)$ is represented by the following mixture distribution:

$$f(x) = \sum_{i=1}^{m} p_i f_i(x), \tag{3.1}$$

where $f_i(x)$, $i = 1, 2, \cdots, m$, are density functions or probability functions of random variables. We have $\sum_{i=1}^{m} p_i = 1$. The random draws of X are obtained as follows: (i) pick up i with probability p_i and (ii) generate random draws of x from $f_i(x)$.

In the case where it is not easy to generate random draws from $f(x)$, we may approximate $f(x)$ as a weighted sum of $f_1(x)$, $f_2(x)$, $\cdots$, $f_m(x)$, where $f_i(x)$ can take

any distribution function. In the next section, a uniform distribution is taken for $f_i(x)$, $i = 1, 2, \cdots, m$.

3.1.1 Composition of Uniform Distributions

Let $x_{(i)}$, $i = 0, 1, 2, \cdots, m$, be the nodes which are fixed and appropriately chosen. The target density $f(x)$ is approximated as the weighted sum of uniform distributions $f_i(x)$, which is given by:

$$f_i(x) \approx \begin{cases} \dfrac{1}{x_{(i)} - x_{(i-1)}}, & \text{if } x_{(i-1)} < x < x_{(i)}, \\ 0, & \text{otherwise.} \end{cases}$$

From the above approximation, the random draws are generated from $f(x)$ in the following way: (i) pick up i with probability p_i and (ii) generate u from $U(x_{(i-1)}, x_{(i)})$, where p_i is given by:

$$p_i = \int_{x_{(i-1)}}^{x_{(i)}} f(t)\, \mathrm{d}t.$$

u is taken as a random draw of X. The integration above may be evaluated by the trapezoid rule, i.e.,

$$p_i = \int_{x_{(i-1)}}^{x_{(i)}} f(t)\, \mathrm{d}t \approx \frac{1}{2}(f(x_{(i)}) + f(x_{(i-1)}))(x_{(i)} - x_{(i-1)}).$$

Any evaluation method can be taken for the integration, although the approximation method based on the trapezoid rule is shown above. Using p_i and $f_i(x)$, the target density $f(x)$ is represented as follows:

$$f(x) \approx \sum_{i=1}^{m} p_i f_i(x) = \sum_{i=1}^{m} p_i \frac{1}{x_{(i)} - x_{(i-1)}} I(x_{(i-1)}, x_{(i)}),$$

where $I(a, b)$ implies that $I(a, b) = 1$ when $a < x < b$ and $I(a, b) = 0$ otherwise. Clearly, as m increases, the approximation above shows a good performance.

Here, we take $f_i(x)$ as the uniform distribution between $x_{(i-1)}$ and $x_{(i)}$, although $f_i(x)$ can take any distribution. Now, using the composition method we consider an example of generating the standard normal random draws.

3.1.2 Normal Distribution: $N(0, 1)$

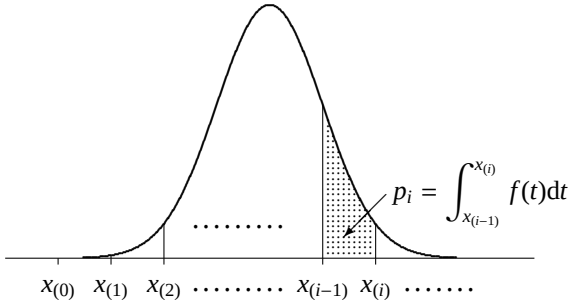
We generate the standard normal random draws using the composition method. As shown in Section 3.1.1, the standard normal distribution $N(0, 1)$ is approximated as the weighted sum of the uniform distributions. When $X \sim N(0, 1)$, we divide the standard normal density function into m regions. Using the trapezoid rule, the probability

which we are in ith region, i.e., p_i, is approximately given by:

$$p_i = \int_{x_{(i-1)}}^{x_{(i)}} f(t)\,\mathrm{d}t = \frac{1}{\sqrt{2\pi}} \int_{x_{(i-1)}}^{x_{(i)}} \exp(-\frac{1}{2}t^2)\,\mathrm{d}t$$
$$\approx \frac{1}{\sqrt{2\pi}} \frac{1}{2}\Big(\exp(-\frac{1}{2}x_{(i)}^2) + \exp(-\frac{1}{2}x_{(i-1)}^2)\Big)(x_{(i)} - x_{(i-1)}),$$

where $x_{(i)} - x_{(i-1)}$ may be set to be equal for all $i = 1, 2, \cdots, m$. p_i corresponds to the dotted area in Figure 3.1.

Figure 3.1: Approximation of Standard Normal Density



In practice, a domain of x is truncated, because it is divided into the m regions within the interval $(x_{(0)}, x_{(m)})$, where the regions less than x_0 and greater than $x_{(m)}$ are ignored. $\sum_{i=1}^{m} p_i$ is not necessarily equal to one. Therefore, we need to modify p_i as follows:

$$p_i \approx \frac{\int_{x_{(i-1)}}^{x_{(i)}} f(t)\,\mathrm{d}t}{\sum_{i=1}^{m} \int_{x_{(i-1)}}^{x_{(i)}} f(t)\,\mathrm{d}t} \approx \frac{(\exp(-\frac{1}{2}x_{(i)}^2) + \exp(-\frac{1}{2}x_{(i-1)}^2))(x_{(i)} - x_{(i-1)})}{\sum_{i=1}^{m}(\exp(-\frac{1}{2}x_{(i)}^2) + \exp(-\frac{1}{2}x_{(i-1)}^2))(x_{(i)} - x_{(i-1)})}.$$

Thus, under the modification shown above, clearly we have $\sum_{i=1}^{m} p_i = 1$. The source code is given by `snrnd5(ix,iy,x_L,x_U,m,rn)`, where `x_L` and `x_U` correspond to $x_{(0)}$ and $x_{(m)}$, respectively, and `m` denotes the number of regions.

`snrnd5(ix,iy,x_L,x_U,m,rn)`

```
1:       subroutine snrnd5(ix,iy,x_L,x_U,m,rn)
2:       dimension x(0:1001),prob(0:1001)
3: c
4: c  Use "snrnd5(ix,iy,x_L,x_U,m,rn)"
5: c  together with "urnd(ix,iy,rn)".
6: c
7: c  Input:
8: c    x_L: Lower bound of the Interval
9: c    x_U: Upper bound of the Interval
10: c    m:  The Number of Regions
11: c       (less than or equal to 1000)
```

```
12: c    fcn(x):  Target Density Function
13: c    ix, iy:  Seeds
14: c  Output:
15: c    rn: Standard Normal Random Draw
16: c
17:       x(0)=x_L
18:       x(m)=x_U
19:       prob(0)=0.0
20:       f0=fcn( x(0) )
21:       width=( x(m)-x(0) )/m
22:         do 1 i=1,m
23:       x(i)=x(0)+width*i
24:       f1=fcn( x(i) )
25:       prob(i)=prob(i-1)+0.5*(f1+f0)*width
26:     1 f0=f1
27:         do 2 i=1,m
28:     2 prob(i)=prob(i)/prob(m)
29:       call urnd(ix,iy,rn1)
30:         do 3 j=1,m
31:          if(prob(j-1).le.rn1.and.rn1.lt.prob(j)) then
32:       i=j
33:       go to 4
34:          endif
35:     3   continue
36:     4 call urnd(ix,iy,rn1)
37:       rn=x(i-1)+rn1*( x(i)-x(i-1) )
38:       return
39:       end
40: c ---------------------------------------
41:       function fcn(x)
42:         fcn=exp(-0.5*x*x)
43:       return
44:       end
```

snrnd5(ix,iy,x_L,x_U,m,rn) should be used together with urnd(ix,iy,rn) on p.83. prob(i) in snrnd5 indicates $\sum_{j=1}^{i} p_j$. Lines 27 and 28 represent the modification on p_i, as shown above. Moreover, $x_{(i)} - x_{(i-1)}$ is set to be equal for all $i = 1, 2, \cdots, m$, which corresponds to width in Line 21. In Lines 30 – 35, the region is randomly chosen. In Lines 36 and 37, a uniform random draw between $x_{(i-1)}$ and $x_{(i)}$ is generated within the randomly chosen region i. fcn(x) in Lines 40 – 44 corresponds to the kernel of the target distribution. Here, we take the standard normal distribution, but we can take any continuous type of distribution by changing Line 42.

Clearly, when the distribution lies on the interval $(x_{(0)}, x_{(m)})$ with high probability, the approximation above shows a good performance as m goes to infinity. However, we have a problem, which is as follows. When we want to generate a lot of random draws, the source code snrnd5(ix,iy,x_L,x_U,m,rn) is inefficient in terms of computational time. In order to improve this problem, it is important to divide snrnd5 into two parts, i.e., Lines 17 – 28 and Lines 29 – 37 should be separated as shown in weight(x_L,x_U,m,x,prob) and snrnd5_2(ix,iy,x,prob,m,rn).

weight(x_L,x_U,m,x,prob)

```
      subroutine weight(x_L,x_U,m,x,prob)
      dimension x(0:1001),prob(0:1001)
c
c  Input:
c    x_L: Lower bound of the Interval
c    x_U: Upper bound of the Interval
c    m:  The Number of Regions
c       (less than or equal to 1000)
c  Output:
c    x(i):    Nodes
c    prob(i): Prob( X<x(i) )
c
       x(0)=x_L
       x(m)=x_U
       prob(0)=0.0
       f0=fcn( x(0) )
       width=( x(m)-x(0) )/m
         do 1 i=1,m
       x(i)=x(0)+width*i
       f1=fcn( x(i) )
       prob(i)=prob(i-1)+0.5*(f1+f0)*width
     1 f0=f1
         do 2 i=1,m
     2 prob(i)=prob(i)/prob(m)
       return
       end
```

snrnd5_2(ix,iy,x,prob,m,rn)

```
      subroutine snrnd5_2(ix,iy,x,prob,m,rn)
      dimension x(0:1001),prob(0:1001)
c
c  Use "snrnd5_2(ix,iy,x,prob,m,rn)"
c  together with "urnd(ix,iy,rn)".
c  Also use weight(x_L,x_U,m,x,prob)
c    to obtain x(i) and prob(i).
c
c  Input:
c    x(i):    Nodes
c    prob(i): Prob( X\le x(i) )
c    m:  The Number of Regions
c       (less than or equal to 1000)
c    ix, iy:  Seeds
c  Output:
c    rn: Standard Normal Random Draw
c
       call urnd(ix,iy,rn1)
         do 1 j=1,m
          if(prob(j-1).le.rn1.and.rn1.lt.prob(j)) then
       i=j
       go to 2
          endif
     1   continue
```

```
25:      2 call urnd(ix,iy,rn1)
26:        rn=x(i-1)+rn1*( x(i)-x(i-1) )
27:        return
28:        end
```

Using `weight` and `snrnd5_2`, an example of the main program is shown below. In the main program, the standard normal distribution is approximated by the weighted sum of `m=100` uniform distributions and `n=1000` standard normal random draws are generated. In Lines 3 – 6 of `Main Program for snrnd5_2`, the first 1000 uniform random draws are excluded from consideration, because the initial values `ix=1` and `iy=1` influence `rn` in the first some iterations and as a result the generated random draws are unstable.

Main Program for snrnd5_2

```
1:         dimension x(0:1001),prob(0:1001)
2:  c
3:         ix=1
4:         iy=1
5:           do 99 i=1,1000
6:     99 call urnd(ix,iy,rn)
7:  c
8:         n=1000
9:         x_L=-5.0
10:        x_U= 5.0
11:        m  =100
12:        call weight(x_L,x_U,m,x,prob)
13:          do 1 i=1,n
14:      1 call snrnd5_2(ix,iy,x,prob,m,rn)
15: c
16:        end
```

In Line 12, `prob(i)` is obtained for all `i=1,2,...,m`. After computing all the probability weights `prob(i)`, `i=1,2,...,m`, `n` random draws are generated in Lines 13 and 14.

By separating `weight` and `snrnd5_2` from `snrnd5`, computational burden is extremely reduced. In Section 3.7.1, to examine precision of the random draws and computational time, 10^7 standard normal random draws are generated within the interval $(x_{(0)}, x_{(m)}) = (-5, 5)$. There, we utilize `weight` and `snrnd5_2`, rather than `snrnd5`.

3.1.3 Binomial Distribution: $B(n, p)$

In Section 2.4.5, we have discussed how to generate binomial random draws. In this section, we consider the random number generator based on the composition method.

Again, we denote the binomial probability function as follows:

$$f(x) = P(X = x) = \frac{n!}{x!(n-x)!} p^x (1-p)^{n-x},$$

for $x = 0, 1, 2, \cdots, n$. The probability function above implies that we choose x with probability $P(X = x)$. Therefore, we need to compute $P(X = 1)$, $P(X = 2)$, $\cdots$, $P(X = n)$. Then, the binomial probability function is represented by:

$$f(x) = \sum_{x=0}^{n} p_x I(X = x),$$

where $p_x = P(X = x)$. $I(X = x)$ indicates the indicator function such that $I(X = x) = 1$ when X takes x and $I(X = x) = 0$ otherwise. The indicator function $I(X = x)$ is regarded as the probability function that X takes x with probability one. Thus, the binomial probability function is represented as the weighted sum of the n probability functions.

The Fortran program for the binomial random draws based on the composition method is equivalent to `birnd2(ix,iy,n,p,rn)`. In Section 3.7.3, we compare `birnd`, `birnd2` and `birnd3` with respect to computational CPU time and precision of the random draws.

3.1.4 Bimodal Distribution with Two Normal Densities

Consider the case $m = 2$ in equation (3.1), i.e.,

$$f(x) = p_1 f_1(x) + p_1 f_2(x),$$

where $p_1 + p_2 = 1$. $f_i(x)$ is assumed to be a normal distribution with mean μ_i and variance σ_i^2 for $i = 1, 2$.

Then, a random draw from the bimodal distribution is generated as: (i) pick up i with probability p_i and (ii) generate a random draw from $f_i(x)$. Accordingly, the Fortran 77 source code for the bimodal random number generator is given by `bimodal(ix,iy,p1,a1,v1,a2,v2,rn)`, where `p1`, `a1`, `v1`, `a2` and `v2` represent p_1, μ_1, σ_1^2, μ_2 and σ_2^2, respectively. In Line 20 of the source code, i is randomly chosen. A normal random draw from the first distribution is generated in Line 21 and a normal random draw from the second distribution is in Line 23.

`bimodal(ix,iy,p1,a1,v1,a2,v2,rn)`

```
1:       subroutine bimodal(ix,iy,p1,a1,v1,a2,v2,rn)
2: c
3: c  Use "bimodal(ix,iy,p1,a1,v1,a2,v2,rn)"
4: c  together with "snrnd(ix,iy,rn)"
5: c            and  "urnd(ix,iy,rn)".
```

```
c
c  Input:
c    p1: The probability which we have
c        the 1st distribution
c    a1: mean in the 1st distribution
c    v1: variance in the 1st distribution
c    a2: mean in the 2nd distribution
c    v2: variance in the 2nd distribution
c    ix, iy:  Seeds
c  Output:
c    rn: Bimodal Random Draw
c
       call urnd(ix,iy,ru)
       call snrnd(ix,iy,rn1)
         if(ru.le.p1) then
       rn=a1+sqrt(v1)*rn1
         else
       rn=a2+sqrt(v2)*rn1
         endif
       return
       end
```

Thus, `bimodal(ix,iy,p1,ave1,var1,ave2,var2,rn)` should be used together with `snrnd(ix,iy,rn)` on p.85 and `urnd(ix,iy,rn)` on p.83.

In this section, we consider the bimodal distribution based on the two normal densities. We can easily extend the above source code to the multi-modal cases. It is also possible to take a continuous type of distribution for $i = 1$ and a discrete type for $i = 2$.

3.2 Rejection Sampling

In Section 3.1, we have shown a general solution to generate random draws from any distribution, where the composition method is utilized. As the number of nodes, m, increases, precision of the obtained random draws is certainly improved but computational burden also increases. To reduce the computational disadvantage, we introduce the three sampling methods in Sections 3.2 – 3.4.

In Sections 3.2 – 3.4, we take the same setup as in Section 3.1. That is, we want to generate random draws from $f(x)$, called the **target density**, but we consider the case where it is hard to sample from $f(x)$.

Now, suppose that it is easy to generate a random draw from another density $f_*(x)$, called the **sampling density**. In this case, random draws of X from $f(x)$ are generated by utilizing the random draws sampled from $f_*(x)$. Let x be the the random draw of X generated from $f(x)$. Suppose that $q(x)$ is equal to the ratio of the target density and the sampling density, i.e.,

$$q(x) = \frac{f(x)}{f_*(x)}. \tag{3.2}$$

Then, the target density is rewritten as:

$$f(x) = q(x) f_*(x).$$

Based on $q(x)$, the acceptance probability is obtained. Depending on the structure of the acceptance probability, we have three kinds of sampling techniques, i.e., **rejection sampling** in this section, **importance resampling** in Section 3.3 and the **Metropolis-Hastings algorithm** in Section 3.5. See Liu (1996) for a comparison of the three sampling methods. Thus, to generate random draws of x from $f(x)$, the functional form of $q(x)$ should be known and random draws have to be easily generated from $f_*(x)$.

In order for rejection sampling to work well, the following condition has to be satisfied:

$$q(x) = \frac{f(x)}{f_*(x)} < c,$$

where c is a fixed value. That is, $q(x)$ has an upper limit. As discussed below, $1/c$ is equivalent to the acceptance probability. If the acceptance probability is large, rejection sampling computationally takes a lot of time. Under the condition $q(x) < c$ for all x, we may minimize c. That is, since we have $q(x) < \sup_x q(x) \leq c$, we may take the supremum of $q(x)$ for c. Thus, in order for rejection sampling to work efficiently, c should be the supremum of $q(x)$ with respect to x, i.e., $c = \sup_x q(x)$.

Let x^* be the random draw generated from $f_*(x)$, which is a candidate of the random draw generated from $f(x)$. Define $\omega(x)$ as:

$$\omega(x) = \frac{q(x)}{\sup_z q(z)} = \frac{q(x)}{c},$$

which is called the **acceptance probability**. Note that we have $0 \leq \omega(x) \leq 1$ when $\sup_z q(z) = c < \infty$. The supremum $\sup_z q(z) = c$ has to be finite. This condition is sometimes too restrictive, which is a crucial problem in rejection sampling.

A random draw of X is generated from $f(x)$ in the following way:

(i) Generate x^* from $f_*(x)$ and compute $\omega(x^*)$.

(ii) Set $x = x^*$ with probability $\omega(x^*)$ and go back to (i) otherwise. In other words, generating u from a uniform distribution between zero and one, take $x = x^*$ if $u \leq \omega(x^*)$ and go back to (i) otherwise.

The above random number generation procedure can be justified as follows. Let U be the uniform random variable between zero and one, X be the random variable generated from the target density $f(x)$, X^* be the random variable generated from the sampling density $f_*(x)$, and x^* be the realization (i.e., the random draw) generated from the sampling density $f_*(x)$. Consider the probability $P(X \leq x | U \leq \omega(x^*))$, which should be the cumulative distribution of X, $F(x)$, from Step (ii). The probability $P(X \leq x | U \leq \omega(x^*))$ is rewritten as follows:

$$P(X \leq x | U \leq \omega(x^*)) = \frac{P(X \leq x, U \leq \omega(x^*))}{P(U \leq \omega(x^*))},$$

where the numerator is represented as:

$$\begin{aligned} P(X \le x, U \le \omega(x^*)) &= \int_{-\infty}^{x} \int_{0}^{\omega(t)} f_{u,*}(u,t)\,\mathrm{d}u\,\mathrm{d}t = \int_{-\infty}^{x} \int_{0}^{\omega(t)} f_u(u) f_*(t)\,\mathrm{d}u\,\mathrm{d}t \\ &= \int_{-\infty}^{x} \Big(\int_{0}^{\omega(t)} f_u(u)\,\mathrm{d}u \Big) f_*(t)\,\mathrm{d}t = \int_{-\infty}^{x} \Big(\int_{0}^{\omega(t)} \mathrm{d}u \Big) f_*(t)\,\mathrm{d}t \\ &= \int_{-\infty}^{x} \Big[u \Big]_{0}^{\omega(t)} f_*(t)\,\mathrm{d}t = \int_{-\infty}^{x} \omega(t) f_*(t)\,\mathrm{d}t = \int_{-\infty}^{x} \frac{q(t)}{c} f_*(t)\,\mathrm{d}t = \frac{F(x)}{c}, \end{aligned}$$

and the denominator is given by:

$$P(U \le \omega(x^*)) = P(X \le \infty, U \le \omega(x^*)) = \frac{F(\infty)}{c} = \frac{1}{c}.$$

In the numerator, $f_{u,*}(u,x)$ denotes the joint density of random variables U and X^*. Because the random draws of U and X^* are independently generated in Steps (i) and (ii) we have $f_{u,*}(u,x) = f_u(u) f_*(x)$, where $f_u(u)$ and $f_*(x)$ denote the marginal density of U and that of X^*. The density function of U is given by $f_u(u) = 1$, because the distribution of U is assumed to be uniform between zero and one. Thus, the first four equalities are derived. Furthermore, in the seventh equality of the numerator, since we have:

$$\omega(x) = \frac{q(x)}{c} = \frac{f(x)}{c f_*(x)},$$

$\omega(x) f_*(x) = f(x)/c$ is obtained. Finally, substituting the numerator and denominator shown above, we have the following equality:

$$P(X \le x | U \le \omega(x^*)) = F(x).$$

Thus, the rejection sampling method given by Steps (i) and (ii) is justified.

The rejection sampling method is the most efficient sampling method in the sense of precision of the random draws, because using rejection sampling we can generate mutually independently distributed random draws. However, for rejection sampling we need to obtain the c which is greater than or equal to the supremum of $q(x)$. If the supremum is infinite, i.e., if c is infinite, $\omega(x)$ is zero and accordingly the candidate x^* is never accepted in Steps (i) and (ii).

Moreover, as for another remark, note as follows. Let N_R be the average number of the rejected random draws. We need $(1+N_R)$ random draws in average to generate one random number from $f(x)$. In other words, the acceptance rate is given by $1/(1+N_R)$ in average, which is equal to $1/c$ in average because of $P(U \le \omega(x^*)) = 1/c$. Therefore, to obtain one random draw from $f(x)$, we have to generate $(1+N_R)$ random draws from $f_*(x)$ in average. See, for example, Boswell, Gore, Patil and Taillie (1993), O'Hagan (1994) and Geweke (1996) for rejection sampling.

To examine the condition that $\omega(x)$ is greater than zero, i.e., the condition that the supremum of $q(x)$ exists, consider the case where $f(x)$ and $f_*(x)$ are distributed as

$N(\mu, \sigma^2)$ and $N(\mu_*, \sigma_*^2)$, respectively. $q(x)$ is given by:

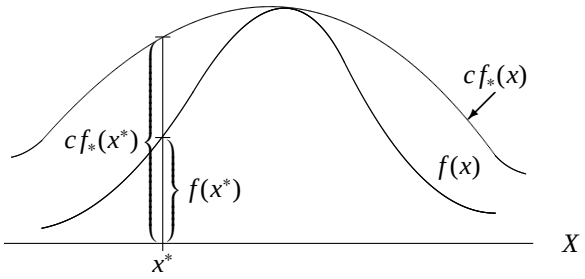
$$\begin{aligned} q(x) = \frac{f(x)}{f_*(x)} &= \frac{(2\pi\sigma^2)^{-1/2}\exp\Big(-\frac{1}{2\sigma^2}(x-\mu)^2\Big)}{(2\pi\sigma_*^2)^{-1/2}\exp\Big(-\frac{1}{2\sigma_*^2}(x-\mu_*)^2\Big)} \\ &= \frac{\sigma_*}{\sigma}\exp\Big(-\frac{1}{2\sigma^2}(x-\mu)^2 + \frac{1}{2\sigma_*^2}(x-\mu_*)^2\Big) \\ &= \frac{\sigma_*}{\sigma}\exp\left(-\frac{1}{2}\frac{\sigma_*^2-\sigma^2}{\sigma^2\sigma_*^2}\Big(x - \frac{\mu\sigma_*^2 - \mu_*\sigma^2}{\sigma_*^2-\sigma^2}\Big)^2 + \frac{1}{2}\frac{(\mu-\mu_*)^2}{\sigma_*^2-\sigma^2}\right). \end{aligned}$$

If $\sigma_*^2 < \sigma^2$, $q(x)$ goes to infinity as x is large. In the case of $\sigma_*^2 > \sigma^2$, the supremum of $q(x)$ exists, which condition implies that $f_*(x)$ should be more broadly distributed than $f(x)$. In this case, the supremum is obtained as:

$$c = \sup_x q(x) = \frac{\sigma_*}{\sigma}\exp\Big(\frac{1}{2}\frac{(\mu-\mu_*)^2}{\sigma_*^2-\sigma^2}\Big).$$

When $\sigma^2 = \sigma_*^2$ and $\mu = \mu_*$, we have $q(x) = 1$, which implies $\omega(x) = 1$. That is, a random draw from the sampling density $f_*(x)$ is always accepted as a random draw from the target density $f(x)$, where $f(x)$ is equivalent to $f_*(x)$ for all x. If $\sigma^2 = \sigma_*^2$ and $\mu \neq \mu_*$, the supremum of $q(x)$ does not exists. Accordingly, the rejection sampling method does not work in this case.

Figure 3.2: Rejection Sampling



From the definition of $\omega(x)$, we have the inequality $f(x) \leq cf_*(x)$. $cf_*(x)$ and $f(x)$ are displayed in Figure 3.2. The ratio of $f(x^*)$ and $cf_*(x^*)$ corresponds to the acceptance probability at x^*, i.e., $\omega(x^*)$. Thus, for rejection sampling, $cf_*(x)$ has to be greater than or equal to $f(x)$ for all x, which implies that the sampling density $f_*(x)$ needs to be more widely distributed than the target density $f(x)$.

Finally, note that the above discussion holds without any modification even though $f(x)$ is a kernel of the target density, i.e., even though $f(x)$ is proportional to the target density, because the constant term is canceled out between the numerator and denominator (remember that $\omega(x) = q(x)/\sup_z q(z)$).

3.2.1 Normal Distribution: $N(0, 1)$

First, denote the half-normal distribution by:

$$f(x) = \begin{cases} \dfrac{2}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}, & \text{for } 0 \le x < \infty, \\ 0, & \text{otherwise.} \end{cases}$$

The half-normal distribution above corresponds to the positive part of the standard normal probability density function. Using rejection sampling, we consider generating standard normal random draws based on the half-normal distribution. We take the sampling density as the exponential distribution:

$$f_*(x) = \begin{cases} \lambda e^{-\lambda x}, & \text{for } 0 \le x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

where $\lambda > 0$. Since $q(x)$ is defined as $q(x) = f(x)/f_*(x)$, the supremum of $q(x)$ is given by:

$$c = \sup_x q(x) = \frac{2}{\lambda\sqrt{2\pi}} e^{\frac{1}{2}\lambda^2}.$$

which depends on parameter λ. Remember that $P(U \le \omega(x^*)) = 1/c$ corresponds to the acceptance probability. Since we need to increase the acceptance probability to reduce computational time, we want to obtain the λ which minimizes $\sup_x q(x)$ with respect to λ. Solving the minimization problem, $\lambda = 1$ is obtained. Substituting $\lambda = 1$, the acceptance probability $\omega(x)$ is derived as:

$$\omega(x) = e^{-\frac{1}{2}(x-1)^2},$$

for $0 < x < \infty$.

Remember that $-\log U$ has an exponential distribution with $\lambda = 1$ when $U \sim U(0, 1)$. Therefore, the algorithm is represented as follows.

(i) Generate two independent uniform random draws u_1 and u_2 between zero and one.

(ii) Compute $x^* = -\log u_2$, which indicates the exponential random draw generated from the target density $f_*(x)$.

(iii) Set $x = x^*$ if $u_1 \le \exp(-\frac{1}{2}(x^* - 1)^2)$, i.e., $-2\log(u_1) \ge (x^* - 1)^2$, and return to (i) otherwise.

x in Step (iii) yields a random draw from the half-normal distribution. To generate a standard normal random draw utilizing the half-normal random draw above, we may put the positive or negative sign randomly with x. Therefore, the following Step (iv) is additionally put.

(iv) Generate a uniform random draw u_3 between zero and one, and set $z = x$ if $u_3 \le 1/2$ and $z = -x$ otherwise.

z gives us a standard normal random draw. Note that the number of iteration in Step (iii) is given by $c = \sqrt{2e/\pi} \approx 1.3155$ in average, or equivalently, the acceptance probability in Step (iii) is $1/c \approx 0.7602$. The source code for this standard normal random number generator is shown in `snrnd6(ix,iy,rn)`.

snrnd6(ix,iy,rn)

```
      subroutine snrnd6(ix,iy,rn)
c
c  Use "snrnd6(ix,iy,rn)"
c  together with "urnd(ix,iy,rn)".
c
c  Input:
c    ix, iy:   Seeds
c  Output:
c    rn: Normal Random Draw N(0,1)
c
     1 call urnd(ix,iy,rn1)
       call urnd(ix,iy,rn2)
       y=-log(rn2)
       if( -2.*log(rn1).lt.(y-1.)**2 ) go to 1
       call urnd(ix,iy,rn3)
         if(rn3.le.0.5) then
       rn= y
         else
       rn=-y
         endif
       return
       end
```

Note that `snrnd6(ix,iy,rn)` should be used together with `urnd(ix,iy,rn)` on p.83. Thus, utilizing rejection sampling, we have the standard normal random number generator, which is based on the half-normal distribution.

3.2.2 Gamma Distribution: $G(\alpha, 1)$ for $0 < \alpha \leq 1$ and $1 < \alpha$

In this section, utilizing rejection sampling we show an example of generating random draws from the gamma distribution with parameters α and $\beta = 1$, i.e., $G(\alpha, 1)$. When $X \sim G(\alpha, 1)$, the density function of X is given by:

$$f(x) = \begin{cases} \dfrac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x}, & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise.} \end{cases}$$

Ahrens and Dieter (1974) consider the case of $0 < \alpha \leq 1$, which is discussed in this section. The case of $\alpha > 1$ will be discussed later in Section 3.5.2. Using the rejection sampling, the composition method and the inverse transform method, we consider

generating random draws from $G(\alpha, 1)$ for $0 < \alpha \leq 1$. The sampling density is taken as:

$$f_*(x) = \frac{e}{\alpha + e}\alpha x^{\alpha-1} I_1(x) + \frac{\alpha}{\alpha + e} e^{-x+1} I_2(x),$$

where both $I_1(x)$ and $I_2(x)$ denote the indicator functions defined as:

$$I_1(x) = \begin{cases} 1, & \text{if } 0 < x \leq 1, \\ 0, & \text{otherwise,} \end{cases} \qquad I_2(x) = \begin{cases} 1, & \text{if } 1 < x, \\ 0, & \text{otherwise.} \end{cases}$$

Random number generation from the sampling density above utilizes the composition method and the inverse transform method, which are shown in Sections 2.3 and 3.1. The cumulative distribution related to $f_*(x)$ is given by:

$$F_*(x) = \begin{cases} \dfrac{e}{\alpha + e} x^{\alpha}, & \text{if } 0 < x \leq 1, \\ \dfrac{e}{\alpha + e} + \dfrac{\alpha}{\alpha + e}(1 - e^{-x+1}), & \text{if } x > 1. \end{cases}$$

Note that $0 < \alpha \leq 1$ is required because the sampling density for $0 < x \leq 1$ has to satisfy the property shown on p.5 (i.e., the integration is equal to one).

The acceptance probability $\omega(x) = q(x)/\sup_z q(z)$ for $q(x) = f(x)/f_*(x)$ is given by:

$$\omega(x) = e^{-x} I_1(x) + x^{\alpha-1} I_2(x).$$

Moreover, the mean number of trials until success, i.e., $c = \sup_z q(z)$ is represented as:

$$c = \frac{\alpha + e}{\alpha e \Gamma(\alpha)},$$

which depends on α and is not greater than 1.39. Note that $q(x)$ takes a maximum value at $x = 1$. The random number generation procedure is given by:

(i) Generate a uniform random draw u_1 from $U(0, 1)$, and set $x^* = ((\alpha/e + 1)u_1)^{1/\alpha}$ if $u_1 \leq e/(\alpha + e)$ and $x^* = -\log((1/e + 1/\alpha)(1 - u_1))$ if $u_1 > e/(\alpha + e)$.

(ii) Obtain $\omega(x^*) = e^{-x^*}$ if $u_1 \leq e/(\alpha + e)$ and $\omega(x^*) = x^{*\alpha-1}$ if $u_1 > e/(\alpha + e)$.

(iii) Generate a uniform random draw u_2 from $U(0, 1)$, and set $x = x^*$ if $u_2 \leq \omega(x^*)$ and return to (i) otherwise.

In Step (i) a random draw x^* from $f_*(x)$ can be generated by the inverse transform method discussed in Section 2.3 and the composition method in Section 3.1.

gammarnd2(ix,iy,alpha,rn)

```
1:       subroutine gammarnd2(ix,iy,alpha,rn)
2: c
3: c  Use "gammarnd2(ix,iy,alpha,rn)"
4: c  together with "urnd(ix,iy,rn)".
```

```
5: c
6: c  Input:
7: c    ix, iy: Seeds
8: c    alpha:  Shape Parameter (0<alpha \le 1)
9: c  Output:
10: c    rn: Gamma Random Draw
11: c        with Parameters alpha and beta=1
12: c
13:       e=2.71828182845905
14:      1 call urnd(ix,iy,rn0)
15:       call urnd(ix,iy,rn1)
16:             if( rn0.le.e/(alpha+e) ) then
17:       rn=( (alpha+e)*rn0/e )**(1./alpha)
18:       if( rn1.gt.e**(-rn) ) go to 1
19:             else
20:       rn=-log((alpha+e)*(1.-rn0)/(alpha*e))
21:       if( rn1.gt.rn**(alpha-1.) ) go to 1
22:             endif
23:       return
24:       end
```

Note that `gammarnd2(ix,iy,alpha,rn)` should be used with `urnd(ix,iy,rn)` on p.83.

In `gammarnd2(ix,iy,alpha,rn)`, the case of $0 < \alpha \leq 1$ has been shown. Now, using rejection sampling, the case of $\alpha > 1$ is discussed in Cheng (1977, 1998). The sampling density is chosen as the following cumulative distribution:

$$F_*(x) = \begin{cases} \dfrac{x^\lambda}{\delta + x^\lambda}, & \text{for } x > 0, \\ 0, & \text{otherwise,} \end{cases}$$

which is sometimes called the **log-logistic distribution**. Then, the probability density function, $f_*(x)$, is given by:

$$f_*(x) = \begin{cases} \dfrac{\lambda\delta x^{\lambda-1}}{(\alpha + x^\lambda)^2}, & \text{for } x > 0, \\ 0, & \text{otherwise.} \end{cases}$$

By the inverse transform method, the random draw from $f_*(x)$, denoted by x, is generated as follows:

$$x = \left(\frac{\delta u}{1-u}\right)^{1/\lambda},$$

where u denotes the uniform random draw generated from $U(0,1)$. For the two parameters, $\lambda = \sqrt{2\alpha - 1}$ and $\delta = \alpha^\lambda$ are chosen, taking into account minimizing $c = \sup_x q(x) = \sup_x f(x)/f_*(x)$ with respect to δ and λ (note that λ and δ are approximately taken, since it is not possible to obtain the explicit solution of δ and λ). Then, the number of rejections in average is given by:

$$c = \frac{4\alpha^\alpha e^{-\alpha}}{\Gamma(\alpha)\sqrt{2\alpha - 1}},$$

which is computed as 1.47 when $\alpha = 1$, 1.25 when $\alpha = 2$, 1.17 when $\alpha = 5$, 1.15 when $\alpha = 10$ and 1.13 when $\alpha = \infty$. Thus, the average number of rejections is quite small for all α. The random number generation procedure is given by:

(i) Set $a = 1/\sqrt{2\alpha - 1}$, $b = \alpha - \log 4$ and $c = \alpha + \sqrt{2\alpha - 1}$.

(ii) Generate two uniform random draws u_1 and u_2 from $U(0, 1)$.

(iii) Set $y = a \log \dfrac{u_1}{1 - u_1}$, $x^* = \alpha e^y$, $z = u_1^2 u_2$ and $r = b + cy - x$.

(iv) Take $x = x^*$ if $r \geq \log z$ and return to (ii) otherwise.

To avoid evaluating the logarithm in Step (iv), we put Step (iii)' between Steps (iii) and (iv), which is as follows:

(iii)' Take $x = x^*$ if $r \geq 4.5z - d$ and go to (iv) otherwise.

d is defined as $d = 1 + \log 4.5$, which has to be computed in Step (i). Note that we have the relation: $\theta z - (1 + \log \theta) \geq \log z$ for all $z > 0$ and any given $\theta > 0$, because $\log z$ is a concave function of z. According to Cheng (1977), the choice of θ is not critical and the suggested value is $\theta = 4.5$, irrespective of α. The source code for Steps (i) – (iv) and (iii)' is given by `gammarnd3(ix,iy,alpha,rn)`.

gammarnd3(ix,iy,alpha,rn)

```
1:        subroutine gammarnd3(ix,iy,alpha,rn)
2:  c
3:  c  Use "gammarnd3(ix,iy,alpha,rn)"
4:  c  together with "urnd(ix,iy,rn)".
5:  c
6:  c  Input:
7:  c    ix, iy: Seeds
8:  c    alpha:  Shape Parameter (1<alpha)
9:  c  Output:
10: c    rn: Gamma Random Draw
11: c        with Parameters alpha and beta=1
12: c
13:       e=2.71828182845905
14:       a=1./sqrt(2.*alpha-1.)
15:       b=alpha-log(4.)
16:       c=alpha+sqrt(2.*alpha-1.)
17:       d=1.+log(4.5)
18:     1 call urnd(ix,iy,u1)
19:       call urnd(ix,iy,u2)
20:       y=a*log(u1/(1.-u1))
21:       rn=alpha*(e**y)
22:       z=u1*u1*u2
23:       r=b+c*y-rn
24:       if( r.ge.4.5*z-d ) go to 2
25:       if( r.lt.log(z)  ) go to 1
26:     2 return
27:       end
```

Note that `gammarnd3(ix,iy,alpha,rn)` requires `urnd(ix,iy,rn)`. Line 24 corresponds to Step (iii)', which gives us a fast acceptance. Taking into account a recent progress of a personal computer, we can erase Lines 17 and 24 from `gammarnd3`, because evaluating the `if(...)` sentences in Lines 24 and 25 sometimes takes more time than computing the logarithm in Line 25.

Thus, using both `gammarnd2` and `gammarnd3`, we have the gamma random number generator with parameters $\alpha > 0$ and $\beta = 1$.

3.3 Importance Resampling

The **importance resampling** method also utilizes the sampling density $f_*(x)$, where we should choose the sampling density from which it is easy to generate random draws. Let x_i^* be the ith random draw of x generated from $f_*(x)$. The acceptance probability is defined as:

$$\omega(x_i^*) = \frac{q(x_i^*)}{\sum_{j=1}^n q(x_j^*)},$$

where $q(\cdot)$ is represented as equation (3.2). To obtain a random draws from $f(x)$, we perform the following procedure:

(i) Generate x_j^* from the sampling density $f_*(x)$ for $j = 1, 2, \cdots, n$.

(ii) Compute $\omega(x_j^*)$ for all $j = 1, 2, \cdots, n$.

(iii) Generate a uniform random draw u between zero and one and take $x = x_j^*$ when $\Omega_{j-1} \le u < \Omega_j$, where $\Omega_j = \sum_{i=1}^j \omega(x_i^*)$ and $\Omega_0 \equiv 0$.

The x obtained in Step (iii) represents a random draw from the target density $f(x)$. In Step (ii), all the probability weights $\omega(x_j^*)$, $j = 1, 2, \cdots, n$, have to be computed for importance resampling. Thus, we need to generate n random draws from the sampling density $f_*(x)$ in advance. When we want to generate more random draws (say, N random draws), we may repeat Step (iii) N times.

In the importance resampling method, there are n realizations, i.e., x_1^*, x_2^*, $\cdots$, x_n^*, which are mutually independently generated from the sampling density $f_*(x)$. The cumulative distribution of $f(x)$ is approximated by the following empirical distribution:

$$\begin{aligned} P(X \le x) &= \int_{-\infty}^x f(t)\,\mathrm{d}t = \int_{-\infty}^x \frac{f(t)}{f_*(t)} f_*(t)\,\mathrm{d}t = \frac{\int_{-\infty}^x q(t) f_*(t)\,\mathrm{d}t}{\int_{-\infty}^\infty q(t) f_*(t)\,\mathrm{d}t} \\ &\approx \frac{(1/n)\sum_{i=1}^n q(x_i^*) I(x, x_i^*)}{(1/n)\sum_{j=1}^n q(x_j^*)} = \sum_{i=1}^n \omega(x_i^*) I(x, x_i^*), \end{aligned}$$

where $I(x, x_i^*)$ denotes the indicator function which satisfies $I(x, x_i^*) = 1$ when $x \ge x_i^*$ and $I(x, x_i^*) = 0$ otherwise. $P(X = x_i^*)$ is approximated as $\omega(x_i^*)$. See Smith and Gelfand (1992) and Bernardo and Smith (1994) for the importance resampling procedure.

As mentioned in Section 3.2, for rejection sampling, $f(x)$ may be a kernel of the target density, or equivalently, $f(x)$ may be proportional to the target density. Similarly, the same situation holds in the case of importance resampling. That is, $f(x)$ may be proportional to the target density for importance resampling, too.

To obtain a random draws from $f(x)$, importance resampling requires n random draws from the sampling density $f_*(x)$, but rejection sampling needs $(1 + N_R)$ random draws from the sampling density $f_*(x)$. For importance resampling, when we have n different random draws from the sampling density, we pick up one of them with the corresponding probability weight. The importance resampling procedure computationally takes a lot of time, because we have to compute all the probability weights Ω_j, $j = 1, 2, \cdots, n$, in advance even when we want only one random draw. When we want to generate N random draws, importance resampling requires n random draws from the sampling density $f_*(x)$, but rejection sampling needs $n(1+N_R)$ random draws from the sampling density $f_*(x)$. Thus, as N increases, importance resampling is relatively less computational than rejection sampling. Note that $N < n$ is recommended for the importance resampling method. In addition, when we have N random draws from the target density $f(x)$, some of the random draws take the exactly same values for importance resampling, while all the random draws take the different values for rejection sampling. Therefore, we can see that importance resampling is inferior to rejection sampling in the sense of precision of the random draws.

Using the importance resampling procedure, the three examples are shown in Sections 3.3.1 – 3.3.3, which are the standard normal random number generator in Section 3.3.1, the gamma random number generator in Section 3.3.2 and the beta random number generator in Section 3.3.3.

3.3.1 Normal Distribution: $N(0, 1)$

Again, we consider an example of generating standard normal random draws based on the half-normal distribution:

$$f(x) = \begin{cases} \dfrac{2}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}, & \text{for } 0 \le x < \infty, \\ 0, & \text{otherwise.} \end{cases}$$

We take the sampling density as the following exponential distribution:

$$f_*(x) = \begin{cases} e^{-x}, & \text{for } 0 \le x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

which is exactly the same sampling density as in Section 3.2.1. Given the random draws x_i^*, $i = 1, \cdots, n$, generated from the above exponential density $f_*(x)$, the acceptance probability $\omega(x_i^*)$ is given by:

$$\omega(x_i^*) = \frac{q(x_i^*)}{\sum_{j=1}^n q(x_j^*)} = \frac{f(x_i^*)/f_*(x_i^*)}{\sum_{j=1}^n f(x_j^*)/f_*(x_j^*)} = \frac{\exp(-\frac{1}{2}x_i^{*2} + x_i^*)}{\sum_{j=1}^n \exp(-\frac{1}{2}x_j^{*2} + x_j^*)}.$$

Therefore, a random draw from the half-normal distribution is generated as follows.

(i) Generate uniform random draws $u_1, u_2, \cdots, u_n$ from $U(0,1)$.

(ii) Obtain $x_i^* = -\log(u_i)$ for $i = 1, 2, \cdots, n$.

(iii) Compute $\omega(x_i^*)$ for $i = 1, 2, \cdots, n$.

(iv) Generate a uniform random draw v_1 from $U(0,1)$.

(v) Set $x = x_j^*$ when $\Omega_{j-1} \leq v_1 < \Omega_j$ for $\Omega_j = \sum_{i=1}^{j} \omega(x_i^*)$ and $\Omega_0 = 0$.

x is taken as a random draw generated from the half-normal distribution $f(x)$. In order to have a standard normal random draw, we additionally put the following step.

(vi) Generate a uniform random draw v_2 from $U(0,1)$, and set $z = x$ if $v_2 \leq 1/2$ and $z = -x$ otherwise.

z represents a standard normal random draw. Note that Step (vi) above corresponds to Step (iv) in Section 3.2.1. Steps (i) – (vi) shown above represent the generator which yields one standard normal random draw. When we want N standard normal random draws, Steps (iv) – (vi) should be repeated N times.

In Step (iii), the source code which computes $\omega(x_i^*)$ for $i = 1, 2, \cdots, n$ and obtains $\Omega_j = \sum_{i=1}^{j} \omega(x_i^*)$ for $j = 1, 2, \cdots, n$ is as follows.

weight2(ix,iy,m,x,prob)

```
1:       subroutine weight2(ix,iy,m,x,prob)
2:       dimension x(0:100001),prob(0:100001)
3: c
4: c  Input:
5: c    m:  The Number of Random Draws from Sampling
6: c        Density (less than or equal to 100000)
7: c    ix, iy:  Seeds
8: c  Output:
9: c    x(i):    Random Draws from Sampling Density
10: c    prob(i): Prob( X\le x(i) )
11: c
12:       prob(0)=0.0
13:         do 1 i=1,m
14:       call urnd(ix,iy,rn)
15:       x(i)=-log(rn)
16:       f0=exp(-x(i))
17:       f1=exp(-0.5*x(i)*x(i))
18:     1 prob(i)=prob(i-1)+f1/f0
19:         do 2 i=1,m
20:     2 prob(i)=prob(i)/prob(m)
21:       return
22:       end
```

In `weight2(ix,iy,m,x,prob)`, The probability weights $\Omega_j = \sum_{i=1}^{j} \omega(x_i^*)$, $j = 1, 2, \cdots, n$, correspond to `prob(j)`, `j=1,2,...,n`. Moreover, x_i^* and n represents `x(i)`

and m, respectively. In Lines 14 and 15 of weight2(ix,iy,m,x,prob), the exponential random draw is generated. Lines 16 and 17 evaluate $f_*(x)$ and a kernel of $f(x)$ at $x = x_i^*$, which are given by f0 and f1. In Line 18, f1/f0 corresponds to $q(x_i^*)$. In Lines 19 and 20, Ω_i is computed for all i, which is denoted by prob(i) in weight2. Thus, weight2 gives us all the probability weights Ω_i, which represents Steps (i) – (iii).

In Steps (iv) and (v), a random draw from $f(x)$ is generated based on Ω_j for $j = 1, 2, \cdots, n$. The Fortran 77 source code for these two steps is given by the subroutine resample(ix,iy,x,prob,m,rn).

resample(ix,iy,x,prob,m,rn)

```
1:       subroutine resample(ix,iy,x,prob,m,rn)
2:       dimension x(0:100001),prob(0:100001)
3: c
4: c  Use "resample(ix,iy,x,prob,m,rn)"
5: c  together with "urnd(ix,iy,rn)".
6: c
7: c  Input:
8: c    x(i):    Random Draws from Sampling Density
9: c    prob(i): Prob( X \le x(i) )
10: c    m:  The Number of Random Draws from Sampling
11: c        Density (less than or equal to 100000)
12: c    ix, iy:  Seeds
13: c  Output:
14: c    rn: Random Draw from Target Density
15: c
16:       call urnd(ix,iy,rn1)
17:         do 1 j=1,m
18:          if(prob(j-1).le.rn1.and.rn1.lt.prob(j)) then
19:       i=j
20:       go to 2
21:          endif
22:     1   continue
23:     2 rn=x(i)
24:       return
25:       end
```

Note that Lines 18 – 24 in snrnd5_2 on p.175 are equivalent to Lines 16 – 22 in resample(ix,iy,x,prob,m,rn). rn1 in Line 16 denotes v_1 in Step (iv). Lines 17 – 23 choose x_i^* with probability $\omega(x_i^*)$, where $x = x_i^*$ in Step (v) is given by rn=x(i). rn gives us the random draw generated from the target density $f(x)$. That is, in this case, rn is a random draw from the half-normal distribution.

Combining the above two source codes, i.e., weight2(ix,iy,m,x,prob) and resample(ix,iy,x,prob,m,rn), an example of the main program for the standard normal random number generator is shown in snrnd7.

snrnd7

```
1:        dimension x(0:100001),prob(0:100001)
2:  c
3:        ix=1
4:        iy=1
5:          do 99 i=1,1000
6:     99 call urnd(ix,iy,rn)
7:  c
8:        n=1000
9:        m=10000
10:       call weight2(ix,iy,m,x,prob)
11:         do 1 i=1,n
12:       call resample(ix,iy,x,prob,m,rn)
13:       call urnd(ix,iy,rn2)
14:       if(rn2.gt.0.5) rn=-rn
15:      1 continue
16:       end
```

`weight2(ix,iy,m,x,prob)` and `resample(ix,iy,x,prob,m,rn)` should be used together with `urnd(ix,iy,rn)` on p.83 for `snrnd7`.

As mentioned in Section 3.1.2 (p.176), in Lines 3 – 6 of `snrnd7`, the first 1000 uniform random draws are excluded from consideration, because the initial values `ix=1` and `iy=1` influences `rn` in the first some iterations. `n=1000` in Line 8 implies that Steps (iv) – (vi) are repeated $N = 1000$ times, i.e., $N = 1000$ standard normal random draws are generated. In Line 9, `m` denotes the number of probability weights, i.e., n in Steps (i) – (vi). Therefore, in Line 10, `m=10000` (i.e., $n = 10000$) probability weights are computed in advance and Ω_i, $i = 1, 2, \cdots, n$, are obtained, which are denoted by `prob(j)` in `resample`. In Line 12, one half-normal random draw `rn` is generated. In Lines 13 and 14, the half-normal random draw is converted to one standard normal random draw, where the positive or negative sign is randomly assigned with equal probability. `rn2` in Line 13 denotes v_2 in Step (vi). In Line 14, `rn` indicates z in Step (vi). In Lines 11 – 14 `n=1000` (i.e., $N = 1000$) standard normal random draws are generated, which corresponds to Step (vi).

From `weight2` and `resample`, we can see that we need a great amount of storage for the probability weights (`m` dimensional vector) and the random draws (`m` dimensional vector) generated from the sampling density.

3.3.2 Gamma Distribution: $G(\alpha, 1)$ for $0 < \alpha \leq 1$

When $X \sim G(\alpha, 1)$, the density function of X is given by:

$$f(x) = \begin{cases} \dfrac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x}, & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise.} \end{cases}$$

The sampling density is taken as:

$$f_*(x) = \frac{e}{\alpha + e}\alpha x^{\alpha-1} I_1(x) + \frac{\alpha}{\alpha + e} e^{-x+1} I_2(x),$$

which is the same function as in `gammarnd2` of Section 3.2.2, where both $I_1(x)$ and $I_2(x)$ denote the indicator functions defined in Section 3.2.2. The probability weights are given by:

$$\begin{aligned}\omega(x_i^*) &= \frac{q(x_i^*)}{\sum_{j=1}^n q(x_j^*)} = \frac{f(x_i^*)/f_*(x_i^*)}{\sum_{j=1}^n f(x_j^*)/f_*(x_j^*)} \\ &= \frac{x_i^{*\alpha-1} e^{-x_i^*}/(x_i^{*\alpha-1} I_1(x_i^*) + e^{-x_i^*} I_2(x_i^*))}{\sum_{j=1}^n x_j^{*\alpha-1} e^{-x_j^*}/(x_j^{*\alpha-1} I_1(x_j^*) + e^{-x_j^*} I_2(x_j^*))},\end{aligned}$$

for $i = 1, 2, \cdots, n$. The cumulative distribution function of $f_*(x)$ is represented as:

$$F_*(x) = \begin{cases} \dfrac{e}{\alpha + e} x^\alpha, & \text{if } 0 < x \le 1, \\ \dfrac{e}{\alpha + e} + \dfrac{\alpha}{\alpha + e}(1 - e^{-x+1}), & \text{if } x > 1. \end{cases}$$

Therefore, x_i^* can be generated by utilizing both the composition method and the inverse transform method. Given x_i^*, compute $\omega(x_i^*)$ for $i = 1, 2, \cdots, n$, and take $x = x_i^*$ with probability $\omega(x_i^*)$. Summarizing above, the random number generation procedure for the gamma distribution is given by:

(i) Generate uniform random draws u_i, $i = 1, 2, \cdots, n$, from $U(0, 1)$, and set $x_i^* = ((\alpha/e + 1)u_i)^{1/\alpha}$ and $\omega(x_i^*) = e^{-x_i^*}$ if $u_i \le e/(\alpha + e)$ and take $x_i^* = -\log((1/e + 1/\alpha)(1 - u_i))$ and $\omega(x_i^*) = x_i^{*\alpha-1}$ if $u_i > e/(\alpha + e)$ for $i = 1, 2, \cdots, n$.

(ii) Compute $\Omega_i = \sum_{j=1}^i \omega(x_j^*)$ for $i = 1, 2, \cdots, n$, where $\Omega_0 = 0$.

(iii) Generate a uniform random draw v from $U(0, 1)$, and take $x = x_i^*$ when $\Omega_{i-1} \le v < \Omega_i$.

As mentioned above, this algorithm yields one random draw. If we want N random draws, Step (iii) should be repeated N times. The Fortran program to compute Ω_i for $i = 1, 2, \cdots, n$, i.e., Steps (i) and (ii) in the above algorithm, is shown in `weight3(ix,iy,alpha,m,x,prob)`, where `alpha`, `m`, `x(i)` and `prob(i)` correspond to α, n, x_i^* and Ω_i, respectively.

`weight3(ix,iy,alpha,m,x,prob)`

```
1:       subroutine weight3(ix,iy,alpha,m,x,prob)
2:       dimension x(0:100001),prob(0:100001)
3: c
4: c  Input:
```

```
5: c    alpha: Parameters
6: c    m:  The Number of Random Draws from Sampling
7: c        Density (less than or equal to 100000)
8: c    ix, iy:  Seeds
9: c  Output:
10: c    x(i):    Random Draws from Sampling Density
11: c    prob(i): Prob( X\le x(i) )
12: c
13:       e=2.71828182845905
14:       prob(0)=0.0
15:         do 1 i=1,m
16:       call urnd(ix,iy,u1)
17:             if( u1.le.e/(alpha+e) ) then
18:       x(i)=( (alpha+e)*u1/e )**(1./alpha)
19:       w=e**(-x(i))
20:             else
21:       x(i)=-log((alpha+e)*(1.-u1)/(alpha*e))
22:       w=x(i)**(alpha-1.)
23:             endif
24:     1 prob(i)=prob(i-1)+w
25:         do 2 i=1,m
26:     2 prob(i)=prob(i)/prob(m)
27:       return
28:       end
```

weight3(ix,iy,alpha,m,x,prob) requires urnd(ix,iy,rn). Lines 16 – 23 give us Step (i) and Lines 24 – 26 indicate Step (ii). Since resample on p.190 indicates Step (iii), weight3 and resample have to be combined in order to generate gamma random draws as shown in gammarnd4. Thus, an example of generating n=1000 gamma random draws is shown in the main program gammarnd4, where n corresponds to N.

gammarnd4

```
1:       dimension x(0:100001),prob(0:100001)
2: c
3:       ix=1
4:       iy=1
5:         do 99 i=1,1000
6:    99 call urnd(ix,iy,rn)
7: c
8:       alpha=0.5
9:       n=1000
10:      m=10000
11:      call weight3(ix,iy,alpha,m,x,prob)
12:        do 1 i=1,n
13:      call resample(ix,iy,x,prob,m,rn)
14:    1 continue
15:      end
```

In gammarnd4, an example of the case $\alpha = 0.5$, $n = 10000$ and $N = 1000$ is shown. 1000 gamma random draws with parameters $\alpha = 0.5$ and $\beta = 1$ are simply just generated, where nothing is done with the random draws.

3.3.3 Beta Distribution

The beta distribution with parameters α and β is of the form:

$$f(x) = \begin{cases} \dfrac{1}{B(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1}, & \text{for } 0 < x < 1, \\ 0, & \text{otherwise.} \end{cases}$$

The sampling density is taken as:

$$f_*(x) = \begin{cases} 1, & \text{for } 0 < x < 1, \\ 0, & \text{otherwise,} \end{cases}$$

which represents the uniform distribution between zero and one. The probability weights $\omega(x_i^*)$, $i = 1, 2, \cdots, n$, are given by:

$$\omega(x_i^*) = \frac{q(x_i^*)}{\sum_{j=1}^n q(x_j^*)} = \frac{f(x_i^*)/f_*(x_i^*)}{\sum_{j=1}^n f(x_j^*)/f_*(x_j^*)} = \frac{x_i^{*\alpha-1}(1-x_i^*)^{\beta-1}}{\sum_{j=1}^n x_j^{*\alpha-1}(1-x_j^*)^{\beta-1}}.$$

Therefore, to generate a random draw from $f(x)$, first generate x_i^*, $i = 1, 2, \cdots, n$, from $U(0,1)$, second compute $\omega(x_i^*)$ for $i = 1, 2, \cdots$,n, and finally take $x = x_i^*$ with probability $\omega(x_i^*)$.

The cumulative probability weight `prob(i)`, which corresponds to $\Omega(x_i^*)$, is computed in `weight4(ix,iy,alpha,beta,m,x,prob)`, where `alpha`, `beta`, `m`, `x(i)` and `prob(i)` are denoted by α, β, n, x_i^* and $\Omega(x_i^*)$, respectively.

`weight4(ix,iy,alpha,beta,m,x,prob)`

```
1:        subroutine weight4(ix,iy,alpha,beta,m,x,prob)
2:        dimension x(0:100001),prob(0:100001)
3:  c
4:  c  Input:
5:  c    alpha, beta: Parameters
6:  c    m:  The Number of Random Draws from Sampling
7:  c        Density (less than or equal to 100000)
8:  c    ix, iy:  Seeds
9:  c  Output:
10: c    x(i):    Random Draws from Sampling Density
11: c    prob(i): Prob( X\le x(i) )
12: c
13:       prob(0)=0.0
14:         do 1 i=1,m
15:       call urnd(ix,iy,x(i))
16:       f0=1.0
17:       f1=( x(i)**(alpha-1.) )*( (1.-x(i))**(beta-1.) )
18:     1 prob(i)=prob(i-1)+f1/f0
19:         do 2 i=1,m
20:     2 prob(i)=prob(i)/prob(m)
21:       return
22:       end
```

`weight4(ix,iy,alpha,beta,m,x,prob)` requires `urnd(ix,iy,rn)`.

Using `weight4` and `resample`, an example of the main program for the beta random draws is shown in `betarnd2`, where parameters $\alpha = \beta = 0.5$ are taken.

betarnd2

```
      dimension x(0:100001),prob(0:100001)
c
      ix=1
      iy=1
        do 99 i=1,1000
   99 call urnd(ix,iy,rn)
c
      alpha=0.5
      beta=0.5
      n=1000
      m=10000
      call weight4(ix,iy,alpha,beta,m,x,prob)
        do 1 i=1,n
      call resample(ix,iy,x,prob,m,rn)
    1 continue
      end
```

We have shown three examples of the importance resampling procedure in this section. One of the advantages of importance resampling is that it is really easy to construct a Fortran source code. `resample(ix,iy,x,prob,m,rn)` on p.190 can be utilized for any distribution. `weight2`, `weight3` or `weight4` has to be properly modified, when we want to generate random draws from the other distributions. However, the disadvantages are that (i) importance resampling takes quite a long time because we have to obtain all the probability weights in advance and (ii) importance resampling requires a great amount of storages for x_i^* and Ω_i for $i = 1, 2, \cdots, n$. For (ii), remember that `x(0:100001)` and `prob(0:100001)` are taken in the example of Section 3.3.1.

3.4 Metropolis-Hastings Algorithm

This section is based on Geweke and Tanizaki (2003), where three sampling distributions are compared with respect to precision of the random draws from the target density $f(x)$.

The **Metropolis-Hastings algorithm** is also one of the sampling methods to generate random draws from any target density $f(x)$, utilizing sampling density $f_*(x)$, even in the case where it is not easy to generate random draws from the target density.

Let us define the acceptance probability by:

$$\omega(x_{i-1}, x^*) = \min\Big(\frac{q(x^*)}{q(x_{i-1})}, 1\Big) = \min\Big(\frac{f(x^*)/f_*(x^*)}{f(x_{i-1})/f_*(x_{i-1})}, 1\Big),$$

where $q(\cdot)$ is defined as equation (3.2). By the Metropolis-Hastings algorithm, a random draw from $f(x)$ is generated in the following way:

(i) Take the initial value of x as x_{-M}.
(ii) Generate x^* from $f_*(x)$ and compute $\omega(x_{i-1}, x^*)$ given x_{i-1}.
(iii) Set $x_i = x^*$ with probability $\omega(x_{i-1}, x^*)$ and $x_i = x_{i-1}$ otherwise.
(iv) Repeat Steps (ii) and (iii) for $i = -M+1, -M+2, \cdots, 1$.

In the above algorithm, x_1 is taken as a random draw from $f(x)$. When we want more random draws (say, N), we replace Step (iv) by Step (iv)', which is represented as follows:

(iv)' Repeat Steps (ii) and (iii) for $i = -M+1, -M+2, \cdots, N$.

When we implement Step (iv)', we can obtain a series of random draws x_{-M}, x_{-M+1}, $\cdots$, x_0, x_1, x_2, $\cdots$, x_N, where x_{-M}, x_{-M+1}, $\cdots$, x_0 are discarded from further consideration. The last N random draws are taken as the random draws generated from the target density $f(x)$. Thus, N denotes the number of random draws. M is sometimes called the **burn-in period**.

We can justify the above algorithm given by Steps (i) – (iv) as follows. The proof is very similar to the case of rejection sampling in Section 3.2. We show that x_i is the random draw generated from the target density $f(x)$ under the assumption x_{i-1} is generated from $f(x)$. Let U be the uniform random variable between zero and one, X be the random variable which has the density function $f(x)$ and x^* be the realization (i.e., the random draw) generated from the sampling density $f_*(x)$. Consider the probability $P(X \le x | U \le \omega(x_{i-1}, x^*))$, which should be the cumulative distribution of X, i.e., $F(x)$. The probability $P(X \le x | U \le \omega(x_{i-1}, x^*))$ is rewritten as follows:

$$P(X \le x | U \le \omega(x_{i-1}, x^*)) = \frac{P(X \le x, U \le \omega(x_{i-1}, x^*))}{P(U \le \omega(x_{i-1}, x^*))},$$

where the numerator is represented as:

$$\begin{aligned}
P(X \le x, U \le \omega(x_{i-1}, x^*)) &= \int_{-\infty}^{x} \int_{0}^{\omega(x_{i-1},t)} f_{u,*}(u,t)\, \mathrm{d}u\, \mathrm{d}t \\
&= \int_{-\infty}^{x} \int_{0}^{\omega(x_{i-1},t)} f_u(u) f_*(t)\, \mathrm{d}u\, \mathrm{d}t = \int_{-\infty}^{x} \Big(\int_{0}^{\omega(x_{i-1},t)} f_u(u)\, \mathrm{d}u \Big) f_*(t)\, \mathrm{d}t \\
&= \int_{-\infty}^{x} \Big(\int_{0}^{\omega(x_{i-1},t)} \mathrm{d}u \Big) f_*(t)\, \mathrm{d}t = \int_{-\infty}^{x} \Big[u \Big]_{0}^{\omega(x_{i-1},t)} f_*(t)\, \mathrm{d}t \\
&= \int_{-\infty}^{x} \omega(x_{i-1}, t) f_*(t)\, \mathrm{d}t = \int_{-\infty}^{x} \frac{f_*(x_{i-1}) f(t)}{f(x_{i-1})}\, \mathrm{d}t = \frac{f_*(x_{i-1})}{f(x_{i-1})} F(x)
\end{aligned}$$

and the denominator is given by:

$$P(U \leq \omega(x_{i-1}, x^*)) = P(X \leq \infty, U \leq \omega(x_{i-1}, x^*)) = \frac{f_*(x_{i-1})}{f(x_{i-1})} F(\infty) = \frac{f_*(x_{i-1})}{f(x_{i-1})}.$$

The density function of U is given by $f_u(u) = 1$ for $0 < u < 1$. Let X^* be the random variable which has the density function $f_*(x)$. In the numerator, $f_{u,*}(u, x)$ denotes the joint density of random variables U and X^*. Because the random draws of U and X^* are independently generated, we have $f_{u,*}(u, x) = f_u(u) f_*(x) = f_*(x)$. Thus, the first four equalities are derived. Substituting the numerator and denominator shown above, we have the following equality:

$$P(X \leq x | U \leq \omega(x_{i-1}, x^*)) = F(x).$$

Thus, the x^* which satisfies $u \leq \omega(x_{i-1}, x^*)$ indicates a random draw from $f(x)$. We set $x_i = x_{i-1}$ if $u \leq \omega(x_{i-1}, x^*)$ is not satisfied. x_{i-1} is already assumed to be a random draw from $f(x)$. Therefore, it is shown that x_i is a random draw from $f(x)$. See Gentle (1998) for the discussion above.

As in the case of rejection sampling and importance resampling, note that $f(x)$ may be a kernel of the target density, or equivalently, $f(x)$ may be proportional to the target density. The same algorithm as Steps (i) – (iv) can be applied to the case where $f(x)$ is proportional to the target density, because $f(x^*)$ is divided by $f(x_{i-1})$ in $\omega(x_{i-1}, x^*)$.

As a general formulation of the sampling density, instead of $f_*(x)$, we may take the sampling density as the following form: $f_*(x|x_{i-1})$, where a candidate random draw x^* depends on the $(i-1)$th random draw, i.e., x_{i-1}.

For choice of the sampling density $f_*(x|x_{i-1})$, Chib and Greenberg (1995) pointed out as follows. $f_*(x|x_{i-1})$ should be chosen so that the chain travels over the support of $f(x)$, which implies that $f_*(x|_{i-1})$ should not have too large variance and too small variance, compared with $f(x)$. See, for example, Smith and Roberts (1993), Bernardo and Smith (1994), O'Hagan (1994), Tierney (1994), Geweke (1996), Gamerman (1997), Robert and Casella (1999) and so on for the Metropolis-Hastings algorithm.

As an alternative justification, note that the Metropolis-Hastings algorithm is formulated as follows:

$$f_i(u) = \int f^*(u|v) f_{i-1}(v) \, \mathrm{d}v,$$

where $f^*(u|v)$ denotes the transition distribution, which is characterized by Step (iii). x_{i-1} is generated from $f_{i-1}(\cdot)$ and x_i is from $f^*(\cdot|x_{i-1})$. x_i depends only on x_{i-1}, which is called the **Markov property**. The sequence $\{\cdots, x_{i-1}, x_i, x_{i+1}, \cdots\}$ is called the **Markov chain**. The Monte Carlo statistical methods with the sequence $\{\cdots, x_{i-1}, x_i, x_{i+1}, \cdots\}$ is called the **Markov chain Monte Carlo (MCMC)**. From Step (iii), $f^*(u|v)$ is given by:

$$f^*(u|v) = \omega(v, u) f_*(u|v) + \Big(1 - \int \omega(v, u) f_*(u|v) \, \mathrm{d}u\Big) p(u), \tag{3.3}$$

where $p(x)$ denotes the following probability function:

$$p(u) = \begin{cases} 1, & \text{if } u = v, \\ 0, & \text{otherwise.} \end{cases}$$

Thus, x is generated from $f_*(u|v)$ with probability $\omega(v,u)$ and from $p(u)$ with probability $1 - \int \omega(v,u) f_*(u|v)\, \mathrm{d}u$. Now, we want to show $f_i(u) = f_{i-1}(u) = f(u)$ as i goes to infinity, which implies that both x_i and x_{i-1} are generated from the invariant distribution function $f(u)$ for sufficiently large i. To do so, we need to consider the condition satisfying the following equation:

$$f(u) = \int f^*(u|v) f(v)\, \mathrm{d}v. \tag{3.4}$$

Equation (3.4) holds if we have the following equation:

$$f^*(v|u) f(u) = f^*(u|v) f(v), \tag{3.5}$$

which is called the **reversibility condition**. By taking the integration with respect to v on both sides of equation (3.5), equation (3.4) is obtained. Therefore, we have to check whether the $f^*(u|v)$ shown in equation (3.3) satisfies equation (3.5). It is straightforward to verify that

$$\begin{aligned} &\omega(v,u) f_*(u|v) f(v) = \omega(u,v) f_*(v|u) f(u), \\ &\Big(1 - \int \omega(v,u) f_*(u|v)\, \mathrm{d}u\Big) p(u) f(v) = \Big(1 - \int \omega(u,v) f_*(v|u)\, \mathrm{d}v\Big) p(v) f(u). \end{aligned}$$

Thus, as i goes to infinity, x_i is a random draw from the target density $f(\cdot)$. If x_i is generated from $f(\cdot)$, then x_{i+1} is also generated from $f(\cdot)$. Therefore, all the x_i, x_{i+1}, x_{i+2}, $\cdots$ are taken as random draws from the target density $f(\cdot)$.

The requirement for uniform convergence of the Markov chain is that the chain should be **irreducible** and **aperiodic**. See, for example, Roberts and Smith (1993). Let $C_i(x_0)$ be the set of possible values of x_i from starting point x_0. If there exist two possible starting values, say x^* and x^{**}, such that $C_i(x^*) \cap C_i(x^{**}) = \emptyset$ (i.e., empty set) for all i, then the same limiting distribution cannot be reached from both starting points. Thus, in the case of $C_i(x^*) \cap C_i(x^{**}) = \emptyset$, the convergence may fail. A Markov chain is said to be **irreducible** if there exists an i such that $P(x_i \in C|x_0) > 0$ for any starting point x_0 and any set C such that $\int_C f(x)\, \mathrm{d}x > 0$. The irreducible condition ensures that the chain can reach all possible x values from any starting point. Moreover, as another case in which convergence may fail, if there are two disjoint set C^1 and C^2 such that $x_{i-1} \in C^1$ implies $x_i \in C^2$ and $x_{i-1} \in C^2$ implies $x_i \in C^1$, then the chain oscillates between C^1 and C^2 and we again have $C_i(x^*) \cap C_i(x^{**}) = \emptyset$ for all i when $x^* \in C^1$ and $x^{**} \in C^2$. Accordingly, we cannot have the same limiting distribution in this case, either. It is called **aperiodic** if the chain does not oscillate between two sets C^1 and C^2 or cycle around a partition C^1, C^2, $\cdots$, C^r of r disjoint sets for $r > 2$. See O'Hagan (1994) for the discussion above.

For the Metropolis-Hastings algorithm, x_1 is taken as a random draw of x from $f(x)$ for sufficiently large M. To obtain N random draws, we need to generate $M + N$ random draws. Moreover, clearly we have $\text{Cov}(x_{i-1}, x_i) > 0$, because x_i is generated based on x_{i-1} in Step (iii). Therefore, for precision of the random draws, the Metropolis-Hastings algorithm gives us the worst random number of the three sampling methods. i.e., rejection sampling in Section 3.2, importance resampling in Section 3.3 and the Metropolis-Hastings algorithm in this section.

Based on Steps (i) – (iii) and (iv)', under some conditions the basic result of the Metropolis-Hastings algorithm is as follows:

$$\frac{1}{N}\sum_{i=1}^{N} g(x_i) \longrightarrow \text{E}(g(x)) = \int g(x) f(x)\, \text{d}x, \qquad \text{as } N \longrightarrow \infty,$$

where $g(\cdot)$ is a function, which is representatively taken as $g(x) = x$ for mean and $g(x) = (x - \overline{x})^2$ for variance. $\overline{x}$ denotes $\overline{x} = (1/N)\sum_{i=1}^{N} x_i$. Thus, it is shown that $(1/N)\sum_{i=1}^{N} g(x_i)$ is a consistent estimate of $\text{E}(g(x))$, even though x_1, x_2, $\cdots$, x_N are mutually correlated.

As an alternative random number generation method to avoid the positive correlation, we can perform the case of $N = 1$ as in the above procedures (i) – (iv) N times in parallel, taking different initial values for x_{-M}. In this case, we need to generate $M + 1$ random numbers to obtain one random draw from $f(x)$. That is, N random draws from $f(x)$ are based on $N(1 + M)$ random draws from $f_*(x|x_{i-1})$. Thus, we can obtain mutually independently distributed random draws. For precision of the random draws, the alternative Metropolis-Hastings algorithm should be similar to rejection sampling. However, this alternative method is too computer-intensive, compared with the above procedures (i) – (iii) and (iv)', which takes more time than rejection sampling in the case of $M > N_R$.

Furthermore, the sampling density has to satisfy the following conditions: (i) we can quickly and easily generate random draws from the sampling density and (ii) the sampling density should be distributed with the same range as the target density. See, for example, Geweke (1992) and Mengersen, Robert and Guihenneuc-Jouyaux (1999) for the MCMC convergence diagnostics. Since the random draws based on the Metropolis-Hastings algorithm heavily depend on choice of the sampling density, we can see that the Metropolis-Hastings algorithm has the problem of specifying the sampling density, which is the crucial criticism. Several generic choices of the sampling density are discussed by Tierney (1994) and Chib and Greenberg (1995). We can consider several candidates for the sampling density $f_*(x|x_{i-1})$, i.e., Sampling Densities I – III.

3.4.1.1 Sampling Density I (Independence Chain)

For the sampling density, we have started with $f_*(x)$ in this section. Thus, one possibility of the sampling density is given by: $f_*(x|x_{i-1}) = f_*(x)$, where $f_*(\cdot)$ does not

depend on x_{i-1}. This sampling density is called the **independence chain**. For example, it is possible to take $f_*(x) = N(\mu_*, \sigma_*^2)$, where μ_* and σ_*^2 are the hyper-parameters. Or, when x lies on a certain interval, say (a, b), we can choose the uniform distribution $f_*(x) = 1/(b-a)$ for the sampling density.

3.4.1.2 Sampling Density II (Random Walk Chain)

We may take the sampling density called the **random walk chain**, i.e., $f_*(x|x_{i-1}) = f_*(x - x_{i-1})$. Representatively, we can take the sampling density as $f_*(x|x_{i-1}) = N(x_{i-1}, \sigma_*^2)$, where σ_*^2 denotes the hyper-parameter. Based on the random walk chain, we have a series of the random draws which follow the random walk process.

3.4.1.3 Sampling Density III (Taylored Chain)

The alternative sampling distribution is based on approximation of the log-kernel (see Geweke and Tanizaki (1999, 2001, 2003)), which is a substantial extension of the **Taylored chain** discussed in Chib, Greenberg and Winkelmann (1998). Let $p(x) = \log(f(x))$, where $f(x)$ may denote the kernel which corresponds to the target density. Approximating the log-kernel $p(x)$ around x_{i-1} by the second order Taylor series expansion, $p(x)$ is represented as:

$$p(x) \approx p(x_{i-1}) + p'(x_{i-1})(x - x_{i-1}) + \frac{1}{2}p''(x_{i-1})(x - x_{i-1})^2, \tag{3.6}$$

where $p'(\cdot)$ and $p''(\cdot)$ denote the first- and second-derivatives. Depending on the values of $p'(x)$ and $p''(x)$, we have the four cases, i.e., Cases 1 – 4, which are classified by (i) $p''(x) < -\epsilon$ in Case 1 or $p''(x) \geq -\epsilon$ in Cases 2 – 4 and (ii) $p'(x) < 0$ in Case 2, $p'(x) > 0$ in Case 3 or $p'(x) = 0$ in Case 4. Geweke and Tanizaki (2003) suggested introducing ϵ into the Taylored chain discussed in Geweke and Tanizaki (1999, 2001). Note that $\epsilon = 0$ is chosen in Geweke and Tanizaki (1999, 2001). To improve precision of random draws, ϵ should be a positive value, which will be discussed later in detail (see Remark 1 for ϵ). In Monte Carlo studies of Section 3.7.5, $\epsilon = 0.0, 0.2, 0.3, 0.4$ are taken.

Case 1: $p''(x_{i-1}) < -\epsilon$: Equation (3.6) is rewritten by:

$$p(x) \approx p(x_{i-1}) - \frac{1}{2}\Big(\frac{1}{-1/p''(x_{i-1})}\Big)\Big(x - (x_{i-1} - \frac{p'(x_{i-1})}{p''(x_{i-1})})\Big)^2 + r(x_{i-1}),$$

where $r(x_{i-1})$ is an appropriate function of x_{i-1}. Since $p''(x_{i-1})$ is negative, the second term in the right-hand side is equivalent to the exponential part of the normal density. Therefore, $f_*(x|x_{i-1})$ is taken as $N(\mu_*, \sigma_*^2)$, where $\mu_* = x_{i-1} - p'(x_{i-1})/p''(x_{i-1})$ and $\sigma_*^2 = -1/p''(x_{i-1})$.

Case 2: $p''(x_{i-1}) \geq -\epsilon$ **and** $p'(x_{i-1}) < 0$: Perform linear approximation of $p(x)$. Let x^+ be the nearest mode with $x^+ < x_{i-1}$. Then, $p(x)$ is approximated by a line passing between x^+ and x_{i-1}, which is written as:

$$p(x) \approx p(x^+) + \frac{p(x^+) - p(x_{i-1})}{x^+ - x_{i-1}}(x - x^+).$$

From the second term in the right-hand side, the sampling density is represented as the exponential distribution with $x > x^+ - d$, i.e., $f_*(x|x_{i-1}) = \lambda \exp\Big(-\lambda(x - (x^+ - d))\Big)$ if $x^+ - d < x$ and $f_*(x|x_{i-1}) = 0$ otherwise, where λ is defined as:

$$\lambda = \left|\frac{p(x^+) - p(x_{i-1})}{x^+ - x_{i-1}}\right|.$$

d is a positive value, which will be discussed later (see Remark 2 for d). Thus, a random draw x^* from the sampling density is generated by $x^* = w + (x^+ - d)$, where w represents the exponential random variable with parameter λ.

Case 3: $p''(x_{i-1}) \geq -\epsilon$ **and** $p'(x_{i-1}) > 0$: Similarly, perform linear approximation of $p(x)$ in this case. Let x^+ be the nearest mode with $x_{i-1} < x^+$. Approximation of $p(x)$ is exactly equivalent to that of Case 2. Taking into account $x < x^+ + d$, the sampling density is written as: $f_*(x|x_{i-1}) = \lambda \exp\Big(-\lambda((x^+ + d) - x)\Big)$ if $x < x^+ + d$ and $f_*(x|x_{i-1}) = 0$ otherwise. Thus, a random draw x^* from the sampling density is generated by $x^* = (x^+ + d) - w$, where w is distributed as the exponential random variable with parameter λ.

Case 4: $p''(x_{i-1}) \geq -\epsilon$ **and** $p'(x_{i-1}) = 0$: In this case, $p(x)$ is approximated as a uniform distribution at the neighborhood of x_{i-1}. As for the range of the uniform distribution, we utilize the two appropriate values x^+ and x^{++}, which satisfies $x^+ < x < x^{++}$. When we have two modes, x^+ and x^{++} may be taken as the modes. Thus, the sampling density $f_*(x|x_{i-1})$ is obtained by the uniform distribution on the interval between x^+ and x^{++}, i.e., $f_*(x|x_{i-1}) = 1/(x^{++} - x^+)$ if $x^+ < x < x^{++}$ and $f_*(x|x_{i-1}) = 0$ otherwise.

Thus, for approximation of the kernel, all the possible cases are given by Cases 1 – 4, depending on the values of $p'(\cdot)$ and $p''(\cdot)$. Moreover, in the case where x is a vector, applying the procedure above to each element of x, Sampling III is easily extended to multivariate cases. Finally, we discuss about ϵ and d in the following remarks.

Remark 1: ϵ in Cases 1 – 4 should be taken as an appropriate positive number. In the simulation studies of Section 3.7.5, we choose $\epsilon = 0.0, 0.2, 0.3, 0.4$. It may seem more natural to take $\epsilon = 0$, rather than $\epsilon > 0$. The reason why $\epsilon > 0$ is taken is as follows. Consider the case of $\epsilon = 0$. When $p''(x_{i-1})$ is negative and it is very close to zero, variance σ_*^2 in Case 1 becomes extremely large because of

$\sigma_*^2 = -1/p''(x_{i-1})$. In this case, the obtained random draws are too broadly distributed and accordingly they become unrealistic, which implies that we have a lot of outliers. To avoid this situation, ϵ should be positive. It might be appropriate that ϵ should depend on variance of the target density, because ϵ should be small if variance of the target density is large. Thus, in order to reduce a number of outliers, $\epsilon > 0$ is recommended.

Remark 2: For d in Cases 2 and 3, note as follows. As an example, consider the unimodal density in which we have Cases 2 and 3. Let x^+ be the mode. We have Case 2 in the right-hand side of x^+ and Case 3 in the left-hand side of x^+. In the case of $d = 0$, we have the random draws generated from either Case 2 or 3. In this situation, the generated random draw does not move from one case to another. In the case of $d > 0$, however, the distribution in Case 2 can generate a random draw in Case 3. That is, for positive d, the generated random draw may move from one case to another, which implies that the irreducibility condition of the MH algorithm is guaranteed. In the simulation studies of Section 3.7.5, We take $d = 1/\lambda$, which represents the standard error of the exponential distribution with parameter λ.

3.4.1 Normal Distribution: $N(0, 1)$

As in Sections 3.2.1 and 3.3.1, we consider an example of generating standard normal random draws based on the half-normal distribution:

$$f(x) = \begin{cases} \dfrac{2}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}, & \text{for } 0 \le x < \infty, \\ 0, & \text{otherwise.} \end{cases}$$

As in Sections 3.2.1 and 3.3.1, we take the sampling density as the following exponential distribution:

$$f_*(x) = \begin{cases} e^{-x}, & \text{for } 0 \le x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

which is the independence chain, i.e., $f_*(x|x_{i-1}) = f_*(x)$. Then, the acceptance probability $\omega(x_{i-1}, x^*)$ is given by:

$$\begin{aligned} \omega(x_{i-1}, x^*) &= \min\Big(\frac{f(x^*)/f_*(x^*)}{f(x_{i-1})/f_*(x_{i-1})}, 1\Big) \\ &= \min\Big(\exp(-\frac{1}{2}x^{*2} + x^* + \frac{1}{2}x_{i-1}^2 - x_{i-1}), 1\Big). \end{aligned}$$

Utilizing the Metropolis-Hastings algorithm, the standard normal random number generator is shown as follows:

(i) Take an appropriate initial value of x as x_{-M} (for example, $x_{-M} = 0$).

(ii) Set $y_{i-1} = |x_{i-1}|$.

(iii) Generate a uniform random draw u_1 from $U(0,1)$ and compute $\omega(y_{i-1}, y^*)$ where $y^* = -\log(u_1)$.

(iv) Generate a uniform random draw u_2 from $U(0,1)$, and set $y_i = y^*$ if $u_2 \leq \omega(y_{i-1}, y^*)$ and $y_i = y_{i-1}$ otherwise.

(v) Generate a uniform random draw u_3 from $U(0,1)$, and set $x_i = y_i$ if $u_3 \leq 0.5$ and $x_i = -y_i$ otherwise.

(vi) Repeat Steps (ii) – (v) for $i = -M+1, -M+2, \cdots, 1$.

y_1 is taken as a random draw from $f(x)$. M denotes the burn-in period. If a lot of random draws (say, N random draws) are required, we replace Step (vi) by Step (vi)' represented as follows:

(vi)' Repeat Steps (ii) – (v) for $i = -M+1, -M+2, \cdots, N$.

In Steps (ii) – (iv), a half-normal random draw is generated. Note that the absolute value of x_{i-1} is taken in Step (ii) because the half-normal random draw is positive. In Step (v), the positive or negative sign is randomly assigned to y_i. The source code for Steps (ii) – (v) is represented by `snrnd8(ix,iy,rn)`.

`snrnd8(ix,iy,rn)`

```
1:        subroutine snrnd8(ix,iy,rn)
2:  c
3:  c  Use "snrnd8(ix,iy,rn)"
4:  c  together with "urnd(ix,iy,rn)".
5:  c
6:  c  Input:
7:  c    rn:     Previously Generated Random Draw
8:  c    ix, iy: Seeds
9:  c  Output:
10: c    rn: Standard Normal Random Draw
11: c
12:       rn0=abs(rn)
13:       call urnd(ix,iy,u1)
14:       rn1=-log(u1)
15:       w=exp( (-0.5*rn1*rn1+rn1)-(-0.5*rn0*rn0+rn0) )
16:       call urnd(ix,iy,u2)
17:       if(u2.le.w) rn=rn1
18:       call urnd(ix,iy,u3)
19:       if(u3.gt.0.5) rn=-rn
20:       return
21:       end
```

Note that `snrnd8(ix,iy,rn)` should be used together with `urnd(ix,iy,rn)` on p.83. Line 12 corresponds to Step (ii) and the exponential random draw in Step (iii) is obtained from Lines 13 and 14. `w` in Line 15 represents the acceptance probability $\omega(x_{i-1}, x^*)$. Lines 16 and 17 is Step (iv), and Lines 18 and 19 gives us Step (v).

An example of the usage of `snrnd8(ix,iy,rn)` is shown in the following program titled by `Main Program for snrnd8`, where 5000 + 10000 random draws are generated.

Main Program for snrnd8

```
1:        ix=1
2:        iy=1
3:          do 99 i=1,1000
4:     99 call urnd(ix,iy,rn)
5:  c
6:        m=5000
7:          do 1 i=1,m
8:      1 call snrnd8(ix,iy,rn)
9:        n=10000
10:         do 2 i=1,n
11:       call snrnd8(ix,iy,rn)
12:     2 continue
13:       end
```

m and n correspond to M and N, respectively. $M = 5000$ burn-in periods are taken (see Line 6), i.e., $M = 5000$ random draws are abandoned. $N = 10000$ random draws are actually used for analysis and investigation (see Line 9). In the above example, $N = 10000$ random draws are simply just generated, where nothing is done with the random draws. Note that the 1000th uniform random draw in Line 4 is the initial value denoted by x_{-M}, which gives us Step (i). Lines 7 and 8 yield Step (vi), while Lines 6 – 12 correspond to Step (vi)'.

3.4.2 Gamma Distribution: $G(\alpha, 1)$ for $0 < \alpha \leq 1$

When $X \sim G(\alpha, 1)$, the density function of X is given by:

$$f(x) = \begin{cases} \dfrac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x}, & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise.} \end{cases}$$

As in gammarnd2 of Sections 3.2.2 and gammarnd4 of 3.3.2, the sampling density is taken as:

$$f_*(x) = \frac{e}{\alpha + e} \alpha x^{\alpha-1} I_1(x) + \frac{\alpha}{\alpha + e} e^{-x+1} I_2(x),$$

where both $I_1(x)$ and $I_2(x)$ denote the indicator functions defined in Section 3.2.2. Then, the acceptance probability is given by:

$$\begin{aligned} \omega(x_{i-1}, x^*) &= \min\Big(\frac{q(x^*)}{q(x_{i-1})}, 1\Big) = \min\Big(\frac{f(x^*)/f_*(x^*)}{f(x_{i-1})/f_*(x_{i-1})}, 1\Big) \\ &= \min\Big(\frac{x^{*\alpha-1} e^{-x^*} / (x^{*\alpha-1} I_1(x^*) + e^{-x^*} I_2(x^*))}{x_{i-1}^{\alpha-1} e^{-x_{i-1}} / (x_{i-1}^{\alpha-1} I_1(x_{i-1}) + e^{-x_{i-1}} I_2(x_{i-1}))}, 1\Big). \end{aligned}$$

As shown in Section 3.2.2, the cumulative distribution function of $f_*(x)$ is represented as:

$$F_*(x) = \begin{cases} \dfrac{e}{\alpha+e}x^{\alpha}, & \text{if } 0 < x \leq 1, \\ \dfrac{e}{\alpha+e} + \dfrac{\alpha}{\alpha+e}(1-e^{-x+1}), & \text{if } x > 1. \end{cases}$$

Therefore, a candidate of the random draw, i.e., x^*, can be generated from $f_*(x)$, by utilizing both the composition method and the inverse transform method. Then, using the Metropolis-Hastings algorithm, the gamma random number generation method is shown as follows.

(i) Take an appropriate initial value as x_{-M}.

(ii) Generate a uniform random draw u_1 from $U(0,1)$, and set $x^* = ((\alpha/e+1)u_1)^{1/\alpha}$ if $u_1 \leq e/(\alpha+e)$ and $x^* = -\log((1/e+1/\alpha)(1-u_1))$ if $u_1 > e/(\alpha+e)$.

(iii) Compute $\omega(x_{i-1}, x^*)$.

(iv) Generate a uniform random draw u_2 from $U(0,1)$, and set $x_i = x^*$ if $u_2 \leq \omega(x_{i-1}, x^*)$ and $x_i = x_{i-1}$ otherwise.

(v) Repeat Steps (ii) – (iv) for $i = -M+1, -M+2, \cdots, 1$.

For sufficiently large M, x_1 is taken as a random draw from $f(x)$. u_1 and u_2 should be independently distributed. M denotes the burn-in period. If we need a lot of random draws (say, N random draws), replace Step (v) by Step (v)', which is given by:

(v)' Repeat Steps (ii) – (iv) for $i = -M+1, -M+2, \cdots, N$.

The source code for the above algorithm (ii) – (iv), written by Fortran 77, is shown in `gammarnd5(ix,iy,alpha,rn)`. In Lines 16 and 19, x^* is generated from the sampling density $f_*(x)$, where x^* is denoted by `rn1`. In Lines 17 and 20, $q(x^*)$ is computed, where $q(x^*)$ is represented by `q1`. In Lines 22 – 26, $q(x_{i-1})$ is obtained, where $q(x_{i-1})$ is given by `q`. Line 28 corresponds to Step (iv), where $\omega(x_{i-1}, x^*)$ indicates `q1/q`.

gammarnd5(ix,iy,alpha,rn)

```
1:        subroutine gammarnd5(ix,iy,alpha,rn)
2:  C
3:  C  Use "gammarnd5(ix,iy,alpha,rn)"
4:  C  together with "urnd(ix,iy,rn)".
5:  C
6:  C  Input:
7:  C    alpha:  Parameter (alpha \le 1)
8:  C    rn:     Previously Generated Random Draw
9:  C    ix, iy: Seeds
10: C  Output:
11: C    rn: Gamma Random Draw with alpha and beta=1
12: C
13:       e=2.71828182845905
```

```
14:       call urnd(ix,iy,u1)
15:         if( u1.le.e/(alpha+e) ) then
16:       rn1=( (alpha+e)*u1/e )**(1./alpha)
17:       q1=e**(-rn1)
18:         else
19:       rn1=-log((alpha+e)*(1.-u1)/(alpha*e))
20:       q1=rn1**(alpha-1.)
21:         endif
22:         if( rn.le.1. ) then
23:       q=e**(-rn)
24:         else
25:       q=rn**(alpha-1.)
26:         endif
27:       call urnd(ix,iy,u2)
28:       if( u2.le.q1/q ) rn=rn1
29:       return
30:       end
```

When `snrnd8(ix,iy,rn)` is replaced by `gammarnd5(ix,iy,alpha,rn)` in Lines 8 and 11 of the main program `Main Program for snrnd8` on p.203, we can obtain the main program for the gamma random number generator with parameters $0 < \alpha \leq 1$ and $\beta = 1$. As shown in Line 8 of `gammarnd4` on p.193, when `snrnd8(ix,iy,rn)` is replaced by `gammarnd5(ix,iy,alpha,rn)` in Lines 8 and 11 of the main program `Main Program for snrnd8` on p.203, we have to give some value to α before Line 7 of the main program on p.203.

3.4.3 Beta Distribution

The beta distribution with parameters α and β is of the form:

$$f(x) = \begin{cases} \dfrac{1}{B(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1}, & \text{for } 0 < x < 1, \\ 0, & \text{otherwise.} \end{cases}$$

The sampling density is taken as:

$$f_*(x) = \begin{cases} 1, & \text{for } 0 < x < 1, \\ 0, & \text{otherwise,} \end{cases}$$

which represents the uniform distribution between zero and one. The probability weights $\omega(x_i^*)$, $i = 1, 2, \cdots, n$, are given by:

$$\omega(x_{i-1}, x^*) = \min\Big(\frac{f(x^*)/f_*(x^*)}{f(x_{i-1})/f_*(x_{i-1})}, 1\Big) = \min\Big(\Big(\frac{x^*}{x_{i-1}}\Big)^{\alpha-1}\Big(\frac{1-x^*}{1-x_{i-1}}\Big)^{\beta-1}, 1\Big).$$

Then, utilizing the Metropolis-Hastings algorithm, the random draws are generated as follows.

(i) Take an appropriate initial value as x_{-M}.

(ii) Generate a uniform random draw x^* from $U(0,1)$, and compute $\omega(x_{i-1}, x^*)$.

(iii) Generate a uniform random draw u from $U(0,1)$, which is independent of x^*, and set $x_i = x^*$ if $u \leq \omega(x_{i-1}, x^*)$ and $x_i = x_{i-1}$ if $u > \omega(x_{i-1}, x^*)$.

(iv) Repeat Steps (ii) and (iii) for $i = -M+1, -M+2, \cdots, 1$.

For sufficiently large M, x_1 is taken as a random draw from $f(x)$. M denotes the burn-in period. If we want a lot of random draws (say, N random draws), replace Step (iv) by Step (iv)', which is represented as follows:

(iv)' Repeat Steps (ii) and (iii) for $i = -M+1, -M+2, \cdots, N$.

Therefore, the random number generator based on Steps (i) – (iv) is given by the subroutine `betarnd3(ix,iy,alpha,beta,rn)`.

`betarnd3(ix,iy,alpha,beta,rn)`

```
1:        subroutine betarnd3(ix,iy,alpha,beta,rn)
2:  c
3:  c  Use "betarnd3(ix,iy,alpha,beta,rn)"
4:  c  together with "urnd(ix,iy,rn)".
5:  c
6:  c  Input:
7:  c    alpha, beta:     Parameters
8:  c    rn:     Previously Generated Random Draw
9:  c    ix, iy: Seeds
10: c  Output:
11: c    rn: Standard Normal Random Draw
12: c
13:       call urnd(ix,iy,rn1)
14:       w=( (rn1/rn)**(alpha-1.) )
15:      & *( ((1.-rn1)/(1.-rn))**(beta-1.) )
16:       call urnd(ix,iy,u2)
17:       if(u2.le.w) rn=rn1
18:       return
19:       end
```

Note that `betarnd3(ix,iy,alpha,beta,rn)` should be used simultaneously with `urnd(ix,iy,rn)` on p.83. `rn1` in Line 13 denotes the random draw generated from the sampling density $f_*(x) = 1$ for $0 < x < 1$. In Lines 14 and 15, `w` is obtained, which is the probability weight $\omega(x_{i-1}, x^*)$. Lines 13 – 15 gives us Step (ii). Lines 16 and 17 correspond to Step (iii).

When `snrnd8(ix,iy,rn)` is replaced by `betarnd3(ix,iy,alpha,beta,rn)` in Lines 8 and 11 of the main program `Main Program for snrnd8` on p.203, we can obtain the main program for the beta random number generator with parameters α and β. As shown in Lines 8 and 9 of `betarnd2` (p.195), when `snrnd8(ix,iy,rn)` is replaced by `betarnd3(ix,iy,alpha,beta,rn)` in Lines 8 and 11 of the main program `Main Program for snrnd8` on p.203, we have to give some values to α and β before Line 7 of the main program on p.203.

3.5 Ratio-of-Uniforms Method

As an alternative random number generation method, in this section we introduce the **ratio-of-uniforms method**. This generation method does not require the sampling density utilized in rejection sampling (Section 3.2), importance resampling (Section 3.3) and the Metropolis-Hastings algorithm (Section 3.4).

Suppose that a bivariate random variable (U_1, U_2) is uniformly distributed, which satisfies the following inequality:

$$0 \leq U_1 \leq \sqrt{h(U_2/U_1)},$$

for any nonnegative function $h(x)$. Then, $X = U_2/U_1$ has a density function $f(x) = h(x)/\int h(x)\,\mathrm{d}x$. Note that the domain of (U_1, U_2) will be discussed below.

The above random number generation method is justified in the following way. The joint density of U_1 and U_2, denoted by $f_{12}(u_1, u_2)$, is given by:

$$f_{12}(u_1, u_2) = \begin{cases} k, & \text{if } 0 \leq u_1 \leq \sqrt{h(u_2/u_1)}, \\ 0, & \text{otherwise,} \end{cases}$$

where k is a constant value, because the bivariate random variable (U_1, U_2) is uniformly distributed. Consider the following transformation from (u_1, u_2) to (x, y):

$$x = \frac{u_2}{u_1}, \qquad y = u_1,$$

i.e.,

$$u_1 = y, \qquad u_2 = xy.$$

The Jacobian for the transformation is:

$$J = \begin{vmatrix} \dfrac{\partial u_1}{\partial x} & \dfrac{\partial u_1}{\partial y} \\ \dfrac{\partial u_2}{\partial x} & \dfrac{\partial u_2}{\partial y} \end{vmatrix} = \begin{vmatrix} 0 & 1 \\ y & x \end{vmatrix} = -y.$$

Therefore, the joint density of X and Y, denoted by $f_{xy}(x, y)$, is written as:

$$f_{xy}(x, y) = |J| f_{12}(y, xy) = ky,$$

for $0 \leq y \leq \sqrt{h(x)}$. The marginal density of X, denoted by $f_x(x)$, is obtained as follows:

$$f_x(x) = \int_0^{\sqrt{h(x)}} f_{xy}(x, y)\,\mathrm{d}y = \int_0^{\sqrt{h(x)}} ky\,\mathrm{d}y = k\Big[\frac{y^2}{2}\Big]_0^{\sqrt{h(x)}} = \frac{k}{2}h(x) = f(x),$$

where k is taken as: $k = 2/\int h(x)\,\mathrm{d}x$. Thus, it is shown that $f_x(\cdot)$ is equivalent to $f(\cdot)$. This result is due to Kinderman and Monahan (1977). Also see Ripley (1987), O'Hagan (1994), Fishman (1996) and Gentle (1998).

Now, we take an example of choosing the domain of (U_1, U_2). In practice, for the domain of (U_1, U_2), we may choose the rectangle which encloses the area $0 \leq U_1 \leq \sqrt{h(U_2/U_1)}$, generate a uniform point in the rectangle, and reject the point which does not satisfy $0 \leq u_1 \leq \sqrt{h(u_2/u_1)}$. That is, generate two independent uniform random draws u_1 and u_2 from $U(0, b)$ and $U(c, d)$, respectively. The rectangle is given by:

$$0 \leq u_1 \leq b, \qquad c \leq u_2 \leq d,$$

where b, c and d are given by:

$$b = \sup_x \sqrt{h(x)}, \qquad c = -\sup_x x\sqrt{h(x)}, \qquad d = \sup_x x\sqrt{h(x)},$$

because the rectangle has to enclose $0 \leq u_1 \leq \sqrt{h(u_2/u_1)}$, which is verified as follows:

$$\begin{aligned} &0 \leq u_1 \leq \sqrt{h(u_2/u_1)} \leq \sup_x \sqrt{h(x)}, \\ &-\sup_x x\sqrt{h(x)} \leq -x\sqrt{h(x)} \leq u_2 \leq x\sqrt{h(x)} \leq \sup_x x\sqrt{h(x)}. \end{aligned}$$

The second line also comes from $0 \leq u_1 \leq \sqrt{h(u_2/u_1)}$ and $x = u_2/u_1$. We can replace $c = -\sup_x x\sqrt{h(x)}$ by $c = \inf_x x\sqrt{h(x)}$, taking into account the case of $-\sup_x x\sqrt{h(x)} \leq \inf_x x\sqrt{h(x)}$. The discussion above is shown in Ripley (1987). Thus, in order to apply the ratio-of-uniforms method with the domain $\{0 \leq u_1 \leq b,\ c \leq u_2 \leq d\}$, we need to have the condition that $h(x)$ and $x^2 h(x)$ are bounded.

The algorithm for the ratio-of-uniforms method is as follows:

(i) Generate u_1 and u_2 independently from $U(0, b)$ and $U(c, d)$.

(ii) Set $x = u_2/u_1$ if $u_1^2 \leq h(u_2/u_1)$ and return to (i) otherwise.

As shown above, the x accepted in Step (ii) is taken as a random draw from $f(x) = h(x)/\int h(x)\,\mathrm{d}x$. The acceptance probability in Step (ii) is $\int h(x)\,\mathrm{d}x/(2b(d-c))$.

We have shown the rectangular domain of (U_1, U_2). It may be possible that the domain of (U_1, U_2) is a parallelogram. In Sections 3.5.1 and 3.5.2, we show two examples as applications of the ratio-of-uniforms method. Especially, in the subroutine `gammarnd7(ix,iy,alpha,rn)`, p.212, of Section 3.5.2, the parallelogram domain of (U_1, U_2) is taken as an example.

3.5.1 Normal Distribution: $N(0, 1)$

The kernel of the standard normal distribution is given by: $h(x) = \exp(-\frac{1}{2}x^2)$. In this case, b, c and d are obtained as follows:

$$\begin{aligned} b &= \sup_x \sqrt{h(x)} = 1, \\ c &= \inf_x x\sqrt{h(x)} = -\sqrt{2e^{-1}}, \\ d &= \sup_x x\sqrt{h(x)} = \sqrt{2e^{-1}}. \end{aligned}$$

Accordingly, the standard normal random number based on the ratio-of-uniforms method is represented as follows.

(i) Generate two independent uniform random draws u_1 and v_2 from $U(0,1)$ and define $u_2 = (2v_2 - 1)\sqrt{2e^{-1}}$.

(ii) Set $x = u_2/u_1$ if $u_1^2 \leq \exp(-\frac{1}{2}u_2^2/u_1^2)$, i.e., $-4u_1^2\log(u_1) \geq u_2^2$, and return to (i) otherwise.

The acceptance probability is given by:

$$\frac{\int h(x)\,\mathrm{d}x}{2b(d-c)} = \frac{\sqrt{\pi e}}{4} \approx 0.7306,$$

which is slightly smaller than the acceptance probability in the case of rejection sampling, i.e., $1/\sqrt{2e/\pi} \approx 0.7602$ as shown in Section 3.2.1, p.182.

The Fortran source code for the standard normal random number generator based on the ratio-of-uniforms method is shown in `snrnd9(ix,iy,rn)`.

`snrnd9(ix,iy,rn)`

```
1:        subroutine snrnd9(ix,iy,rn)
2:  c
3:  c  Use "snrnd9(ix,iy,rn)"
4:  c  together with "urnd(ix,iy,rn)".
5:  c
6:  c  Input:
7:  c    ix, iy:   Seeds
8:  c  Output:
9:  c    rn: Normal Random Draw N(0,1)
10: c
11:       e1=1./2.71828182845905
12:    1  call urnd(ix,iy,rn1)
13:       call urnd(ix,iy,rn2)
14:       rn2=(2.*rn2-1.)*sqrt(2.*e1)
15:       if(-4.*rn1*rn1*log(rn1).lt.rn2*rn2 ) go to 1
16:       rn=rn2/rn1
17:       return
18:       end
```

Note that `snrnd9(ix,iy,rn)` should be used together with `urnd(ix,iy,rn)` on p.83. Lines 11 – 14 represent Step (i), while Lines 15 and 16 are related to Step (ii).

Various standard normal random number generators (`snrnd1` – `snrnd9`) have been introduced until now, which will be compared later in Sections 3.7.1.

3.5.2 Gamma Distribution: $G(\alpha,\beta)$

When random variable X has a gamma distribution with parameters α and β, i.e., $X \sim G(\alpha,\beta)$, the density function of X is written as follows:

$$f(x) = \frac{1}{\beta^\alpha\Gamma(\alpha)}x^{\alpha-1}e^{-\frac{x}{\beta}},$$

for $0 < x < \infty$. When $X \sim G(\alpha, 1)$, we have $Y = \beta X \sim G(\alpha, \beta)$. Therefore, first we consider generating a random draw of $X \sim G(\alpha, 1)$.

Since we have discussed the case of $0 < \alpha \leq 1$ in Sections 3.2 – 3.4, now we consider the case of $\alpha > 1$. Using the ratio-of-uniforms method, the gamma random number generator is introduced. $h(x)$, b, c and d are set to be:

$$
\begin{aligned}
h(x) &= x^{\alpha-1}e^{-x}, \\
b &= \sup_x \sqrt{h(x)} = \left(\frac{\alpha-1}{e}\right)^{(\alpha-1)/2}, \\
c &= \inf_x x\sqrt{h(x)} = 0, \\
d &= \sup_x x\sqrt{h(x)} = \left(\frac{\alpha+1}{e}\right)^{(\alpha+1)/2}.
\end{aligned}
$$

Note that $\alpha > 1$ guarantees the existence of the supremum of $h(x)$, which implies $b > 0$. See Fishman (1996, pp.194 – 195) and Ripley (1987, pp.88 – 89). By the ratio-of-uniforms method, the gamma random number with parameter $\alpha > 1$ and $\beta = 1$ is represented as follows:

(i) Generate two independent uniform random draws u_1 and u_2 from $U(0, b)$ and $U(c, d)$, respectively.

(ii) Set $x = u_2/u_1$ if $u_1 \leq \sqrt{(u_2/u_1)^{\alpha-1}e^{-u_2/u_1}}$ and go back to (i) otherwise.

Thus, the x obtained in Steps (i) and (ii) is taken as a random draw from $G(\alpha, 1)$ for $\alpha > 1$.

Based on the above algorithm represented by Steps (i) and (ii), the Fortran 77 program for the gamma random number generator with parameters $\alpha > 1$ and $\beta = 1$ is shown in `gammarnd6(ix,iy,alpha,rn)`.

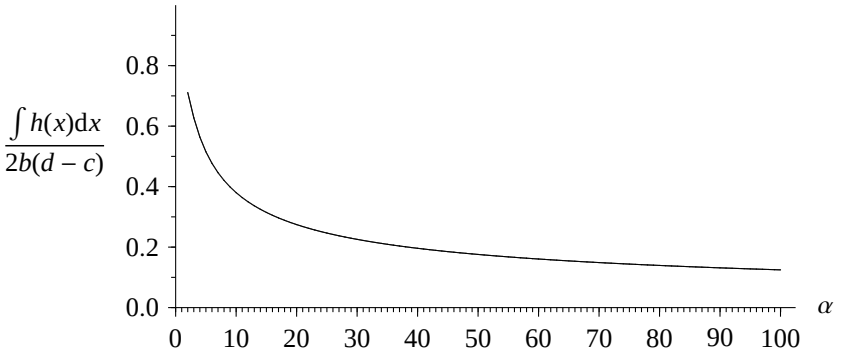
`gammarnd6(ix,iy,alpha,rn)`

```
      subroutine gammarnd6(ix,iy,alpha,rn)
c
c  Use "gammarnd6(ix,iy,alpha,rn)"
c  together with "urnd(ix,iy,rn)".
c
c  Input:
c    ix, iy:   Seeds
c    alpha:  Shape Parameter (alpha>1)
c  Output:
c    rn: Gamma Random Draw
c        with Parameters alpha and beta=1
c
       e=2.71828182845905
       b=( (alpha-1.)/e )**(0.5*alpha-0.5)
       d=( (alpha+1.)/e )**(0.5*alpha+0.5)
     1 call urnd(ix,iy,rn0)
       call urnd(ix,iy,rn1)
```

```
18:       u=rn0*b
19:       v=rn1*d
20:       rn=v/u
21:       if( 2.*log(u).gt.(alpha-1.)*log(rn)-rn ) go to 1
22:       return
23:       end
```

gammarnd6(ix,iy,alpha,rn) should be used together with urnd(ix,iy,rn) on p.83. b and d are obtained in Lines 14 and 15. Lines 16 –19 gives us two uniform random draws u and v, which correspond to u_1 and u_2. rn in Line 20 indicates a candidate of the gamma random draw. Line 21 represents Step (ii).

Figure 3.3: Acceptance Probability: Equation (3.7)



To see efficiency or inefficiency of the generator above, we compute the acceptance probability in Step (ii) as follows:

$$\frac{\int h(x)\,\mathrm{d}x}{2b(d-c)} = \frac{e^{\alpha}\Gamma(\alpha)}{2(\alpha-1)^{(\alpha-1)/2}(\alpha+1)^{(\alpha+1)/2}}, \tag{3.7}$$

which is displayed in Figure 3.3, where $\alpha = 2, 3, \cdots, 100$ is taken. It is known that the acceptance probability decreases by the order of $O(\alpha^{-1/2})$, i.e., in other words, computational time for random number generation increases by the order of $O(\alpha^{1/2})$. Therefore, as α is larger, the generator is less efficient. See Fishman (1996) and Gentle (1998). To improve inefficiency for large α, various methods have been proposed, for example, Cheng and Feast (1979, 1980), Schmeiser and Lal (1980), Sarkar (1996) and so on.

As mentioned above, the algorithm gammarnd6 takes a long time computationally by the order of $O(\alpha^{1/2})$ as shape parameter α is large. Chen and Feast (1979) suggested the algorithm which does not depend too much on shape parameter α. As α increases the acceptance region shrinks toward $u_1 = u_2$. Therefore, Chen and Feast (1979) suggested generating two uniform random draws within the parallelogram around $u_1 = u_2$, rather than the rectangle. The source code is shown in gammarnd7(ix,iy,alpha,rn).

gammarnd7(ix,iy,alpha,rn)

```
1:        subroutine gammarnd7(ix,iy,alpha,rn)
2:  c
3:  c  Use "gammarnd7(ix,iy,alpha,rn)"
4:  c  together with "urnd(ix,iy,rn)".
5:  c
6:  c  Input:
7:  c    ix, iy:   Seeds
8:  c    alpha:  Shape Parameter (alpha>1)
9:  c  Output:
10: c    rn: Gamma Random Draw
11: c        with Parameters alpha and beta=1
12: c
13:       e =2.71828182845905
14:       c0=1.857764
15:       c1=alpha-1.
16:       c2=( alpha-1./(6.*alpha) )/c1
17:       c3=2./c1
18:       c4=c3+2.
19:       c5=1./sqrt(alpha)
20:     1 call urnd(ix,iy,u1)
21:       call urnd(ix,iy,u2)
22:       if(alpha.gt.2.5) u1=u2+c5*(1.-c0*u1)
23:       if(0.ge.u1.or.u1.ge.1.) go to 1
24:       w=c2*u2/u1
25:       if(c3*u1+w+1./w.le.c4) go to 2
26:       if(c3*log(u1)-log(w)+w.ge.1.) go to 1
27:     2 rn=c1*w
28:       return
29:       end
```

See Fishman (1996, p.200) and Ripley (1987, p.90). In Line 22, we use the rectangle for $1 < \alpha \leq 2.5$ and the parallelogram for $\alpha > 2.5$ to give a fairly constant speed as α is varied. Line 25 gives us a fast acceptance to avoid evaluating the logarithm. From computational efficiency, gammarnd7(ix,iy,alpha,rn) is preferred to gammarnd6(ix,iy,alpha,rn).

Gamma Distribution: $G(\alpha,\beta)$ **for** $\alpha > 0$ **and** $\beta > 0$**:** Combining gammarnd2 on p.184 and gammarnd7 on p.212, we introduce the gamma random number generator in the case of $\alpha > 0$. In addition, utilizing $Y = \beta X \sim G(\alpha,\beta)$ when $X \sim G(\alpha,1)$, the random number generator for $G(\alpha,\beta)$ is introduced as in the source code gammarnd8(ix,iy,alpha,beta,rn).

gammarnd8(ix,iy,alpha,beta,rn)

```
1:        subroutine gammarnd8(ix,iy,alpha,beta,rn)
2:  c
3:  c  Use "gammarnd8(ix,iy,alpha,beta,rn)"
4:  c  together with "gammarnd2(ix,iy,alpha,rn)",
```

```
5: c                "gammarnd7(ix,iy,alpha,rn)"
6: c           and  "urnd(ix,iy,rn)".
7: c
8: c  Input:
9: c    ix, iy:   Seeds
10: c   alpha:  Shape Parameter
11: c   beta:   Scale Parameter
12: c  Output:
13: c   rn: Gamma Random Draw
14: c       with Parameters alpha and beta
15: c
16:        if( alpha.le.1. ) then
17:      call gammarnd2(ix,iy,alpha,rn1)
18:        else
19:      call gammarnd7(ix,iy,alpha,rn1)
20:        endif
21:      rn=beta*rn1
22:      return
23:      end
```

Note that gammarnd8(ix,iy,alpha,beta,rn) have to be utilized simultaneously with urnd(ix,iy,rn) on p.83, gammarnd2(ix,iy,alpha,rn) on p.184 and gammarnd7(ix,iy,alpha,rn) on p.212.

Lines 16 – 20 show that we use gammarnd2 for $\alpha \leq 1$ and gammarnd7 for $\alpha > 1$. In Line 21, $X \sim G(\alpha, 1)$ is transformed into $Y \sim G(\alpha, \beta)$ by $Y = \beta X$, where X and Y indicates rn1 and rn, respectively.

Chi-Square Distribution: $\chi^2(k)$: The gamma distribution with $\alpha = k/2$ and $\beta = 2$ reduces to the chi-square distribution with k degrees of freedom. As an example of the chi-square random number generator, we show chi2rnd4(ix,iy,k,rn), where gammarnd2 for $0 < \alpha \leq 1$ and gammarnd3 for $\alpha > 1$ are utilized.

chi2rnd4(ix,iy,k,rn)

```
1:       subroutine chi2rnd4(ix,iy,k,rn)
2: c
3: c  Use "chi2rnd4(ix,iy,k,rn)"
4: c  together with "gammarnd2(ix,iy,alpha,rn)",
5: c                "gammarnd3(ix,iy,alpha,rn)"
6: c           and  "urnd(ix,iy,rn)".
7: c
8: c  Input:
9: c    ix, iy:  Seeds
10: c   k:  Degree of Freedom
11: c  Output:
12: c   rn: Chi-Square Random Draw
13: c       with k Degrees of Freedom
14: c
15:      alpha=.5*float(k)
16:      beta =2.
17:        if( alpha.le.1. ) then
18:      call gammarnd2(ix,iy,alpha,rn1)
19:        else
```

```
20:         call gammarnd3(ix,iy,alpha,rn1)
21:           endif
22:         rn=beta*rn1
23:         return
24:         end
```

In the above algorithm, note that `chi2rnd4(ix,iy,k,rn)` should be utilized together with `urnd(ix,iy,rn)` on p.83, `gammarnd2(ix,iy,alpha,rn)` on p.184 and `gammarnd3(ix,iy,alpha,rn)` on p.186.

When `gammarnd3` in Line 20 of `chi2rnd4` is replaced by `gammarnd6` on p.211 and `gammarnd7` on p.212, `chi2rnd4` is taken as `chi2rnd5` and `chi2rnd6`. Now, we have shown `chi2rnd`, `chi2rnd2`, `chi2rnd3`, `chi2rnd4`, `chi2rnd5` and `chi2rnd6`, which are compared in Section 3.7.2.

3.6 Gibbs Sampling

The sampling methods introduced in Sections 3.2 – 3.4 can be applied to the cases of both univariate and multivariate distributions. The composition method discussed in Section 3.1.1 requires the nodes to approximate a density function as a weighted sum of uniform distributions. Therefore, the number of nodes drastically increases as the dimension of the random variable vector increases. That is, a density function is approximated as m^k uniform distributions in the case of k-dimensional random variable and m nodes. The ratio-of-uniforms method shown in Section 3.5 is helpful to the univariate cases. However, the Gibbs sampler in this section is the random number generation method in the multivariate cases. The Gibbs sampler shows how to generate random draws from the unconditional densities under the situation that we can generate random draws from two conditional densities.

Geman and Geman (1984), Tanner and Wong (1987), Gelfand, Hills, Racine-Poon and Smith (1990), Gelfand and Smith (1990), Carlin and Polson (1991), Zeger and Karim (1991), Casella and George (1992), Gamerman (1997) and so on developed the Gibbs sampling theory. Carlin, Polson and Stoffer (1992), Carter and Kohn (1994, 1996) and Geweke and Tanizaki (1999, 2001) applied the Gibbs sampler to the nonlinear and/or non-Gaussian state-space models. There are numerous other applications of the Gibbs sampler. The Gibbs sampling theory is concisely described as follows.

We can deal with more than two random variables, but we consider two random variables X and Y in order to make things easier. Two conditional density functions, $f_{x|y}(x|y)$ and $f_{y|x}(y|x)$, are assumed to be known, which denote the conditional distribution function of X given Y and that of Y given X, respectively. Suppose that we can easily generate random draws of X from $f_{x|y}(x|y)$ and those of Y from $f_{y|x}(y|x)$. However, consider the case where it is not easy to generate random draws from the joint density of X and Y, denoted by $f_{xy}(x,y)$. In order to have the random draws of (X,Y) from the joint density $f_{xy}(x,y)$, we take the following procedure:

(i) Take the initial value of X as x_{-M}.

(ii) Given x_{i-1}, generate a random draw of Y, i.e., y_i, from $f(y|x_{i-1})$.

(iii) Given y_i, generate a random draw of X, i.e., x_i, from $f(x|y_i)$.

(iv) Repeat the procedure for $i = -M+1, -M+2, \cdots, 1$.

From the convergence theory of the Gibbs sampler, as M goes to infinity, we can regard x_1 and y_1 as random draws from $f_{xy}(x,y)$, which is a joint density function of X and Y. M denotes the **burn-in period**, and the first M random draws, (x_i, y_i) for $i = -M+1, -M+2, \cdots, 0$, are excluded from further consideration. When we want N random draws from $f_{xy}(x,y)$, Step (iv) should be replaced by Step (iv)', which is as follows.

(iv)' Repeat the procedure for $i = -M+1, -M+2, \cdots, N$.

As in the Metropolis-Hastings algorithm, the algorithm shown in Steps (i) – (iii) and (iv)' is formulated as follows:

$$f_i(u) = \int f^*(u|v) f_{i-1}(v)\, \mathrm{d}v.$$

For convergence of the Gibbs sampler, we need to have the invariant distribution $f(u)$ which satisfies $f_i(u) = f_{i-1}(u) = f(u)$. If we have the reversibility condition shown in equation (3.5), i.e.,

$$f^*(v|u) f(u) = f^*(u|v) f(v),$$

the random draws based on the Gibbs sampler converge to those from the invariant distribution, which implies that there exists the invariant distribution $f(u)$. Therefore, in the Gibbs sampling algorithm, we have to find the transition distribution, i.e., $f^*(u|v)$. Here, we consider that both u and v are bivariate vectors. That is, $f^*(u|v)$ and $f_i(u)$ denote the bivariate distributions. x_i and y_i are generated from $f_i(u)$ through $f^*(u|v)$, given $f_{i-1}(v)$. Note that $u = (u_1, u_2) = (x_i, y_i)$ is taken while $v = (v_1, v_2) = (x_{i-1}, y_{i-1})$ is set. The transition distribution in the Gibbs sampler is taken as:

$$f^*(u|v) = f_{y|x}(u_2|u_1) f_{x|y}(u_1|v_2)$$

Thus, we can choose $f^*(u|v)$ as shown above. Then, as i goes to infinity, (x_i, y_i) tends in distribution to a random vector whose joint density is $f_{xy}(x,y)$. See, for example, Geman and Geman (1984) and Smith and Roberts (1993).

Furthermore, under the condition that there exists the invariant distribution, the basic result of the Gibbs sampler is as follows:

$$\frac{1}{N}\sum_{i=1}^{N} g(x_i, y_i) \longrightarrow \mathrm{E}(g(x,y)) = \iint g(x,y) f_{xy}(x,y)\, \mathrm{d}x\, \mathrm{d}y, \quad \text{as } N \longrightarrow \infty,$$

where $g(\cdot,\cdot)$ is a function.

The Gibbs sampler is a powerful tool in a Bayesian framework. Based on the conditional densities, we can generate random draws from the joint density. Some applications are shown in Chapter 4.

Remark 1: We have considered the bivariate case, but it is easily extended to the multivariate cases. That is, it is possible to take multi-dimensional vectors for x and y. Taking an example, as for the tri-variate random vector (X, Y, Z), if we generate the ith random draws from $f_{x|yz}(x|y_{i-1}, z_{i-1})$, $f_{y|xz}(y|x_i, z_{i-1})$ and $f_{z|xy}(z|x_i, y_i)$, sequentially, we can obtain the random draws from $f_{xyz}(x, y, z)$.

Remark 2: Let X, Y and Z be the random variables. Take an example of the case where X is highly correlated with Y. If we generate random draws from $f_{x|yz}(x|y, z)$, $f_{y|xz}(y|x, z)$ and $f_{z|xy}(z|x, y)$, it is known that convergence of the Gibbs sampler is slow. In this case, without separating X and Y, random number generation from $f(x, y|z)$ and $f(z|x, y)$ yields better random draws from the joint density $f(x, y, z)$.

3.7 Comparison of Sampling Methods

We have introduced several kinds of random number generators. Taking the standard normal distribution, the chi-squared distribution and the binomial distribution, the random number generators are compared with respect to computational CPU time and precision of the estimates of $\mathrm{E}(X^k)$ for $k = 1, 2, \cdots$. Dual Pentium III 1GHz CPU, Microsoft Windows 2000 Professional Operating System and Open Watcom FORTRAN 77/32 Optimizing Compiler (Version 1.0) are utilized for all the computations in this section. Note that WATCOM Fortran 77 and C Compilers can be downloaded from `http://www.openwatcom.org/`.

3.7.1 Standard Normal Random Number Generators

In this section, we examine several standard normal random number generators introduced in previous sections with respect to precision of the random draws and computational time. Each source code is characterized as follows. `snrnd` on p.85 is given by the Box-Muller transformation, `snrnd2` on p.87 is derived from the modified Box-Muller transformation, `snrnd3` on p.88 is based on the central limit theorem, `snrnd4` on p.126 is the inverse transform method obtained from the approximated percent points, `snrnd5_2` on p.175 approximates the standard normal density as the sum of uniform distributions, `snrnd6` on p.183 utilizes rejection sampling, `snrnd7` on p.191 is based on importance resampling, `snrnd8` on p.203 is from the Metropolis-Hastings algorithm and `snrnd9` on p.210 is derived from the ratio-of-uniforms method.

The results are in Table 3.1. For all the random number generators, 10^7 standard normal random draws are generated. As for `snrnd5_2` in Section 3.1.1, the random draws have to be generated within the interval $(x_{(0)}, x_{(m)})$, where $x_{(0)} = -5$ and $x_{(m)} = 5$ are taken. Let x_i be the ith random draw generated from each source code. Then, we compute $(1/n)\sum_{i=1}^{n} x_i^k$, which corresponds to the estimate of the kth moment, $\mathrm{E}(X^k)$, where $n = 10^7$ and $k = 1, 2, 3, 4$ are taken. We choose $m = 20, 50, 100, 200$ for the number of regions in `snrnd5_2` and $m = 200, 500, 1000, 2000$ for the number

Table 3.1: Precision of the Estimates and Computational Time

	m	E(X)	E(X^2)	E(X^3)	E(X^4)	CPU Time (Seconds)
snrnd		0.000	1.000	−0.001	3.000	5.68
snrnd2		0.000	1.000	0.000	3.001	6.17
snrnd3		0.000	1.000	0.001	2.896	20.65
snrnd4		0.000	1.000	−0.001	3.003	4.49
snrnd5_2	20	0.000	1.084	0.000	3.515	4.50
snrnd5_2	50	0.000	1.014	0.000	3.082	5.78
snrnd5_2	100	0.000	1.004	0.000	3.022	7.87
snrnd5_2	200	0.000	1.001	0.000	3.007	12.11
snrnd6		0.001	1.000	0.002	2.999	8.49
snrnd7	200	0.000	1.065	−0.001	3.237	13.61
snrnd7	500	0.000	1.026	−0.001	3.073	29.40
snrnd7	1000	0.000	1.004	0.001	2.975	55.77
snrnd7	2000	0.000	0.985	0.000	2.988	108.00
snrnd8		0.000	1.001	−0.001	3.004	7.06
snrnd9		0.000	1.001	−0.002	3.006	6.28
—		0.000	1.000	0.000	3.000	

of candidate random draws in snrnd7. The burn-in period is $M = 1000$ in snrnd8. In Table 3.1, the bottom line implies the theoretical values of E(X^k) for $k = 1, 2, 3, 4$ from the standard normal distribution. All the values should close to those in the bottom line. CPU Time indicates computation time (seconds) of generating 10^7 random draws and computing E(X^k) for $k = 1, 2, 3, 4$.

For snrnd5_2 and snrnd7, as m (i.e., the number of regions or the number of the random draws generated from the sampling density) increases, each estimate of E(X^k) approaches the true value shown in the bottom line. All the values except for the case of $m = 20$ in snrnd5_2 and the case of $m = 200$ in snrnd7 are very close to the true values shown in the bottom line. However, as it is expected, snrnd5_2 and snrnd7 take a lot of time computationally when m is large. snrnd4 is less computational than any other source codes, but it is derived from the approximated percent points based on snperpt(p,x) on p.90. Therefore, it might be expected that snrnd4 gives us less precise random draws than any other source codes (remember that snrnd5_2 and snrnd7 generates random draws from the true distribution as m is large). However, from Table 3.1, snrnd4 shows the best performance with respect to both precision and computational time. This implies that the approximation of the percent points obtained by snperpt(p,x) is really good. For the other source codes, it is shown that snrnd, snrnd2 and snrnd9 are practically acceptable, but snrnd is slightly less computational than snrnd2 and snrnd9. As a result, snrnd might be recommended in practice (snrnd4 is excluded from consideration even though it shows the best performance, because it utilized the approximated percent point). Originally, snrnd2 and snrnd9 was proposed to improve snrnd from computational point of view, be-

cause snrnd includes computation of the sine or cosine in the Box-Muller algorithm. A long time ago, computation of the sine or cosine took a lot of time. However, in recent progress of personal computers, it can be seen from the above results that we do not have to care about computation of the sine or cosine.

3.7.2 Chi-Square Random Number Generators

We have introduced several chi-square random number generators, which are given by: chi2rnd (p.102), chi2rnd2 (p.103) and chi2rnd3 (p.104) in Section 2.2.8, and chi2rnd4 (p.214), chi2rnd5 (p.215) and chi2rnd6 (p.215) in Section 3.5.2. Each chi-square random number generator with k degrees of freedom has the following features. chi2rnd is based on the sum of k squared standard normal random draws, chi2rnd2 is from the sum of $[k/2]$ exponential random draws and $k - [k/2]2$ squared standard normal random draw, chi2rnd3 is based on the central limit theorem, and chi2rnd4, chi2rnd5 and chi2rnd6 utilize the gamma random draw with parameters $\alpha = k/2$ and $\beta = 2$. Remember that the gamma distribution with parameters $\alpha = k/2$ and $\beta = 2$ reduces to the chi-square distribution with k degrees of freedom. Note that chi2rnd4 utilizes gammarnd2 (p.184) and gammarnd3 (p.186), chi2rnd5 is based on gammarnd2 (p.184) and gammarnd6 (p.211), and chi2rnd6 uses gammarnd2 (p.184) and gammarnd7 (p.212). Now we compare the six algorithms with respect to precision of the random draws and computational time. 10^7 random draws are generated and k =2, 5, 10, 20, 50, 100 are taken. Mean, variance, skewness and kurtosis are computed for comparison, which are denoted by $\mathrm{E}(X) = \mu$, $\mathrm{V}(X) = \mu_2 = \sigma^2$, μ_3/σ^3 and μ_4/σ^4, where $\mu_j = \mathrm{E}(X-\mu)^j$. "—" represents the theoretical values of mean, variance, skewness and kurtosis of the chi-square random variable with k degrees of freedom.

The results are in Table 3.2. Theoretically, μ and σ^2 of the chi-square random variable with k degrees of freedom are k and $2k$, respectively. For both μ and σ^2, all the random number generators perform well, because μ and σ^2 estimated by chi2rnd, chi2rnd2, chi2rnd3, chi2rnd4, chi2rnd5 and chi2rnd6 are very close to the true values shown in "—". Furthermore, for skewness and kurtosis, chi2rnd, chi2rnd2, chi2rnd4, chi2rnd5 and chi2rnd6 are very close to "—" for all k =2, 5, 10, 20, 50, 100, but chi2rnd3 is very different from "—". From the central limit theorem (p.33), under the condition that mean and variance exist, any arithmetic average is approximately normally distributed as the sample size is large. As k is large the theoretical skewness and kurtosis go to 0 and 3, respectively. We can see from the table that the normal approximation becomes practical when k is much larger, i.e., k less than 100 is too small to approximate the chi-square distribution, even though chi2rnd3 is the least computational of the six algorithms. As discussed in Section 3.5.2, computational time of gammarnd6 (p.211) increases in the order of $O(\alpha^{1/2})$, where $\alpha/2 = k$. Accordingly, chi2rnd5 takes a long time in the order of $O(k^{1/2})$. For chi2rnd4 and chi2rnd6, however, computational time does not depend on α. When we compare these two source codes, chi2rnd6 is less computational than chi2rnd4 for all α. As a

Table 3.2: Precision of the Estimates and Computational Time

k		μ	σ^2	μ_3/σ^3 (Skewness)	μ_4/σ^4 (Kurtosis)	CPU Time (Seconds)
2	chi2rnd	2.000	4.000	2.001	9.023	11.40
	chi2rnd2	2.001	4.005	2.003	9.025	2.97
	chi2rnd3	2.000	4.000	0.000	3.000	5.81
	chi2rnd4	2.000	4.001	1.997	8.987	10.41
	chi2rnd5	2.000	4.001	1.997	8.987	10.44
	chi2rnd6	2.000	4.001	1.997	8.987	10.42
	—	2.000	4.000	2.000	9.000	
5	chi2rnd	4.999	9.998	1.266	5.405	28.00
	chi2rnd2	5.000	10.006	1.267	5.416	10.97
	chi2rnd3	5.000	10.001	0.000	3.000	6.22
	chi2rnd4	4.999	10.008	1.268	5.408	11.80
	chi2rnd5	5.002	10.012	1.264	5.400	14.43
	chi2rnd6	5.002	10.013	1.264	5.399	9.25
	—	5.000	10.000	1.265	5.400	
10	chi2rnd	10.000	19.995	0.892	4.191	55.82
	chi2rnd2	10.001	20.017	0.897	4.212	13.12
	chi2rnd3	9.999	20.002	0.000	3.000	6.22
	chi2rnd4	9.999	20.000	0.895	4.200	11.94
	chi2rnd5	10.001	20.007	0.893	4.192	12.64
	chi2rnd6	9.998	19.994	0.894	4.202	8.86
	—	10.000	20.000	0.894	4.200	
20	chi2rnd	20.000	39.981	0.631	3.597	111.11
	chi2rnd2	20.000	40.013	0.634	3.604	25.89
	chi2rnd3	19.999	40.003	0.000	3.000	6.22
	chi2rnd4	19.999	40.009	0.634	3.603	11.73
	chi2rnd5	20.002	39.982	0.631	3.592	19.91
	chi2rnd6	19.996	39.970	0.633	3.603	9.20
	—	20.000	40.000	0.632	3.600	
50	chi2rnd	49.999	99.999	0.401	3.243	277.00
	chi2rnd2	49.998	99.960	0.400	3.237	63.92
	chi2rnd3	49.999	100.009	0.000	3.000	6.22
	chi2rnd4	50.000	100.032	0.402	3.243	11.63
	chi2rnd5	50.002	100.073	0.400	3.237	22.95
	chi2rnd6	49.995	100.002	0.402	3.247	9.22
	—	50.000	100.000	0.400	3.240	
100	chi2rnd	100.002	199.982	0.282	3.117	553.51
	chi2rnd2	99.998	199.901	0.281	3.121	127.32
	chi2rnd3	99.998	200.017	0.000	3.000	6.22
	chi2rnd4	99.999	200.012	0.285	3.122	11.56
	chi2rnd5	99.999	200.153	0.283	3.119	34.72
	chi2rnd6	99.992	200.051	0.284	3.123	9.28
	—	100.000	200.000	0.283	3.120	

$\mu_j = \mathrm{E}(X-\mu)^j$ for $j = 2, 3, 4$, where $\mu = \mathrm{E}(X)$ and $\mu_2 = \sigma^2 = \mathrm{V}(X)$

result, it might be concluded that `chi2rnd6` is the best algorithm from computational cost and accuracy.

3.7.3 Binomial Random Number Generators

Given $n = 5, 10, 15, 30, 50, 100$ and $p = 0.1$, we generate 10^7 binomial random draws, using the subroutines `birnd` on p.144, `birnd2` on p.145 and `birnd3` on p.146, shown in Section 2.4.5. $\mathrm{E}(X) = \mu$, $\mathrm{V}(X) = \mu_2 = \sigma^2$, μ_3/σ^3 (skewness) and μ_4/σ^4 (kurtosis) are evaluated for $n = 5, 10, 15, 30, 50, 100$. `birnd` utilizes the fact that the sum of n Bernoulli random variables reduces to a binomial random variable. `birnd2` uses the inverse transform method. In `birnd3`, the central limit theorem is applied. Therefore, it might be expected that `birnd3` gives us the approximated binomial random draws, while `birnd` and `birnd2` yield the exact ones.

Table 3.3: Precision of the Estimates and Computational Time

n		μ	σ^2	μ_3/σ^3 (Skewness)	μ_4/σ^4 (Kurtosis)	CPU Time (Seconds)
5	`birnd`	0.500	0.450	1.193	4.023	8.19
	`birnd2`	0.500	0.450	1.192	4.017	4.65
	`birnd3`	0.500	0.534	0.000	2.953	8.47
	—	0.500	0.450	1.193	4.022	
10	`birnd`	1.000	0.900	0.842	3.508	16.11
	`birnd2`	1.000	0.900	0.843	3.508	5.21
	`birnd3`	1.000	0.983	0.000	2.992	8.43
	—	1.000	0.900	0.843	3.511	
15	`birnd`	1.500	1.350	0.689	3.342	23.91
	`birnd2`	1.500	1.350	0.689	3.339	5.58
	`birnd3`	1.500	1.434	−0.001	2.995	8.43
	—	1.500	1.350	0.689	3.341	
30	`birnd`	3.000	2.701	0.486	3.170	47.30
	`birnd2`	3.000	2.701	0.486	3.168	6.72
	`birnd3`	3.000	2.784	0.000	2.999	8.42
	—	3.000	2.700	0.487	3.170	
50	`birnd`	5.000	4.499	0.376	3.101	78.49
	`birnd2`	5.000	4.501	0.377	3.100	8.09
	`birnd3`	5.000	4.584	−0.001	2.999	8.41
	—	5.000	4.500	0.377	3.102	
100	`birnd`	10.001	9.001	0.266	3.048	156.45
	`birnd2`	9.999	9.003	0.266	3.050	11.44
	`birnd3`	10.000	9.084	0.000	3.000	8.40
	—	10.000	9.000	0.267	3.051	

$\mu_j = \mathrm{E}(X - \mu)^j$ for $j = 2, 3, 4$, where $\mu = \mathrm{E}(X)$ and $\mu_2 = \sigma^2 = \mathrm{V}(X)$

The results are in Table 3.3, where "—" indicates the mean, variance, skewness and kurtosis obtained theoretically. Therefore, each value in `birnd`, `birnd2` and

birnd3 should be close to "—". When n is small, birnd3 is not close to "—". In the case where n is large, however, birnd3 is very close to each other. birnd and birnd2 are almost the same as "—" for any n. Computationally, both birnd and birnd2 take a lot of time when n increases, but birnd2 has advantages over birnd. Therefore, birnd2 might be recommended from precision and computational time.

3.7.4 Rejection Sampling, Importance Resampling and the Metropolis-Hastings Algorithm

We compare rejection sampling, importance resampling and the Metropolis-Hastings algorithm from precision of the estimated moments and CPU time. All the three sampling methods utilize the sampling density and they are useful when it is not easy to generate random draws directly from the target density. When the sampling density is too far from the target density, it is known that rejection sampling takes a lot of time computationally while importance resampling and the Metropolis-Hastings algorithm yields unrealistic random draws. In this section, therefore, we investigate how the sampling density depends on the three sampling methods.

For simplicity of discussion, consider the case where both the target and sampling densities are normal. That is, the target density $f(x)$ is given by $N(0,1)$ and the sampling density $f_*(x)$ is $N(\mu_*, \sigma_*^2)$. $\mu_* = 0$, 1, 2, 3 and $\sigma_* = 0.5$, 1.0, 1.5, 2.0, 3.0, 4.0 are taken. For each of the cases, the first three moments $\mathrm{E}(X^j)$, $j = 1, 2, 3$, are estimated, generating 10^7 random draws. For importance resampling, $n = 10^4$ is taken, which is the number of candidate random draws. The Metropolis-Hastings algorithm takes $M = 1000$ as the burn-in period and the initial value is $x_{-M} = \mu_*$. As for the Metropolis-Hastings algorithm, note that is the independence chain is taken for $f_*(x)$ because of $f_*(x|z) = f_*(x)$. The source code used in this section will be shown later.

In Tables 3.4 and 3.5, RS, IR and MH denotes rejection sampling, importance resampling and the Metropolis-Hastings algorithm, respectively. In each table, "—" in RS implies the case where rejection sampling cannot be applied because the supremum of $q(x)$, $\sup_x q(x)$, does not exist. As for MH in the case of $\mathrm{E}(X) = 0$, the values in the parentheses represent the acceptance rate (percent) in the Metropolis-Hastings algorithm. Table 3.5 represents CPU times (seconds) to obtain each cell in the case of $\mathrm{E}(X) = 0$ of Table 3.4.

The results obtained from each table are as follows. $\mathrm{E}(X)$ in Table 3.4 should be close to zero because we have $\mathrm{E}(X) = 0$ from $X \sim N(0,1)$. When $\mu_* = 0.0$, all of RS, IR and MH are very close to zero and show a good performance. When $\mu_* = 1$, 2, 3, for $\sigma_* = 1.5$, 2.0, 3.0, 4.0, all of RS, IR and MH perform well, but IR and MH in the case of $\sigma_* = 0.5$, 1.0 have the case where the estimated mean is too different from zero. For IR and MH, we can see that given σ_* the estimated mean is far from the true mean as μ_* is far from mean of the target density. Also, it might be concluded that given μ_* the estimated mean approaches the true value as σ_* is large.

$\mathrm{E}(X^2)$ in Table 3.4 should be close to one because we have $\mathrm{E}(X^2) = \mathrm{V}(X) = 1$

Table 3.4: Comparison of Three Sampling Methods

	$\mu_* \backslash \sigma_*$		0.5	1.0	1.5	2.0	3.0	4.0
E(X) = 0	0	RS	—	—	0.000	0.000	0.000	0.000
		IR	0.060	0.005	0.000	0.005	0.014	0.014
		MH	−0.004	0.000	0.000	0.000	0.000	0.000
			(59.25)	(100.00)	(74.89)	(59.04)	(40.99)	(31.21)
	1	RS	—	—	0.000	0.000	0.000	0.000
		IR	0.327	0.032	0.025	0.016	0.011	0.011
		MH	0.137	0.000	0.001	0.000	0.000	0.000
			(36.28)	(47.98)	(55.75)	(51.19)	(38.68)	(30.23)
	2	RS	—	—	0.000	0.000	0.000	0.000
		IR	0.851	0.080	0.031	0.030	0.003	0.005
		MH	0.317	0.005	0.001	0.001	0.000	0.001
			(8.79)	(15.78)	(26.71)	(33.78)	(32.50)	(27.47)
	3	RS	—	—	0.000	0.000	0.000	−0.001
		IR	1.590	0.337	0.009	0.029	0.021	−0.007
		MH	0.936	0.073	−0.002	0.000	0.001	−0.001
			(1.68)	(3.53)	(9.60)	(17.47)	(24.31)	(23.40)
E(X^2) = 1	0	RS	—	—	1.000	1.000	1.000	0.999
		IR	0.822	0.972	0.969	0.978	0.994	1.003
		MH	0.958	1.000	1.000	1.000	1.001	1.001
	1	RS	—	—	1.000	1.000	1.000	1.000
		IR	0.719	0.980	0.983	0.993	1.010	1.004
		MH	0.803	1.002	0.999	0.999	1.001	1.002
	2	RS	—	—	1.000	1.000	1.001	1.001
		IR	1.076	0.892	1.014	0.984	1.000	1.012
		MH	0.677	0.992	1.001	0.999	1.001	1.002
	3	RS	—	—	1.000	1.000	1.000	1.000
		IR	2.716	0.696	1.013	1.025	0.969	1.002
		MH	1.165	0.892	1.005	1.001	0.999	0.999
E(X^3) = 0	0	RS	—	—	0.000	0.000	0.000	−0.001
		IR	0.217	0.034	−0.003	−0.018	0.018	0.036
		MH	−0.027	0.001	0.001	−0.001	−0.002	−0.004
	1	RS	—	—	0.002	−0.001	0.000	0.001
		IR	0.916	0.092	0.059	0.058	0.027	0.032
		MH	0.577	−0.003	0.003	0.000	0.002	−0.001
	2	RS	—	—	−0.001	0.002	0.001	0.001
		IR	1.732	0.434	0.052	0.075	0.040	0.001
		MH	0.920	0.035	0.003	0.004	0.004	0.004
	3	RS	—	—	0.000	0.001	0.001	−0.001
		IR	5.030	0.956	0.094	0.043	0.068	0.020
		MH	1.835	0.348	−0.002	0.003	0.001	−0.001

Table 3.5: Comparison of Three Sampling Methods: CPU Time (Seconds)

μ_* \ σ_*		0.5	1.0	1.5	2.0	3.0	4.0
0	RS	—	—	15.96	20.50	30.69	39.62
	IR	431.89	431.40	431.53	432.58	435.37	437.16
	MH	9.70	9.24	9.75	9.74	9.82	9.77
1	RS	—	—	23.51	24.09	32.77	41.03
	IR	433.22	427.96	426.41	426.36	427.80	430.39
	MH	9.73	9.54	9.81	9.75	9.83	9.76
2	RS	—	—	74.08	38.75	39.18	45.18
	IR	435.90	432.23	425.06	423.78	421.46	422.35
	MH	9.71	9.52	9.83	9.77	9.82	9.77
3	RS	—	—	535.55	87.00	52.91	53.09
	IR	437.32	439.31	429.97	424.45	422.91	418.38
	MH	9.72	9.48	9.79	9.75	9.81	9.76

from $X \sim N(0,1)$. The cases of $\sigma_* = 1.5, 2.0, 3.0, 4.0$ and the cases of $\mu_* = 0, 1$ and $\sigma_* = 1.0$ are very close to one, but the other cases are different from one. These are the same results as the case of $\mathrm{E}(X)$. $\mathrm{E}(X^3)$ in Table 3.4 should be close to zero because $\mathrm{E}(X^3)$ represents skewness. For skewness, we obtain the similar results, i.e., the cases of $\sigma_* = 1.5, 2.0, 3.0, 4.0$ and the cases of $\mu_* = 0, 1$ and $\sigma_* = 0.5, 1.0$ perform well for all of RS, IR and MH.

In the case where we compare RS, IR and MH, RS shows the best performance of the three, and IR and MH is quite good when σ_* is relatively large. We can conclude that IR is slightly worse than RS and MH.

As for the acceptance rates of MH in $\mathrm{E}(X) = 0$, from the table a higher acceptance rate generally shows a better performance. The high acceptance rate implies high randomness of the generated random draws. In Section 3.7.5, the sampling density in the MH algorithm will be discussed in detail. From Table 3.4, for variance of the sampling density, both too small variance and too large variance give us the relatively low acceptance rate, which result is consistent with the discussion in Chib and Greenberg (1995).

For each random number generator and each case of μ_* and σ_*, the computation time which is required to generate 10^7 random draws is shown in Table 3.5. MH has the advantage over RS and IR from computational point of view. IR takes a lot of time because all the acceptance probabilities have to be computed in advance (see Section 3.3 for IR). That is, 10^4 candidate random draws are generated from the sampling density $f_*(x)$ and therefore 10^4 acceptance probabilities have to be computed. For MH and IR, computational CPU time does not depend on μ_* and σ_*. However, for RS, given σ_* computational time increases as μ_* is large. In other words, as the sampling density is far from the target density the number of rejections increases. When σ_*

increases given μ_*, the acceptance rate does not necessarily increase. However, from the table a large σ_* is better than a small σ_* in general. Accordingly, as for RS, under the condition that mean of $f(x)$ is unknown, we can conclude that relatively large variance of $f_*(x)$ should be taken.

Finally, the results obtained from Tables 3.4 and 3.5 are summarized as follows.

(1) For IR and MH, depending on choice of the sampling density $f_*(x)$, we have the cases where the estimates of mean, variance and skewness are biased. For RS, we can always obtain the unbiased estimates without depending on choice of the sampling density.

(2) In order to avoid the biased estimates, it is safe for IR and MH to choose the sampling density with relatively large variance. Furthermore, for RS we should take the sampling density with relatively large variance to reduce computational burden. But, note that too large variance leads to an increase in computational disadvantages.

(3) MH is the least computational sampling method of the three. For IR, all the acceptance probabilities have to be computed in advance and therefore IR takes a lot of time to generate random draws. In the case of RS, the amount of computation increases as $f_*(x)$ is far from $f(x)$.

(4) For the sampling density in MH, it is known that both too large variance and too small variance yield slow convergence of the obtained random draws. The slow convergence implies that a great amount of random draws have to be generated from the sampling density for evaluation of the expectations such as E(X) and V(X). Therefore, choice of the sampling density has to be careful,

Thus, RS gives us the best estimates in the sense of unbiasedness, but RS sometimes has the case where the supremum of $q(x)$ does not exist and in this case it is impossible to implement RS. As the sampling method which can be applied to any case, MH might be preferred to IR and RS in a sense of less risk, judging from Tables 3.4 and 3.5. However, we should keep in mind that MH also has the problem which choice of the sampling density is very important. This will be discussed in Section 3.7.5.

Source Code: Now, we briefly discuss the source code used in this section, which is shown in `Source Code for Section 3.7.4`, where Lines 1 – 11 represents the rejection sampling method (Section 3.2), Lines 12 – 37 indicates the importance resampling procedure (Section 3.3) and Lines 38 – 50 shows the Metropolis-Hastings algorithm (Section 3.4). Note that this source code requires both `snrnd(ix,iy,rn)` and `urnd(ix,iy,rn)`.

In Lines 2, 26 and 39 of the Fortran source code, `ave` and `ser` represent μ_* and σ_*, which are the parameters in the sampling density $f_*(x)$. Both `ix` and `iy` indicate the seeds for the uniform random number generator `urnd(ix,iy,rn)`. `rn` denotes the random draw generated from the target density $f(x)$.

As for `resample(ix,iy,x,prob,m,rn)` in Line 13, all the `m` random draws generated from the sampling density and the corresponding probability weights have to be stored in `x(i)` and `prob(i)`. Note that `m`, `x(i)` and `prob(i)` in Lines 12 – 37 denote n, x_i^* and $\omega(x_i^*)$ shown in Section 3.3. Both `x(i)` and `prob(i)` are obtained from `weight(ix,iy,ave,ser,m,x,prob)` in Lines 25 – 37. In Line 39 of the subroutine `metropolis(ix,iy,ave,ser,rn,accept)`, `accept` represents the number of acceptances, which is utilized to obtain the acceptance probabilities shown in the parentheses of Table 3.4.

Source Code for Section 3.7.4

```
1: C ========================================================
2:       subroutine rejection(ix,iy,ave,ser,rn)
3:       q(z)=-.5*z*z -( -log(ser)-.5*((z-ave)**2)/(ser**2) )
4:       q_max=q( ave/(1.-ser*ser) )
5:     1 call snrnd(ix,iy,rn0)
6:       rn=ave+ser*rn0
7:       w=exp( q(rn)-q_max )
8:       call urnd(ix,iy,ru)
9:       if(ru.gt.w) go to 1
10:      return
11:      end
12: C ========================================================
13:      subroutine resample(ix,iy,x,prob,m,rn)
14:      dimension x(0:100001),prob(0:100001)
15:      call urnd(ix,iy,rn1)
16:        do 1 j=1,m
17:         if(prob(j-1).le.rn1.and.rn1.lt.prob(j)) then
18:      i=j
19:      go to 2
20:         endif
21:    1   continue
22:    2 rn=x(i)
23:      return
24:      end
25: C ----------------------------------------
26:      subroutine weight(ix,iy,ave,ser,m,x,prob)
27:      dimension x(0:100001),prob(0:100001)
28:      q(z)=-.5*z*z -( -log(ser)-.5*((z-ave)**2)/(ser**2) )
29:      prob(0)=0.0
30:        do 1 i=1,m
31:      call snrnd(ix,iy,rn)
32:      x(i)=ave+ser*rn
33:    1 prob(i)=prob(i-1)+exp( q(x(i)) )
34:        do 2 i=1,m
35:    2 prob(i)=prob(i)/prob(m)
36:      return
37:      end
38: C ========================================================
39:      subroutine metropolis(ix,iy,ave,ser,rn,accept)
40:      q(z)=-.5*z*z -( -log(ser)-.5*((z-ave)**2)/(ser**2) )
41:      call snrnd(ix,iy,rn0)
42:      rn1=ave+ser*rn0
43:      w=exp( q(rn1)-q(rn) )
44:      call urnd(ix,iy,ru)
45:         if(ru.le.w) then
46:      rn=rn1
```

```
47:         accept=accept+1.0
48:            endif
49:         return
50:         end
```

`q(z)` defined in Lines 3, 28 and 40 represents the subtraction of the constant term from the logarithm of the ratio of the target density and the sampling density, i.e., $\log q(z)$, where $q(\cdot)$ is shown in equation (3.2). By taking the logarithm, the possibility of underflow or overflow is reduced. For rejection sampling, `q_max` in Line 4 gives us the maximum value of $q(z)$ with respect to z, which shows that $q(z)$ takes the maximum value when $z = \mu_*/(1-\sigma_*^2)$. See p.181 for this discussion.

The candidate random draws are generated from the sampling density $f_*(x)$ in Lines 5 and 6 for rejection sampling, Lines 31 and 32 for importance resampling and Lines 41 and 42 for the Metropolis-Hastings algorithm. In Lines 7, 33 – 35 and 43, the acceptance probabilities are computed for each sampling method.

3.7.5 Sampling Density in the Metropolis-Hastings Algorithm

We have discussed the Metropolis-Hastings algorithm in Section 3.4. In this section, we examine Sampling Densities I – III, which are introduced in Section 3.4. Sampling Density I is given by $f_*(x_i|x_{i-1}) = f_*(x_i)$, typically $N(\mu_*, \sigma_*^2)$, where the sampling density does not depend on the $(i-1)$th random draw, x_{i-1}, which is called the independence chain. Sampling Density II represents, for example, $N(x_{i-1}, \sigma_*^2)$, which is called the random walk chain. Sampling Density III utilized the second order Taylor series expansion of the target density, which is called the Taylored chain, where the target density is approximated by the normal distribution, the exponential distribution or the uniform distribution. See p.199 for Sampling Densities I – III. In this section, taking some examples we examine how the three sampling densities depend on precision of the random draws.

3.7.5.1 Bimodal Distribution

The target density $f(x)$ is given by:

$$f(x) = \frac{1}{2}N(\mu_1, \sigma_1^2) + \frac{1}{2}N(\mu_2, \sigma_2^2),$$

which consists of two normal densities, where $(\mu_1, \sigma_1^2) = (1, 1^2)$ and $(\mu_2, \sigma_2^2) = (-1, 0.5^2)$ are taken, which is described in Figure 3.4. Theoretically, we have the following moments: $\mathrm{E}(X) = 0$, $\mathrm{E}(X^2) = 1.625$, $\mathrm{E}(X^3) = 1.125$ and $\mathrm{E}(X^4) = 6.344$.

It is very easy to generate random draws from the bimodal density shown above. However, in this section, using the Metropolis-Hastings algorithm we consider generating random draws from the above bimodal distribution, because we want to compare

Figure 3.4: Bimodal Normal Distribution: $f(x)$

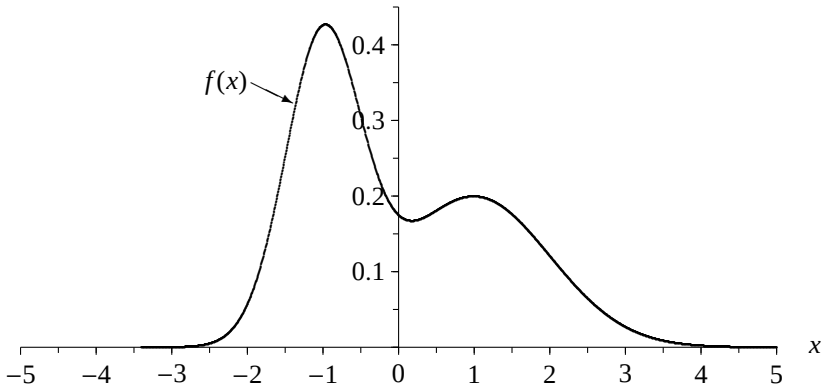
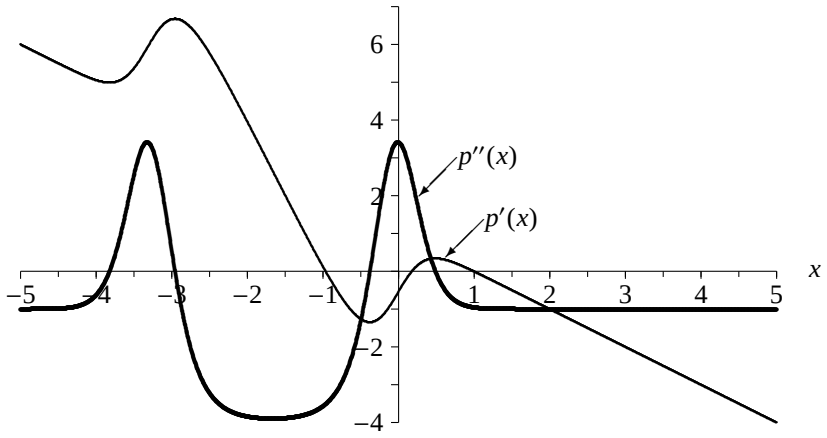


Figure 3.5: Bimodal Normal Distribution: $p'(x)$ and $p''(x)$, where $p(x) = \log f(x)$



Sampling Densities I – III with respect to precision of the generated random draws. The following three types of the sampling densities are examined:

$$f_*(x|x_{i-1}) = N(\mu_*, \sigma_*^2), \tag{3.8}$$

$$f_*(x|x_{i-1}) = N(x_{i-1}, \sigma_*^2), \tag{3.9}$$

$$\begin{aligned} f_*(x|x_{i-1}) = {} & f_{*1}(x|x_{i-1})I_1(x_{i-1}) + f_{*2}(x|x_{i-1})I_2(x_{i-1}) \\ & + f_{*3}(x|x_{i-1})I_3(x_{i-1}) + f_{*4}(x|x_{i-1})I_4(x_{i-1}), \end{aligned} \tag{3.10}$$

where $f_{*i}(x|x_{i-1})$, $i = 1, 2, 3, 4$, in equation (3.10) are given by:

$$f_{*1}(x|x_{i-1}) = N\Big(x_{i-1} - \frac{p'(x_{i-1})}{p''(x_{i-1})}, \frac{-1}{p''(x_{i-1})}\Big),$$

$$f_{*2}(x|x_{i-1}) = \lambda \exp\Big(-\lambda(x - (x^+ - d))\Big), \qquad \text{for } x^+ - d < x,$$

$$f_{*3}(x|x_{i-1}) = \lambda \exp\Big(-\lambda((x^+ + d) - x)\Big), \qquad \text{for } x < x^+ + d,$$

Table 3.6: Sampling Density III: Classification into the Four Cases

x	$-\infty$	$\sim$	-3.821	$\sim$	-2.956	$\sim$	-0.964	$\sim$	-0.378
$p'(x)$	+	+	+	+	+	+	0	−	−
$p''(x)$	−	−	0	+	0	−	−	−	0
Case	1	1	3	3	3	1	1	1	2
x	$\sim$	0.179	$\sim$	0.488	$\sim$	0.994	$\sim$	∞	
$p'(x)$	−	0	+	+	+	0	−	−	
$p''(x)$	+	+	+	0	−	−	−	−	
Case	2	4	3	3	1	1	1	1	

$$f_{*4}(x|x_{i-1}) = \frac{1}{x^{++} - x^{+}}, \qquad \text{for } x^{+} < x < x^{++}.$$

Note that λ, x^{+}, x^{++} and d are discussed in Section 3.4.1, p.201. Moreover, $I_i(\cdot)$, $i = 1, 2, 3, 4$, denote the following indicator functions:

$$I_1(x_{i-1}) = \begin{cases} 1, & \text{if } p''(x_{i-1}) < -\epsilon, \\ 0, & \text{otherwise,} \end{cases}$$

$$I_2(x_{i-1}) = \begin{cases} 1, & \text{if } p''(x_{i-1}) \geq -\epsilon \text{ and } p'(x_{i-1}) < 0, \\ 0, & \text{otherwise,} \end{cases}$$

$$I_3(x_{i-1}) = \begin{cases} 1, & \text{if } p''(x_{i-1}) \geq -\epsilon \text{ and } p'(x_{i-1}) > 0, \\ 0, & \text{otherwise,} \end{cases}$$

$$I_4(x_{i-1}) = \begin{cases} 1, & \text{if } p''(x_{i-1}) \geq -\epsilon \text{ and } p'(x_{i-1}) = 0, \\ 0, & \text{otherwise,} \end{cases}$$

where ϵ in $I_i(\cdot)$, $i = 1, 2, 3, 4$, is taken as 0.0, 0.2, 0.3 or 0.4 in this section. Equations (3.8) – (3.10) correspond to Sampling Densities I – III, respectively, which are discussed in Section 3.4 (see p.199). For Sampling Density III, the first and second derivatives of the logarithm of $f(x)$, i.e., $p'(x)$ and $p''(x)$ for $p(x) = \log f(x)$, have to be required. Therefore, $p'(x)$ and $p''(x)$ are computed and displayed in Figure 3.5. Each case of 1 – 4 is summarized in Table 3.6, where $\sim$ should be read as "between" and + and − indicate positive and negative values. Based on the positive or negative signs of $p'(x)$ and $p''(x)$, the sampling density $f_*(x)$ is classified into the four cases, i.e., Cases 1 – 4.

In Tables 3.7 and 3.8, we take $\mu_* = -3, -2, -1, 0, 1, 2, 3$ and σ_* =0.5, 1.0, 1.5, 2.0, 3.0, 4.0 for Sampling Density I and σ_* =0.5, 1.0, 1.5, 2.0, 3.0, 4.0 for Sampling Density II. Sampling Density III does not have the hyper-parameter such as μ_* and σ_*, but it depends on ϵ. It will be shown in Table 3.9 that choice of ϵ is not too crucial to precision of the generated random draws. For each case and each sampling density, the moments $\mathrm{E}(X^k)$, $k = 1, 2, 3, 4$, are estimated, generating N random draws, where $N = 10^7$ is taken. Note that the estimates are given by $(1/N)\sum_{i=1}^{N} x_i^k$, $k = 1, 2, 3, 4$.

The estimated moments should be close to the theoretical values, which are given by $\mathrm{E}(X) = 0$, $\mathrm{E}(X^2) = 1.625$, $\mathrm{E}(X^3) = 1.125$ and $\mathrm{E}(X^4) = 6.344$, as mentioned above. In Tables 3.7 and 3.9, AP represents the acceptance probability (%) corresponding to each case of μ_* and σ_* and each sampling density, where the acceptance probability represents the ratio of the number of the cases where x_i is updated for $i = 1, 2, \cdots, N$ (i.e., $x_i \neq x_{i-1}$) relative to the number of random draws (i.e., $N = 10^7$). For Sampling Density I, when σ_* is small and μ_* is far from $\mathrm{E}(X) = 0$, the estimated moments are very different from the true moments. Since we have $\mathrm{E}(X) = 0$ for $f(x)$, $\mathrm{E}(X^2)$ is equivalent to variance of X. When σ_* is larger than $\sqrt{\mathrm{E}(X^2)}$, all the moments are very close to the true values. Therefore, for Sampling Density I we can conclude from Table 3.7 that variance of the sampling density should be larger than that of the target density. As for Sampling Density II, all the estimated moments are close to the true moments for all σ_* =0.5, 1.0, 1.5, 2.0, 3.0, 4.0. Similarly, Sampling Density III also shows a good performance, because all the estimated moments are close to the true values (see Estimate in Table 3.9). Both Sampling Densities I and II depend on the hyper-parameters μ_* and σ_*, while Sampling Density III does not. Sampling Density III depends on ϵ, but the cases of $\epsilon = 0.2, 0.3, 0.4$ are very similar to each other while the case of $\epsilon = 0.0$ gives us the slightly overestimated moments. Accordingly, ϵ does not influence precision of the random draws unless ϵ is close to zero. For Sampling Densities I and II, we need to choose the hyper-parameters, which problem is a crucial criticism. Thus, we can see that it is easy to use Sampling Density III in practice. Note that computational time in each cell of Table 3.7 is approximately given by 18 seconds for Sampling Density I, 16 seconds for Sampling Density II and 160 seconds for Sampling Density III. Thus, Sampling Density III takes a lot of time computationally.

In Table 3.7, AP indicates the acceptance probability corresponding to each case of μ_* and σ_* and each sampling density. In Figure 3.6, the acceptance probabilities obtained from Sampling Density I are displayed by contour lines, where the acceptance probabilities are computed in the cases of $\mu_* = -5.0, -4.9, \cdots, 5.0$ and $\sigma_* = 0.1, 0.2, \cdots, 6.0$. The acceptance probabilities are greater than 70% in the area around $\mu = -0.4, \cdots, 0.4$ and $\sigma_* = 1.1, \cdots, 1.6$. Mean and variance of the sampling density should be close to those of the target density to obtain large acceptance probability. Remember that mean and variance of the target density function are given by $\mathrm{E}(X) = 0$ and $\mathrm{E}(X^2) = 1.625$.

In Table 3.7, we examine whether the estimated moments are close to the theoretical values, based on the 10^7 random draws. When we compare two consistent estimates, the estimate with small variance is clearly preferred to that with large variance. Consider dividing the N random draws into N_2 partitions, where each partition consists of N_1 random draws. Therefore, clearly we have $N_1 \times N_2 = N = 10^7$. Let $g(x_i)$ be a function of the ith random draw, x_i, which function is taken as $g(x) = x^k$,

Table 3.7: Estimates of the Moments

Sampling Density	μ_* \ σ_*		0.5	1.0	1.5	2.0	3.0	4.0
I	−3.0	E(X)	−1.031	−0.167	0.002	0.001	0.000	−0.001
		E(X^2)	1.258	1.289	1.635	1.625	1.626	1.623
		E(X^3)	−1.746	−0.029	1.173	1.124	1.125	1.118
		E(X^4)	2.675	3.191	6.568	6.339	6.345	6.333
		AP	0.48	4.18	11.62	19.65	28.76	27.52
	−2.0	E(X)	−0.782	0.006	0.000	0.000	0.000	−0.001
		E(X^2)	1.013	1.626	1.626	1.625	1.625	1.626
		E(X^3)	−1.385	1.093	1.126	1.122	1.124	1.123
		E(X^4)	2.137	6.075	6.353	6.331	6.346	6.343
		AP	12.45	20.21	30.30	37.44	37.87	31.90
	−1.0	E(X)	−0.335	0.002	0.000	0.000	−0.001	−0.001
		E(X^2)	1.070	1.634	1.626	1.627	1.626	1.626
		E(X^3)	−0.684	1.151	1.124	1.128	1.123	1.125
		E(X^4)	2.089	6.414	6.342	6.360	6.348	6.351
		AP	61.56	54.09	56.91	56.89	44.14	34.69
	0.0	E(X)	−0.055	0.001	0.000	0.000	0.000	0.000
		E(X^2)	1.464	1.626	1.625	1.626	1.625	1.624
		E(X^3)	0.503	1.135	1.124	1.128	1.124	1.125
		E(X^4)	4.296	6.373	6.342	6.356	6.346	6.348
		AP	33.39	66.98	72.79	62.91	45.81	35.40
	1.0	E(X)	0.095	0.002	0.001	0.000	0.000	0.000
		E(X^2)	1.516	1.627	1.624	1.624	1.625	1.626
		E(X^3)	1.421	1.127	1.127	1.123	1.125	1.125
		E(X^4)	5.563	6.363	6.342	6.332	6.348	6.350
		AP	31.84	53.74	56.84	53.47	42.48	33.93
	2.0	E(X)	0.657	0.012	0.002	0.002	0.001	0.001
		E(X^2)	1.520	1.621	1.626	1.627	1.625	1.627
		E(X^3)	3.027	1.167	1.131	1.131	1.128	1.132
		E(X^4)	7.442	6.343	6.346	6.354	6.351	6.360
		AP	25.21	24.55	31.53	36.25	35.37	30.56
	3.0	E(X)	1.317	0.030	0.001	0.001	0.001	0.000
		E(X^2)	2.311	1.618	1.632	1.624	1.626	1.626
		E(X^3)	4.901	1.255	1.129	1.130	1.129	1.126
		E(X^4)	11.834	6.308	6.384	6.349	6.351	6.352
		AP	8.79	8.09	13.94	20.30	26.47	25.90
II	—	E(X)	−0.001	0.001	0.001	0.000	−0.001	0.000
		E(X^2)	1.627	1.625	1.626	1.625	1.625	1.626
		E(X^3)	1.126	1.125	1.124	1.123	1.123	1.125
		E(X^4)	6.363	6.342	6.340	6.339	6.340	6.349
		AP	83.14	71.11	61.72	53.76	41.58	33.29

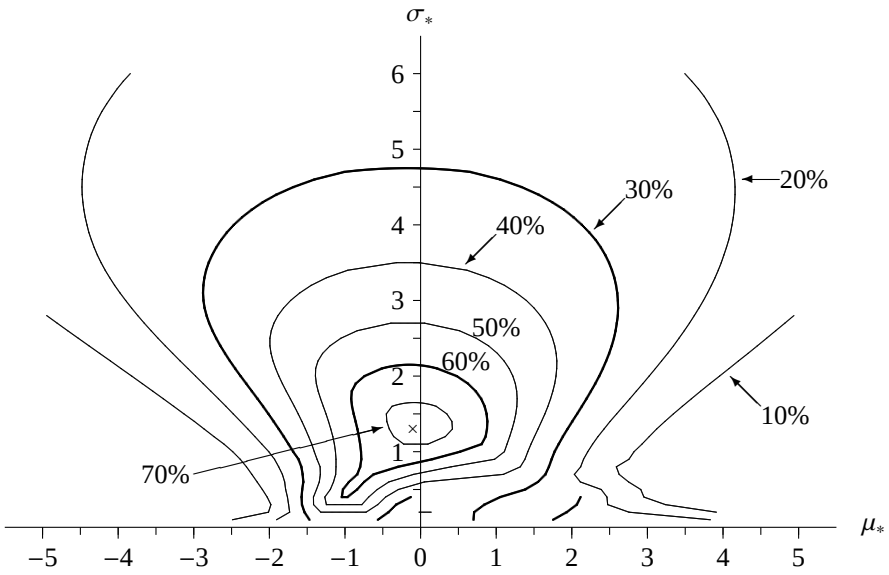
Table 3.8: Standard Errors of the Estimated Moments

Sampling Density	μ_* \ σ_*		0.5	1.0	1.5	2.0	3.0	4.0
I	−3.0	E(X)	0.386	0.798	0.189	0.055	0.032	0.031
		E(X^2)	0.818	0.832	0.411	0.084	0.045	0.045
		E(X^3)	1.479	2.584	1.558	0.278	0.141	0.139
		E(X^4)	2.640	4.333	5.341	0.763	0.408	0.421
	−2.0	E(X)	0.373	0.589	0.064	0.032	0.026	0.028
		E(X^2)	0.318	1.260	0.115	0.044	0.036	0.042
		E(X^3)	0.509	4.376	0.407	0.144	0.117	0.127
		E(X^4)	0.756	12.493	1.242	0.389	0.338	0.382
	−1.0	E(X)	0.558	0.143	0.027	0.020	0.023	0.027
		E(X^2)	0.277	0.388	0.042	0.029	0.034	0.039
		E(X^3)	1.078	1.527	0.143	0.092	0.101	0.120
		E(X^4)	0.628	5.380	0.398	0.264	0.311	0.354
	0.0	E(X)	0.478	0.029	0.017	0.020	0.024	0.028
		E(X^2)	0.845	0.065	0.022	0.025	0.031	0.040
		E(X^3)	2.610	0.255	0.072	0.083	0.098	0.122
		E(X^4)	5.814	0.950	0.206	0.232	0.283	0.359
	1.0	E(X)	0.708	0.048	0.027	0.024	0.026	0.029
		E(X^2)	0.474	0.036	0.027	0.028	0.034	0.039
		E(X^3)	2.397	0.157	0.098	0.092	0.107	0.118
		E(X^4)	4.967	0.256	0.235	0.246	0.304	0.350
	2.0	E(X)	0.779	0.202	0.051	0.036	0.031	0.033
		E(X^2)	0.902	0.151	0.043	0.035	0.038	0.041
		E(X^3)	2.252	0.692	0.167	0.124	0.118	0.132
		E(X^4)	5.598	0.789	0.324	0.297	0.330	0.368
	3.0	E(X)	0.581	0.809	0.122	0.056	0.038	0.036
		E(X^2)	1.541	0.499	0.089	0.054	0.044	0.046
		E(X^3)	3.735	2.511	0.382	0.192	0.143	0.144
		E(X^4)	9.482	2.677	0.592	0.419	0.372	0.412
II	—	E(X)	0.082	0.046	0.036	0.032	0.030	0.032
		E(X^2)	0.089	0.055	0.047	0.043	0.040	0.043
		E(X^3)	0.356	0.213	0.176	0.160	0.138	0.143
		E(X^4)	0.887	0.567	0.478	0.446	0.396	0.404

Table 3.9: Sampling Density III (Estimated Moments and Standard Errors)

	Estimate				Standard Error			
ϵ	0.0	0.2	0.3	0.4	0.0	0.2	0.3	0.4
E(X)	0.000	−0.001	−0.001	−0.001	0.064	0.064	0.062	0.062
E(X^2)	1.630	1.626	1.626	1.625	0.043	0.043	0.041	0.041
E(X^3)	1.130	1.126	1.127	1.125	0.196	0.198	0.193	0.191
E(X^4)	6.371	6.354	6.352	6.350	0.345	0.349	0.341	0.335
AP	74.88	75.20	75.28	75.23				

Figure 3.6: Acceptance Probabilities in Sampling Density I: Contour Lines



× denotes the maximum acceptance probability (74%),
where $(\mu_*, \sigma_*) = (-0.1, 1.3)$

$k = 1, 2, 3, 4$, in Table 3.7. Define $\overline{g_i(x)}$ and $\overline{g(x)}$ as follows:

$$\overline{g_i(x)} = \frac{1}{N_1} \sum_{j=1}^{N_1} g(x_{(i-1)N_1+j}),$$

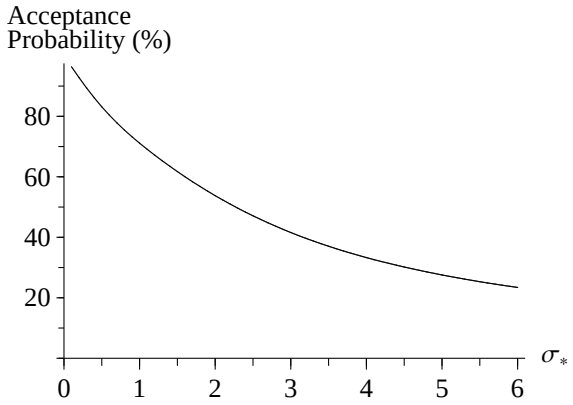
$$\overline{g(x)} = \frac{1}{N_1 \times N_2} \sum_{j=1}^{N_1 \times N_2} g(x_j) = \frac{1}{N_2} \sum_{i=1}^{N_2} \Big(\frac{1}{N_1} \sum_{j=1}^{N_1} g(x_{(i-1)N_1+j}) \Big) = \frac{1}{N_2} \sum_{i=1}^{N_2} \overline{g_i(x)},$$

for $N = N_1 \times N_2$. Thus, $\overline{g_i(x)}$ represents the arithmetic average of $g(x_{(i-1)N_1+j})$, $j = 1, 2, \cdots, N_1$ and $\overline{g(x)}$ denotes the arithmetic average of $g(x_i)$, $i = 1, 2, \cdots, N_1 \times N_2$. Then, the standard error of $\overline{g(x)}$ is given by:

$$\mathrm{Se}(\overline{g(x)}) = \sqrt{\frac{1}{N_2} \sum_{i=1}^{N_2} \Big(\overline{g_i(x)} - \overline{g(x)} \Big)^2}.$$

For each cell in Table 3.7, $\mathrm{Se}(\overline{g(x)})$ is computed and presented in Table 3.8, where $N_1 = 10^4$, $N_2 = 10^3$ and $g(x) = x^k$, $k = 1, 2, 3, 4$, are taken. Note that Table 3.7 indicates the $\overline{g(x)}$ criterion (i.e., unbiasedness), while Table 3.8 represents the $\mathrm{Se}(\overline{g(x)})$ criterion (i.e., minimum variance). For Sampling Density I, the cases of $(\mu_*, \sigma_*) = (-1, 2), (0, 1.5), (0, 2), (0, 3), (1, 1.5), (1, 2)$ are preferred to the other cases from the

Figure 3.7: Acceptance Probabilities in Sampling Density II



standard error criterion. At $(\mu_*, \sigma_*) = (-0.1, 1.3)$, which is the point that gives us the maximum acceptance probability (i.e., 74%), the standard errors of the estimated first, second, third and fourth moments are given by 0.018 0.027 0.089 and 0.258, respectively. The standard errors of the estimated moments at $(\mu_*, \sigma_*) = (-0.1, 1.3)$ are larger than those at $(\mu_*, \sigma_*) = (0, 1.5)$. Note tha the standard errors of the estimated moments at $(\mu_*, \sigma_*) = (0, 1.5)$ are 0.017, 0.022, 0.072 and 0.206 in Table 3.8. Therefore, for the independence chain (Sampling Density I) we can conclude that the sampling density which gives us the maximum acceptance probability is not necessarily the best choice. For Sampling Density II, the acceptance probability decreases as σ_* increases (see Figure 3.7). The case of $\sigma_* = 3$ gives us the smallest standard errors of the estimated moments (see Table 3.8). However, as in Table 3.8, the standard errors in the case of $\sigma_* = 3$ for Sampling Density II are slightly larger than those in the case of $\mu_* = -1$, 0, 1 and $\sigma_* = 1.5$, 2, 3, 4 for Sampling Density I. Therefore, if the appropriate sampling density is chosen, Sampling Density I is better than Sampling Density II. Moreover, from Standard Error (the right-hand side) in Table 3.9, Sampling Density III also shows a good performance. However, Sampling Densities I and II with the appropriate hyper-parameters are better than Sampling Density III from the standard error criterion.

The problem of Sampling Densities I and II is that we have to choose the hyper-parameters. If we take the hyper-parameters which give us the least standard errors of the estimated moments, it extremely takes a lot of time because we have to compute the standard errors for all the possible combinations of the hyper-parameters. Choice of ϵ does not influence precision of the generated random draws, provided that ϵ is not close to zero (see Table 3.9). In the case of $\epsilon = 0.0$ in Table 3.9, all the moments are overestimated, but in the other cases (i.e., $\epsilon = 0.2, 0.3, 0.4$) they are properly estimated. Thus, since Sampling Density III does not have the hyper-parameter such as μ_* and σ_*, it can avoid the criticism of choosing the hyper-parameters included in Sampling Densities I and II.

Source Code: Now, we discuss the source code utilized in this section is shown in Source Code for Section 3.7.5. Sampling Density I is given by Lines 1 – 13, Sampling Density II is in Lines 14 – 25, and Sampling Density III is related to Lines 26 – 123. To reduce the possibility of underflow or overflow, we take the logarithm of the sampling and target densities in Lines 3, 38, 47, 57, 65, 88, 95, 101, 106 and 125 – 126. ave and var correspond to μ_* and σ_*^2 in the sampling density. accept indicates the number of acceptances, where one is added to accept in Lines 10, 22 and 74 whenever rn is accepted. x represents the candidate random draw generated from the sampling density (Lines 5, 17, 37, 47, 56 and 64). In Lines 6, 18 and 71, w gives us the acceptance probability $\omega(x_{i-1}, x^*)$, where x_{i-1} and x^* correspond to rn and x, respectively. As for density3, fcn0 and fcn1 represents $\log f_*(x^*|x_{i-1})$ and $\log f_*(x_{i-1}|x^*)$, and mode(isign,delta,x0,x1) in Lines 111 – 122 is used to obtain the nearest mode, where isign takes +1 or −1 and determines the direction for searching (i.e., positive or negative direction).

Source Code for Section 3.7.5

```
1: c =======================================================
2:       subroutine density1(ix,iy,ave,var,rn,accept)
3:       f0(x,ave,var)=-.5*log(var)-.5*(x-ave)*(x-ave)/var
4:       call snrnd(ix,iy,rn1)
5:       x=ave+sqrt(var)*rn1
6:       w=exp( f(x)-f0(x,ave,var)-f(rn)+f0(rn,ave,var) )
7:       call urnd(ix,iy,ru)
8:          if( ru.le.w ) then
9:       rn=x
10:      accept=accept+1.0
11:         endif
12:      return
13:      end
14: c =======================================================
15:      subroutine density2(ix,iy,var,rn,accept)
16:      call snrnd(ix,iy,rn1)
17:      x=rn+sqrt(var)*rn1
18:      w=exp( f(x)-f(rn) )
19:      call urnd(ix,iy,ru)
20:         if( ru.le.w ) then
21:      rn=x
22:      accept=accept+1.0
23:         endif
24:      return
25:      end
26: c =======================================================
27:      subroutine density3(ix,iy,rn,epsilon,accept)
28:      pi=3.14159265358979
29:      delta  =1./100.
30:      df =( f(rn+delta)-f(rn-delta) )/(2.*delta)
31:      ddf=( f(rn+delta)-2.*f(rn)+f(rn-delta) )/(delta**2)
32:        if( ddf.lt.-epsilon ) then
33: c * Case 1 ***
34:      ave=rn-df/ddf
35:      var=-1./ddf
36:      call snrnd(ix,iy,rn1)
37:      x=ave+sqrt(var)*rn1
38:      fcn1=-.5*log(2.*pi*var)-.5*(x-ave)*(x-ave)/var
```

```
      call log_pdf(rn,x,fcn0,epsilon,delta)
        else
           if( df.lt.0 ) then
c * Case 2 ***
      call mode(-1,delta,rn,x1)
      slope=abs( (f(x1)-f(rn))/(x1-rn) )
      d=1./slope
      call urnd(ix,iy,rn1)
      x=-log(rn1)/slope+(x1-d)
      fcn1=log(slope)-slope*( x-(x1-d) )
      call log_pdf(rn,x,fcn0,epsilon,delta)
           else if( df.gt.0 ) then
c * Case 3 ***
      call mode(+1,delta,rn,x2)
      slope=abs( (f(x2)-f(rn))/(x2-rn) )
      d=1./slope
      call urnd(ix,iy,rn1)
      x=(x2+d)+log(rn1)/slope
      fcn1=log(slope)-slope*( (x2+d)-x )
      call log_pdf(rn,x,fcn0,epsilon,delta)
           else
c * Case 4 ***
      call mode(-1,delta,rn,x1)
      call mode(+1,delta,rn,x2)
      call urnd(ix,iy,rn1)
      x=x1+rn1*(x2-x1)
      fcn1=-log(x2-x1)
      call log_pdf(rn,x,fcn0,epsilon,delta)
           endif
        endif
c
      w=exp( f(x)-fcn1-f(rn)+fcn0 )
      call urnd(ix,iy,ru)
         if( ru.le.w ) then
      rn=x
      accept=accept+1.0
         endif
      return
      end
c ---------------------------------------
      subroutine log_pdf(x,y,fcn,epsilon,delta)
      pi=3.14159265358979
      df =( f(y+delta)-f(y-delta) )/(2.*delta)
      ddf=( f(y+delta)-2.*f(y)+f(y-delta) )/(delta**2)
      fcn=-999999.9
        if( ddf.lt.-epsilon ) then
c * Case 1 ***
      ave= y-df/ddf
      var=-1./ddf
      fcn=-.5*log(2.*pi*var)-.5*(x-ave)*(x-ave)/var
        else
           if( df.lt.0 ) then
c * Case 2 ***
      call mode(-1,delta,y,x1)
      slope=abs( (f(x1)-f(y))/(x1-y) )
      d=1./slope
      if( x.gt.x1-d ) fcn=log(slope)-slope*( x-(x1-d) )
           else if( df.gt.0 ) then
c * Case 3 ***
      call mode(+1,delta,y,x2)
      slope=abs( (f(x2)-f(y))/(x2-y) )
      d=1./slope
      if( x.lt.x2+d ) fcn=log(slope)-slope*( (x2+d)-x )
```

```
102:            else
103: c * Case 4 ***
104:       call mode(-1,delta,rn,x1)
105:       call mode(+1,delta,rn,x2)
106:       if( x1.lt.x.and.x.lt.x2 ) fcn=-log(x2-x1)
107:            endif
108:         endif
109:       return
110:       end
111: c ---------------------------------------
112:       subroutine mode(isign,delta,x0,x1)
113:       f0=f(x0)
114:       x1=x0
115:         do 1 i=1,10*nint(1./delta)
116:       x1=x1+isign*delta
117:       f1=f(x1)
118:       if(f1.lt.f0) go to 2
119:     1 f0=f1
120:     2 x1=x1-isign*delta
121:       return
122:       end
123: c =======================================================
124:       function f(x)
125:       f(x)=log( .5*exp(-.5*(x-1.)*(x-1.) )
126:      &         +.5*exp(-.5*(x+1.)*(x+1.)/0.25)/sqrt(.25) )
127:       return
128:       end
```

In Line 29, `epsilon` implies ϵ, which is taken 0.2 in the source code above. In Table 3.9, $\epsilon = 0.0, 0.2, 0.3, 0.4$ is investigated.

In the next section, the other target densities are examined to see which sampling density works well. In the above source code, when we want to examine the other distribution we may rewrite Lines 125 and 126, where `f(x)` corresponds to the log-kernel $p(x) = \log f(x)$. Therefore, the rest of all the lines can be utilized without any modification.

3.7.5.2 Other Distributions

In this section, using the other distributions we examine Sampling Densities I – III, where we take the following four target distributions, i.e., $t(5)$, logistic, LaPlace and Gumbel distributions. We compare the estimates of the first and second moments with their true values.

- ***t*(5) Distribution:** $f(x) = \dfrac{1}{\sqrt{2\pi}} \dfrac{\Gamma(\frac{k+1}{2})}{\Gamma(\frac{k}{2})} \Big(1 + \dfrac{x^2}{k}\Big)^{-(k+1)/2}$,
  where $\mathrm{E}(X) = 0$ and $\mathrm{E}(X^2) = k/(k-2)$. $k = 5$ is taken in the simulation studies.

- **Logistic Distribution:** $f(x) = \dfrac{e^x}{(1+e^x)^2}$,
  where the first and second moments are $\mathrm{E}(X) = 0$ and $\mathrm{E}(X^2) = \pi^2/3$.

- **LaPlace Distribution:** $f(x) = \frac{1}{2}\exp(-|x|)$,
  where the first two moments are given by $\mathrm{E}(X) = 0$ and $\mathrm{E}(X^2) = 2$.

- **Gumbel Distribution:** $f(x) = e^{-(x-\alpha)}\exp(-e^{-(x-\alpha)})$,
  where mean and variance are $\mathrm{E}(X) = \alpha+\gamma$ and $\mathrm{V}(X) = \pi^2/6$. $\gamma = 0.5772156599$ is known as Euler's constant. Since we take $\alpha = -\gamma$ in this section, the first and second moments are $\mathrm{E}(X) = 0$ and $\mathrm{E}(X^2) = \pi^2/6$.

Results and Discussion: It is very easy to generate random draws from the densities shown above without utilizing the MH algorithm. However, in this section, to see whether the sampling densities work well or not, using the MH algorithm we consider generating random draws from the above target distributions.

The results are in Tables 3.10 – 3.15. In order to obtain the ith random draw (i.e., x_i), we take $N(\mu_*, \sigma_*^2)$ for Sampling Density I (Tables 3.10 – 3.13), where $\mu_* = -3$, -2, -1, 0, 1, 2, 3 and $\sigma_* =$1, 2, 3, 4 are taken, and $N(x_{i-1}, \sigma_*^2)$ for Sampling Density II (Table 3.14), where $\sigma_* =$1, 2, 3, 4, 5, 6, 7 is chosen. Sampling Density III (Table 3.15) does not have the hyper-parameters such as μ_* and σ_*. For each sampling density, the two moments $\mathrm{E}(X^k)$, $k = 1, 2$, are estimated, generating N random draws, where $N = 10^7$ is taken. Note that the estimates of the two moments are given by $(1/N)\sum_{i=1}^{N} x_i^k$, $k = 1, 2$.

The estimated moments should be close to the theoretical values, which are given by $\mathrm{E}(X) = 0$ and $\mathrm{E}(X^2) = 5/3$ in the $t(5)$ distribution, $\mathrm{E}(X) = 0$ and $\mathrm{E}(X^2) = 3.290$ in the logistic distribution, $\mathrm{E}(X) = 0$ and $\mathrm{E}(X^2) = 2$ in the LaPlace distribution and $\mathrm{E}(X) = 0$ and $\mathrm{E}(X^2) = 1.645$ in the Gumbel distribution. Thus, for all the four target distributions taken in this section, $\mathrm{E}(X) = 0$ is chosen. In each table, the values in the parentheses indicate $\mathrm{Se}(\overline{g(x)})$, where $g(x) = x$ in the case of $\mathrm{E}(X)$ and $g(x) = x^2$ for $\mathrm{E}(X^2)$. AP denotes the acceptance probability (%) corresponding to each sampling density. Note that AP is obtained by the ratio of the number of the cases where x_i is updated for $i = 1, 2, \cdots, N$ (i.e., $x_i \neq x_{i-1}$) relative to the number of random draws (i.e., $N = 10^7$).

According to the $\overline{g(x)}$ criterion, we can observe the following results. For Sampling Density I (Tables 3.10 – 3.13), when σ_* is small and μ_* is far from zero, the estimated moments are very different from the true values. Since we have $\mathrm{E}(X) = 0$, $\mathrm{E}(X^2)$ is equivalent to variance of X. When σ_* is larger than $\sqrt{\mathrm{E}(X^2)}$, the first and second moments are very close to the true values. For example, we take Table 3.10 (the $t(5)$ distribution). When $\sigma_* = 3, 4$, the estimates of the first moment are -0.002 to 0.001 and those of the second one are from 1.655 to 1.668. Thus, the cases of $\sigma_* = 3, 4$ perform much better than those of $\sigma_* = 1, 2$. In Tables 3.11 – 3.13, the similar results are obtained. Therefore, for Sampling Density I we can conclude from Tables 3.10 – 3.13 that variance of the sampling density should be larger than that of the target density. As for Sampling Density II (Table 3.14), the estimated moments are close to the true moments for all $\sigma_* =$1, 2, 3, 4, 5, 6, 7. The LaPlace distribution

does not utilize Case 1 in Sampling Density III. Therefore, we do not pay attention to LaPlace in Table 3.15, because LaPlace does not depend on ϵ. For Sampling Density III (Table 3.15), except for the cases of $\epsilon = 0.0$, all the estimates of the two moments are very close to the true values. Thus, by the $\overline{g(x)}$ criterion, Sampling Densities II and III are quite good. Moreover, Sampling Density I also performs good, provided that σ_*^2 is larger than variance of the target density and μ_* is close to mean of the target density.

Next, we focus on the Se($\overline{g(x)}$) criterion. In Tables 3.10 – 3.15, the values in the parentheses indicate the corresponding standard errors. In Table 3.10, the cases of $\mu_* = -1$, 0, 1 and $\sigma_* = 3$, 4 show a quite good performance, and especially the case of $(\mu_*, \sigma_*) = (0, 3)$ gives us the smallest standard errors of the estimated moments. In Table 3.11, the cases of $\mu_* = -1$, 0, 1 and $\sigma_* = 3$ are quite good, and especially the case of $(\mu_*, \sigma_*) = (0, 3)$ indicates the best performance. In Table 3.12, the cases of $\mu_* = -1$, 0, 1 and $\sigma_* = 3$ give us good results, and especially the case of $(\mu_*, \sigma_*) = (0, 3)$ is better than any other cases. In Table 3.13, the cases of $\mu_* = -2$, -1, 0, 1, 2 and $\sigma_* = 3$, 4 are quite good, and especially $(\mu_*, \sigma_*) = (0, 3)$ is the best. Thus, for Sampling Density I, the sampling density should have the same mean as the target density and the sampling density has to be distributed more broadly than the target density. However, we find from Tables 3.10 – 3.13 that too large variance of the sampling density yields poor estimates of the moments. In Table 3.14 (Sampling Density II), the smallest standard errors of the estimated moments are given by $\sigma_* = 4, 5, 6$ for the $t(5)$ distribution, $\sigma_* = 4, 5$ for the logistic distribution, $\sigma_* = 3$ for the LaPlace distribution and $\sigma_* = 3, 4$ for the Gumbel distribution. In Table 3.15 (Sampling Density III), when ϵ is small, i.e., when $\epsilon = 0.0$, 0.05, all the standard errors except for LaPlace are quite large. when $\epsilon = 0.2$, 0.3, 0.4, all the values are close to each other. As a result, we can see from the tables that Sampling Density I shows the best performance when μ_* and σ_* are properly chosen, compared with Sampling Densities II and III. For example, the smallest standard errors in Table 3.10 (Sampling Density I) are 0.019 for E(X) and 0.070 for E(X^2), those of $t(5)$ in Table 3.14 (Sampling Density II) are given by 0.030 for E(X) and 0.124 for E(X^2), and the smallest standard errors of $t(5)$ in Table 3.15 (Sampling Density III) are 0.037 for E(X) and 0.173 for E(X^2). Thus, Sampling Density I gives us the best results.

We compare AP and Se($\overline{g(x)}$). In Table 3.10, at $(\mu_*, \sigma_*) = (0, 1)$, which is the point that gives us the maximum acceptance probability (i.e., 92.80%), the standard errors of the estimated first and second moments are given by 0.157 and 0.747. The standard errors of the estimated moments at $(\mu_*, \sigma_*) = (0, 1)$ are larger than those at $(\mu_*, \sigma_*) = (0, 3)$, because the standard errors of the estimated moments at $(\mu_*, \sigma_*) = (0, 3)$ are given by 0.019 and 0.070. Therefore, judging from the standard error criterion, the case of $(\mu_*, \sigma_*) = (0, 3)$ is preferred to that of $(\mu_*, \sigma_*) = (0, 1)$. In Table 3.11, the largest AP is achieved at $(\mu_*, \sigma_*) = (0, 2)$ and the minimum standard errors are given by $(\mu_*, \sigma_*) = (0, 3)$. Moreover, in Table 3.12, the maximum AP is $(\mu_*, \sigma_*) = (0, 1)$ and the smallest standard errors are around $(\mu_*, \sigma_*) = (0, 3)$. In Table 3.13, the largest AP is achieved at $(\mu_*, \sigma_*) = (0, 1)$ and the minimum standard errors

Table 3.10: $t(5)$ Distribution (Sampling Density I)

μ_* \ σ_*		1	2	3	4
−3	E(X)	−0.122 (0.608)	−0.003 (0.063)	−0.001 (0.028)	−0.002 (0.026)
	E(X^2)	1.274 (0.587)	1.643 (0.316)	1.661 (0.093)	1.665 (0.088)
	AP	6.42	20.13	28.09	27.26
−2	E(X)	−0.008 (0.438)	−0.001 (0.035)	−0.001 (0.023)	0.000 (0.024)
	E(X^2)	1.503 (0.994)	1.643 (0.203)	1.662 (0.107)	1.668 (0.081)
	AP	19.33	37.83	37.34	31.93
−1	E(X)	−0.016 (0.149)	−0.001 (0.027)	0.000 (0.019)	−0.002 (0.022)
	E(X^2)	1.503 (0.524)	1.646 (0.186)	1.662 (0.079)	1.667 (0.082)
	AP	50.13	57.07	44.37	35.13
0	E(X)	−0.003 (0.157)	0.001 (0.026)	0.000 (0.019)	0.000 (0.022)
	E(X^2)	1.541 (0.747)	1.650 (0.213)	1.661 (0.070)	1.666 (0.075)
	AP	92.80	65.95	47.02	36.28
1	E(X)	0.014 (0.195)	0.001 (0.027)	0.000 (0.019)	0.000 (0.022)
	E(X^2)	1.510 (0.699)	1.647 (0.205)	1.658 (0.078)	1.667 (0.076)
	AP	50.08	57.04	44.38	35.15
2	E(X)	0.060 (0.313)	0.003 (0.035)	0.001 (0.023)	0.001 (0.024)
	E(X^2)	1.399 (0.678)	1.642 (0.185)	1.660 (0.088)	1.667 (0.079)
	AP	19.69	37.85	37.36	31.98
3	E(X)	0.138 (0.597)	0.005 (0.052)	0.001 (0.029)	0.000 (0.026)
	E(X^2)	1.276 (0.581)	1.628 (0.219)	1.655 (0.090)	1.663 (0.088)
	AP	6.47	20.11	28.07	27.27

Table 3.11: Logistic Distribution (Sampling Density I)

μ_* \ σ_*		1	2	3	4
−3	E(X)	−0.369 (0.824)	0.002 (0.180)	−0.001 (0.037)	−0.002 (0.033)
	E(X^2)	2.337 (0.580)	3.308 (0.978)	3.292 (0.111)	3.296 (0.107)
	AP	14.67	25.94	37.51	38.29
−2	E(X)	−0.147 (0.669)	−0.002 (0.070)	−0.002 (0.029)	−0.001 (0.028)
	E(X^2)	2.651 (1.183)	3.279 (0.371)	3.293 (0.091)	3.295 (0.101)
	AP	29.77	44.99	50.90	45.13
−1	E(X)	−0.063 (0.431)	−0.001 (0.038)	−0.001 (0.024)	−0.002 (0.025)
	E(X^2)	2.861 (1.505)	3.283 (0.234)	3.294 (0.083)	3.293 (0.096)
	AP	53.80	70.45	61.56	49.86
0	E(X)	−0.001 (0.338)	0.001 (0.031)	0.000 (0.022)	−0.001 (0.027)
	E(X^2)	2.908 (1.463)	3.291 (0.237)	3.289 (0.076)	3.289 (0.093)
	AP	70.65	88.26	65.65	51.52
1	E(X)	0.060 (0.450)	0.002 (0.043)	0.000 (0.023)	0.000 (0.026)
	E(X^2)	2.841 (1.424)	3.292 (0.293)	3.287 (0.079)	3.290 (0.096)
	AP	53.89	70.41	61.54	49.85
2	E(X)	0.218 (0.554)	0.004 (0.068)	0.001 (0.030)	0.001 (0.028)
	E(X^2)	2.552 (0.985)	3.273 (0.339)	3.286 (0.091)	3.286 (0.102)
	AP	30.56	44.99	50.89	45.14
3	E(X)	0.404 (0.785)	0.004 (0.169)	0.002 (0.040)	0.002 (0.034)
	E(X^2)	2.336 (0.626)	3.254 (0.957)	3.292 (0.114)	3.288 (0.114)
	AP	14.87	25.93	37.52	38.29

Table 3.12: LaPlace Distribution (Sampling Density I)

μ_* \ σ_*		1	2	3	4
−3	E(X)	−0.188 (0.599)	−0.001 (0.085)	−0.001 (0.029)	−0.002 (0.027)
	E(X^2)	1.453 (0.578)	1.998 (0.431)	2.001 (0.094)	2.003 (0.093)
	AP	8.32	21.20	29.01	28.40
−2	E(X)	−0.052 (0.469)	−0.001 (0.043)	−0.002 (0.023)	0.000 (0.024)
	E(X^2)	1.692 (1.010)	1.997 (0.235)	2.001 (0.078)	2.006 (0.085)
	AP	20.88	38.44	38.54	33.27
−1	E(X)	−0.039 (0.204)	0.000 (0.029)	−0.001 (0.020)	−0.002 (0.022)
	E(X^2)	1.754 (0.715)	2.000 (0.189)	2.004 (0.071)	2.003 (0.085)
	AP	49.54	57.49	45.86	36.60
0	E(X)	−0.011 (0.249)	0.000 (0.021)	0.000 (0.019)	0.000 (0.023)
	E(X^2)	1.845 (1.151)	2.001 (0.155)	2.000 (0.069)	2.000 (0.083)
	AP	83.97	67.08	48.62	37.79
1	E(X)	0.029 (0.259)	0.000 (0.025)	0.000 (0.019)	0.001 (0.023)
	E(X^2)	1.766 (0.915)	2.001 (0.137)	1.998 (0.072)	2.002 (0.083)
	AP	49.45	57.48	45.86	36.62
2	E(X)	0.108 (0.359)	0.002 (0.044)	0.001 (0.023)	0.001 (0.024)
	E(X^2)	1.598 (0.782)	1.999 (0.224)	1.997 (0.080)	2.000 (0.087)
	AP	21.31	38.45	38.54	33.28
3	E(X)	0.201 (0.579)	0.005 (0.071)	0.001 (0.031)	0.001 (0.027)
	E(X^2)	1.444 (0.564)	1.975 (0.308)	1.998 (0.096)	2.000 (0.097)
	AP	8.35	21.18	29.01	28.39

Table 3.13: Gumbel Distribution (Sampling Density I)

μ_* \ σ_*		1	2	3	4
−3	E(X)	−0.151 (0.679)	0.000 (0.090)	0.000 (0.029)	−0.001 (0.029)
	E(X^2)	1.100 (0.822)	1.637 (0.457)	1.646 (0.078)	1.642 (0.074)
	AP	3.79	19.76	29.21	27.43
−2	E(X)	−0.063 (0.516)	0.000 (0.047)	0.000 (0.025)	0.000 (0.026)
	E(X^2)	1.334 (1.214)	1.643 (0.298)	1.644 (0.065)	1.647 (0.070)
	AP	18.91	39.78	38.17	31.77
−1	E(X)	−0.048 (0.214)	0.000 (0.029)	−0.001 (0.022)	−0.001 (0.026)
	E(X^2)	1.407 (0.674)	1.647 (0.195)	1.646 (0.058)	1.644 (0.065)
	AP	55.87	59.50	44.26	34.51
0	E(X)	−0.017 (0.130)	0.001 (0.023)	0.000 (0.022)	0.000 (0.026)
	E(X^2)	1.533 (0.571)	1.649 (0.155)	1.645 (0.055)	1.644 (0.064)
	AP	80.23	63.99	45.72	35.20
1	E(X)	−0.006 (0.075)	0.000 (0.023)	0.000 (0.024)	0.000 (0.028)
	E(X^2)	1.595 (0.360)	1.645 (0.073)	1.645 (0.057)	1.644 (0.065)
	AP	45.50	52.93	42.26	33.74
2	E(X)	0.006 (0.193)	0.002 (0.032)	0.000 (0.028)	0.001 (0.030)
	E(X^2)	1.627 (0.361)	1.647 (0.059)	1.645 (0.062)	1.647 (0.069)
	AP	20.65	35.44	35.13	30.42
3	E(X)	0.034 (0.717)	0.001 (0.049)	0.002 (0.035)	0.000 (0.033)
	E(X^2)	1.643 (0.607)	1.647 (0.083)	1.647 (0.071)	1.647 (0.075)
	AP	8.55	19.98	26.36	25.80

Table 3.14: Sampling Density II

σ_*		$t(5)$	Logistic	LaPlace	Gumbel
1	E(X)	0.000 (0.054)	0.000 (0.088)	−0.002 (0.063)	0.000 (0.054)
	E(X^2)	1.675 (0.316)	3.301 (0.283)	2.008 (0.220)	1.648 (0.175)
	AP	72.21	80.82	69.94	72.74
2	E(X)	−0.002 (0.037)	−0.001 (0.054)	−0.001 (0.039)	0.000 (0.035)
	E(X^2)	1.672 (0.190)	3.294 (0.175)	2.004 (0.139)	1.645 (0.106)
	AP	53.11	65.04	52.32	53.36
3	E(X)	0.000 (0.031)	−0.002 (0.045)	−0.002 (0.034)	−0.001 (0.032)
	E(X^2)	1.671 (0.142)	3.298 (0.157)	2.007 (0.124)	1.645 (0.093)
	AP	41.01	53.17	41.19	40.95
4	E(X)	−0.002 (0.030)	−0.002 (0.041)	−0.002 (0.031)	−0.001 (0.033)
	E(X^2)	1.667 (0.136)	3.296 (0.141)	2.005 (0.113)	1.646 (0.091)
	AP	33.06	44.41	33.64	32.78
5	E(X)	0.000 (0.030)	−0.003 (0.041)	−0.001 (0.032)	0.000 (0.034)
	E(X^2)	1.674 (0.133)	3.294 (0.140)	2.006 (0.116)	1.646 (0.092)
	AP	27.55	37.87	28.28	27.16
6	E(X)	−0.001 (0.031)	−0.003 (0.041)	0.000 (0.033)	0.002 (0.036)
	E(X^2)	1.676 (0.124)	3.295 (0.145)	2.005 (0.120)	1.648 (0.096)
	AP	23.54	32.85	24.32	23.11
7	E(X)	−0.001 (0.034)	0.001 (0.041)	−0.001 (0.035)	0.002 (0.037)
	E(X^2)	1.667 (0.126)	3.299 (0.150)	2.001 (0.126)	1.645 (0.097)
	AP	20.50	28.93	21.28	20.06

Table 3.15: Sampling Density III

ϵ		$t(5)$	Logistic	LaPlace	Gumbel
0.0	E(X)	0.411 (0.841)	0.006 (0.153)	−0.001 (0.041)	−0.141 (0.287)
	E(X^2)	2.144 (1.159)	2.459 (0.612)	2.033 (0.097)	1.187 (0.647)
	AP	52.24	68.14	61.69	58.34
0.05	E(X)	−0.012 (0.169)	0.000 (0.083)	−0.001 (0.040)	0.005 (0.166)
	E(X^2)	1.642 (0.331)	3.281 (0.349)	2.002 (0.097)	1.663 (0.345)
	AP	66.14	68.55	63.05	56.95
0.2	E(X)	0.001 (0.041)	0.001 (0.039)	−0.001 (0.040)	−0.002 (0.043)
	E(X^2)	1.653 (0.215)	3.281 (0.205)	2.002 (0.096)	1.639 (0.139)
	AP	73.38	83.94	63.05	63.25
0.3	E(X)	0.000 (0.037)	−0.001 (0.037)	−0.001 (0.040)	0.000 (0.042)
	E(X^2)	1.654 (0.173)	3.288 (0.173)	2.001 (0.096)	1.644 (0.125)
	AP	78.14	84.08	63.07	64.49
0.4	E(X)	0.000 (0.037)	0.001 (0.037)	−0.002 (0.040)	0.002 (0.039)
	E(X^2)	1.664 (0.238)	3.287 (0.149)	2.001 (0.095)	1.652 (0.112)
	AP	81.76	75.82	63.06	63.35

are given by $(\mu_*, \sigma_*) = (0, 3)$. The acceptance probabilities which give us the smallest standard error are 47.02 in Table 3.10, 65.65 in Table 3.11, 48.62 in Table 3.12 and 45.72 in Table 3.13, while the maximum acceptance probabilities are given by 92.80, 70.65, 83.97 and 80.23, respectively. Thus, for the independence chain (Sampling Density I) we can conclude that the sampling density which gives us the maximum acceptance probability is not necessarily the best choice. We should choose the larger value than the σ_* which gives us the maximum acceptance probability. For Sampling Density II, the acceptance probability decreases as σ_* increases for all the target distributions (see Table 3.14). As discussed above, the smallest standard errors of the estimated moments are given by $\sigma_* = 4, 5, 6$ for $t(5)$, $\sigma_* = 4, 5$ for Logistic, $\sigma_* = 4$ for LaPlace and $\sigma_* = 3, 4$ for Gumbel (see Table 3.14), where the acceptance probabilities are from 23.54 to 44.41. These acceptance probabilities are consistent with the results obtained in previous research (for example, see Carlin and Louis (1996, p.175), Chen, Shao and Ibrahim (2000, p.24), Gelman, Roberts and Gilks (1995), Besag, Green, Higdon and Mengersen (1995) and Gamerman (1997, p.165)). For all the four distributions, the standard errors at the optimum σ_* in Sampling Density II are larger than those in Sampling Density I. Therefore, if the appropriate sampling density is taken, Sampling Density I is better than Sampling Density II.

Summary: From Tables 3.7 – 3.15, Sampling Densities I and II with the appropriate hyper-parameters are better than Sampling Density III from the $\mathrm{Se}(\overline{g(x)})$ criterion, but Sampling Densities I and II are not too different from Sampling Density III. The problem of Sampling Densities I and II is that we have to choose the hyper-parameters, which is a crucial criticism. If we take the hyper-parameters which give us the least standard errors of the estimated moments, it extremely takes a lot of time because we have to compute the standard errors for many combinations of the hyper-parameters. Since Sampling Density III does not have the hyper-parameters such as μ_* and σ_*, it is easy to use Sampling Density III in practice and it can avoid the criticism of choosing the hyper-parameters included in Sampling Densities I and II. Remember that ϵ is in Sampling Density III, but it does not influence precision of the random draws. In addition, Sampling Density III shows quite good performance for all the four target densities. Therefore, Sampling Density III is more useful than Sampling Densities I and II.

Thus, in this section, we have considered choice of the sampling density in the MH algorithm. The three sampling densities have been examined, i.e., the independence chain (Sampling Density I), the random walk chain (Sampling Density II) and the Taylored chain (Sampling Density III). Through the simulation studies based on the five target distributions, we can summarize the obtained results as follows:

(i) Sampling Density I indicates the best performance when the appropriate hyper-parameters are chosen. That is, a scale parameter (σ_*) of Sampling Density I should be 1.5 – 2.5 times larger than that of the target density and a location parameter (μ_*) of Sampling Density I should be close to that of the target density.

(ii) Sampling Density II is optimal when the acceptance rate is about 30% – 50%, which result is consistent with past studies.

(iii) Sampling Density III shows a good performance for all the simulation studies. Moreover, because it does not depend on any crucial hyper-parameter, Sampling Density III is more useful than Sampling Densities I and II in practical use.

References

Ahrens, J.H. and Dieter, U., 1974, "Computer Methods for Sampling from Gamma, Beta, Poisson and Binomial Distributions," *Computing*, Vol.12, pp.223 – 246.

Bernardo, J.M. and Smith, A.F.M., 1994, *Bayesian Theory*, John Wiley & Sons.

Besag, J., Green, P., Higdon, D. and Mengersen, K., 1995, "Bayesian Computation and Stochastic Systems," *Statistical Science*, Vol.10, No.1, pp.3 – 66 (with discussion).

Boswell, M.T., Gore, S.D., Patil, G.P. and Taillie, C., 1993, "The Art of Computer Generation of Random Variables," in *Handbook of Statistics, Vol.9*, edited by Rao, C.R., pp.661 – 721, North-Holland.

Carlin, B.P. and Louis, T.A., 1996, *Bayes and Empirical Bayes Methods for Data Analysis*, Chapman & Hall.

Carlin, B.P. and Polson, N.G., 1991, "Inference for Nonconjugate Bayesian Models Using the Gibbs Sampler," *Canadian Journal of Statistics*, Vol.19, pp.399 – 405.

Carlin, B.P., Polson, N.G. and Stoffer, D.S., 1992, "A Monte Carlo Approach to Nonnormal and Nonlinear State Space Modeling," *Journal of the American Statistical Association*, Vol.87, No.418, pp.493 – 500.

Carter, C.K. and Kohn, R., 1994, "On Gibbs Sampling for State Space Models," *Biometrika*, Vol.81, No.3, pp.541 – 553.

Carter, C.K. and Kohn, R., 1996, "Markov Chain Monte Carlo in Conditionally Gaussian State Space Models," *Biometrika*, Vol.83, No.3, pp.589 – 601.

Casella, G. and George, E.I., 1992, "Explaining the Gibbs Sampler," *The American Statistician*, Vol.46, pp.167 – 174.

Chen, M.H., Shao, Q.M. and Ibrahim, J.G., 2000, *Monte Carlo Methods in Bayesian Computation*, Springer-Verlag.

Cheng, R.C.H., 1977, "The Generation of Gamma Variables with Non-Integral Shape Parameter," *Applied Statistics*, Vol.26, No.1, pp.71 – 75.

Cheng, R.C.H., 1998, "Random Variate Generation," in *Handbook of Simulation*, Chap.5, edited by Banks, J., pp.139 – 172, John Wiley & Sons.

Cheng, R.C.H. and Feast, G.M., 1979, "Some Simple Gamma Variate Generators," *Applied Statistics*, Vol.28, No.3, pp.290 – 295.

Cheng, R.C.H. and Feast, G.M., 1980, "Gamma Variate Generators with Increased Shape Parameter Range," *Communications of the ACM*, Vol.23, pp.389 – 393.

Chib, S. and Greenberg, E., 1995, "Understanding the Metropolis-Hastings Algorithm," *The American Statistician*, Vol.49, No.4, pp.327 – 335.

Chib, S., Greenberg, E. and Winkelmann, R., 1998, "Posterior Simulation and Bayes Factors in Panel Count Data Models," *Journal of Econometrics*, Vol.86, No.1, pp.33 – 54.

Fishman, G.S., 1996, *Monte Carlo: Concepts, Algorithms, and Applications*, Springer-Verlag.

Gamerman, D., 1997, *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference*, Chapman & Hall.

Gelfand, A.E., Hills, S.E., Racine-Poon, H.A. and Smith, A.F.M., 1990, "Illustration of Bayesian Inference in Normal Data Models Using Gibbs Sampling," *Journal of the American Statistical Association*, Vol.85, No.412, pp.972 – 985.

Gelfand, A.E. and Smith, A.F.M., 1990, "Sampling-Based Approaches to Calculating Marginal Densities," *Journal of the American Statistical Association*, Vol.85, No.410, pp.398 – 409.

Gelman, A., Roberts, G.O. and Gilks, W.R., 1996, "Efficient Metropolis Jumping Rules," in *Bayesian Statistics, Vol.5*, edited by Bernardo, J.M., Berger, J.O., Dawid, A.P. and Smith, A.F.M., pp.599 – 607, Oxford University Press.

Geman, S. and Geman D., 1984, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.Pami-6, No.6, pp.721 – 741.

Gentle, J.E., 1998, *Random Number Generation and Monte Carlo Methods*, Springer-Verlag.

Geweke, J., 1992, "Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments," in *Bayesian Statistics, Vol.4*, edited by Bernardo, J.M., Berger, J.O., Dawid, A.P. and Smith, A.F.M., pp.169 – 193 (with discussion), Oxford University Press.

Geweke, J., 1996, "Monte Carlo Simulation and Numerical Integration," in *Handbook of Computational Economics, Vol.1*, edited by Amman, H.M., Kendrick, D.A. and Rust, J., pp.731 – 800, North-Holland.

Geweke, J. and Tanizaki, H., 1999, "On Markov Chain Monte-Carlo Methods for Nonlinear and Non-Gaussian State-Space Models," *Communications in Statistics, Simulation and Computation*, Vol.28, No.4, pp.867 – 894.

Geweke, J. and Tanizaki, H., 2001, "Bayesian Estimation of State-Space Model Using the Metropolis-Hastings Algorithm within Gibbs Sampling," *Computational Statistics and Data Analysis*, Vol.37, No.2, pp.151-170.

Geweke, J. and Tanizaki, H., 2003, "Note on the Sampling Distribution for the Metropolis-Hastings Algorithm," *Communications in Statistics, Theory and Methods*, Vol.32, No.4, pp.775 – 789.

Kinderman, A.J. and Monahan, J.F., 1977, "Computer Generation of Random Variables Using the Ratio of Random Deviates," *ACM Transactions on Mathematical Software*, Vol.3, pp.257 – 260.

Liu, J.S., 1996, "Metropolized Independent Sampling with Comparisons to Rejection Sampling and Importance Sampling," *Statistics and Computing*, Vol.6, pp.113 – 119.

Mengersen, K.L., Robert, C.P. and Guihenneuc-Jouyaux, C., 1999, "MCMC Convergence Diagnostics: A Reviewww," in *Bayesian Statistics, Vol.6*, edited by Bernardo, J.M., Berger, J.O., Dawid, A.P. and Smith, A.F.M., pp.514 – 440 (with discussion), Oxford University Press.

O'Hagan, A., 1994, *Kendall's Advanced Theory of Statistics*, Vol.2B (Bayesian Inference), Edward Arnold.

Ripley, B.D., 1987, *Stochastic Simulation*, John Wiley & Sons.

Robert, C.P. and Casella, G., 1999, *Monte Carlo Statistical Methods*, Springer-Verlag.

Sarkar, T.K., 1996, "A Composition-Alias Method for Generating Gamma Variates with Shape Parameter Greater Than 1," *ACM Transactions on Mathematical Software*, Vol.22, pp.484 – 492.

Schmeiser, B. and Lal, R., 1980, "Squeeze Methods for Generating Gamma Variates," *Journal of the American Statistical Association*, Vol.75, pp.679 – 682.

Smith, A.F.M. and Gelfand, A.E., 1992, "Bayesian Statistics without Tears: A Sampling-Resampling Perspective," *The American Statistician*, Vol.46, No.2, pp.84 – 88.

Smith, A.F.M. and Roberts, G.O., 1993, "Bayesian Computation via Gibbs Sampler and Related Markov Chain Monte Carlo Methods," *Journal of the Royal Statistical Society*, Ser.B, Vol.55, No.1, pp.3 – 23.

Tanner, M.A. and Wong, W.H., 1987, "The Calculation of Posterior Distributions by Data Augmentation," *Journal of the American Statistical Association*, Vol.82, No.398, pp.528 – 550 (with discussion).

Tierney, L., 1994, "Markov Chains for Exploring Posterior Distributions," *The Annals of Statistics*, Vol.22, No.4, pp.1701 – 1762.

Zeger, S.L. and Karim, M.R., 1991, "Generalized Linear Models with Random Effects: A Gibbs Sampling Approach," *Journal of the American Statistical Association*, Vol.86, No.413, pp.79 – 86.

Part II

Selected Applications of Monte Carlo Methods

Chapter 4

Bayesian Estimation

In Section 4.1, Bayes' procedure is briefly discussed (see Zellner (1971), Bernardo and Smith (1994), O'Hagan (1994), Hogg and Craig (1995) and so on for further discussion). In Sections 4.2 and 4.3, the Bayesian approach is applied to regression models. The heteroscedasticity model proposed by Harvey (1976) is discussed in Section 4.2, while the autocorrelation model discussed by Chib (1993) is introduced in Section 4.3,

4.1 Elements of Bayesian Inference

When we have the random sample $(X_1, X_2, \cdots, X_n)$, consider estimating the unknown parameter θ. In Section 1.7.5, the maximum likelihood estimator is introduced for estimation of the parameter. Suppose that X_1, X_2, $\cdots$, X_n are mutually independently distributed and X_i has a probability density function $f(x;\theta)$, where θ is the unknown parameter to be estimated. As discussed in Section 1.7.5, the joint density of X_1, X_2, $\cdots$, X_n is given by:

$$f(x_1, x_2, \cdots, x_3; \theta) = \prod_{i=1}^{n} f(x_i; \theta),$$

which is called the likelihood function, denoted by $l(\theta) = f(x_1, x_2, \cdots, x_n; \theta)$. In Bayes' estimation, the parameter is taken as a random variable, say Θ, where a prior information on Θ is taken into account for estimation. The joint density function (or the likelihood function) is regarded as the conditional density function of X_1, X_2, $\cdots$, X_n given $\Theta = \theta$. Therefore, we write the likelihood function as the conditional density $f(x_1, x_2, \cdots, x_n|\theta)$. The probability density function of Θ is called the **prior probability density function** and given by $f_\theta(\theta)$. The conditional probability density function, $f_{\theta|x}(\theta|x_1, x_2, \cdots, x_n)$, have to be obtained, which is represented as:

$$\begin{aligned} f_{\theta|x}(\theta|x_1, x_2, \cdots, x_n) &= \frac{f(x_1, x_2, \cdots, x_n|\theta)\, f_\theta(\theta)}{\int f(x_1, x_2, \cdots, x_n|\theta)\, f_\theta(\theta)\, \mathrm{d}\theta} \\ &\propto f(x_1, x_2, \cdots, x_n|\theta)\, f_\theta(\theta). \end{aligned}$$

The relationship in the first equality is known as Bayes' formula. The conditional probability density function of Θ given $X_1 = x_1$, $X_2 = x_2$, $\cdots$, $X_n = x_n$, i.e., $f_{\theta|x}(\theta|x_1, x_2, \cdots, x_n)$, is called the **posterior probability density function**, which is proportional to the product of the likelihood function and the prior density function.

4.1.1 Bayesian Point Estimate

Thus, the Bayesian approach yields the posterior probability density function for Θ. To obtain a point estimate of Θ, we introduce a loss function, denoted by $L(\Theta, \hat{\theta})$, where $\hat{\theta}$ indicates a point estimate depending on $X_1 = x_1$, $X_2 = x_2$, $\cdots$, $X_n = x_n$. Since Θ is considered to be random, $L(\Theta, \hat{\theta})$ is also random. One solution which yields point estimates is to find the value of Θ that minimizes the mathematical expectation of the **loss function**, i.e.,

$$\min_{\hat{\theta}} \mathrm{E}\Big(L(\Theta, \hat{\theta})\Big) = \min_{\hat{\theta}} \int L(\theta, \hat{\theta}) f_{\theta|x}(\theta|x_1, x_2, \cdots, x_n)\, \mathrm{d}\theta,$$

where the absolute value of $\mathrm{E}\Big(L(\Theta, \hat{\theta})\Big)$ is assumed to be finite.

Now we specify the loss function as: $L(\Theta, \hat{\theta}) = (\Theta - \hat{\theta})' A (\Theta - \hat{\theta})$, which is called the **quadratic loss function**, where A is a known nonstochastic positive definite symmetric matrix. Then, the solution gives us the posterior mean of Θ, i.e.,

$$\hat{\theta} = \mathrm{E}(\Theta) = \int \theta f_{\theta|x}(\theta|x_1, x_2, \cdots, x_n)\, \mathrm{d}\theta.$$

An alternative loss function is given by: $L(\Theta, \hat{\theta}) = |\Theta - \hat{\theta}|$, which is called the **absolute error loss function**, where both Θ and $\hat{\theta}$ are assumed to be scalars. Then, the median of the posterior probability density function is an optimal point estimate of θ, i.e.,

$$\hat{\theta} = \text{median of the posterior probability density function.}$$

We have shown two Bayesian point estimates. Hereafter, the quadratic loss function is adopted for estimation. That is, the posterior mean of Θ is taken as the point estimate.

4.1.2 Bayesian Interval for Parameter

Given that the posterior probability density function $f_{\theta|x}(\theta|x_1, x_2, \cdots, x_n)$ has been obtained, it is possible to compute the probability that the parameter Θ lies in a particular subregion, R, of the parameter space. That is, we may compute the following probability:

$$P(\Theta \in R) = \int_R f_{\theta|x}(\theta|x_1, x_2, \cdots, x_n)\, \mathrm{d}\theta.$$

When the above probability is set to be $1 - \alpha$, it is possible to find the region that satisfies $P(\Theta \in R) = 1 - \alpha$, which region is not necessarily unique. In the case where

Θ is a scalar, one possibility to determine the unique region $R = \{\Theta | a < \Theta < b\}$ is to obtain a and b by minimizing the distance $b - a$ subject to $\int_a^b f_{\theta|x}(\theta|x_1, x_2, \cdots, x_n)\, \mathrm{d}\theta = 1-\alpha$. By solving this minimization problem, determining a and b such that $\int_a^b f_{\theta|x}(\theta|x_1, x_2, \cdots, x_n)\, \mathrm{d}\theta = 1 - \alpha$ and $f_{\theta|x}(a|x_1, x_2, \cdots, x_n) = f_{\theta|x}(b|x_1, x_2, \cdots, x_n)$ leads to the shortest interval with probability $1 - \alpha$.

4.1.3 Prior Probability Density Function

We discuss a little bit about the prior probability density function. In the case where we know any information about the parameter θ beforehand, the plausible estimate of the parameter might be obtained if the parameter θ is estimated by including the prior information. For example, if we know that Θ is normally distributed with mean θ_0 and variance Σ_0, the prior density $f_\theta(\theta)$ is given by $N(\theta_0, \Sigma_0)$, where θ_0 and Σ_0 are known. On the contrary, we have the case where we do not know any prior information about the parameter θ. In this case, we may take the prior density as:

$$f_\theta(\theta) \propto \text{constant},$$

where the prior density of Θ is assumed to be uniform, which prior is called the **improper prior**, the **noninformative prior**, the **flat prior** or the **diffuse prior**. Then, the posterior density is given by:

$$f_{\theta|x}(\theta|x_1, x_2, \cdots, x_n) \propto f(x_1, x_2, \cdots, x_n|\theta).$$

That is, the posterior density function is proportional to the likelihood function.

Example: Suppose that $X_1, X_2, \cdots, X_n$ are mutually independently, identically and normally distributed with mean μ and variance σ^2. Then, the likelihood function is given by:

$$\begin{aligned} f(x_1, x_2, \cdots, x_n|\theta) &= \prod_{i=1}^n f(x_i; \theta) = \prod_{i=1}^n (2\pi\sigma^2)^{-1/2} \exp\Big(-\frac{1}{2\sigma^2}(x_i - \mu)^2\Big) \\ &= (2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{1}{2\sigma^2}\sum_{i=1}^n (x_i - \mu)^2\Big), \end{aligned}$$

where θ indicates μ in this case. For simplicity of discussion, we assume that σ^2 is known. Therefore, we focus on μ.

Now, we consider two prior density functions for μ. One is noninformative and another is normal, i.e.,

(i) **Noninformative Prior:** $f_\theta(\mu) \propto$ constant, where μ is uniformly distributed.

(ii) **Normal Prior:** $f_\theta(\mu) = (2\pi\sigma_0^2)^{-1/2} \exp\Big(-\frac{1}{2\sigma_0^2}(\mu - \mu_0)^2\Big)$, where μ_0 and σ_0 are assumed to be known.

For each prior density, we obtain the posterior distributions as follows:

(i) When the prior density is noninformative, the posterior density function is:

$$\begin{aligned}
&f_{\theta|x}(\mu|x_1, x_2, \cdots, x_n) \\
&\quad \propto f(x_1, x_2, \cdots, x_n|\mu) = (2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2\Big) \\
&\quad = (2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\overline{x})^2 - \frac{1}{2\sigma^2}n(\overline{x}-\mu)^2\Big) \\
&\quad = (2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{(n-1)s^2}{2\sigma^2} - \frac{1}{2\sigma^2/n}(\mu-\overline{x})^2\Big) \\
&\quad \propto \exp\Big(-\frac{1}{2\sigma^2/n}(\mu-\overline{x})^2\Big)
\end{aligned}$$

where $\overline{x} = \sum_{i=1}^{n} x_i/n$ and $s^2 = \sum_{i=1}^{n}(x_i-\overline{x})^2/(n-1)$. Thus, the posterior density of μ represents the normal distribution with mean $\overline{x}$ and variance σ^2/n. Since under the quadratic loss function the point estimate of μ is given by the posterior mean, $\overline{x}$ gives us Bayes' point estimate. The Bayesian interval estimate of μ is: $(\overline{x} - z_{\alpha/2}\sigma/\sqrt{n},\ \overline{x} + z_{\alpha/2}\sigma/\sqrt{n})$, because from the posterior density function we have $P\Big(\Big|\dfrac{\mu-\overline{x}}{\sigma/\sqrt{n}}\Big| < z_{\alpha/2}\Big) = 1-\alpha$.

(ii) When the prior density is normal, the posterior density function is given by:

$$\begin{aligned}
&f_{\theta|x}(\mu|x_1, x_2, \cdots, x_n) \\
&\quad \propto f(x_1, x_2, \cdots, x_n|\mu)\, f_\theta(\mu) \\
&\quad = (2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2\Big) \\
&\qquad \times(2\pi\sigma_0^2)^{-1/2} \exp\Big(-\frac{1}{2\sigma_0^2}(\mu-\mu_0)^2\Big) \\
&\quad \propto \exp\Big(-\frac{1}{2\sigma^2/n}(\mu-\overline{x})^2\Big) \times \exp\Big(-\frac{1}{2\sigma_0^2}(\mu-\mu_0)^2\Big) \\
&\quad \propto \exp\Bigg(-\frac{1}{2}\Big(\frac{\sigma_0^2+\sigma^2/n}{\sigma_0^2\sigma^2/n}\Big)\Big(\mu - \frac{\overline{x}\sigma_0^2+\mu_0\sigma^2/n}{\sigma_0^2+\sigma^2/n}\Big)^2\Bigg),
\end{aligned}$$

which indicates that the posterior density of μ is a normal distribution with mean $\dfrac{\overline{x}\sigma_0^2+\mu_0\sigma^2/n}{\sigma_0^2+\sigma^2/n}$ and variance $\Big(\dfrac{\sigma_0^2+\sigma^2/n}{\sigma_0^2\sigma^2/n}\Big)^{-1}$. The posterior mean is rewritten as:

$$\frac{\overline{x}\sigma_0^2+\mu_0\sigma^2/n}{\sigma_0^2+\sigma^2/n} = \overline{x}w + \mu_0(1-w),$$

where $w = \sigma_0^2/(\sigma_0^2 + \sigma^2/n)$. $\overline{x}$ is a maximum likelihood estimate of μ and μ_0 is a prior mean of μ. Thus, the posterior mean is the weighted average of $\overline{x}$ and μ_0. As $w \longrightarrow 1$, i.e., as $\sigma_0^2 \longrightarrow \infty$, the posterior mean approaches $\overline{x}$, which is equivalent to the posterior mean with the noninformative prior.

We have discussed the Bayes procedure briefly. In the next two sections, we deal with simple regression models. In Section 4.2 the regression model with heteroscedastic errors is taken, and in Section 4.3 the regression model with autocorrelated errors is examined. We estimate unknown parameters with the Bayes procedure, where the Gibbs sampler and the Metropolis-Hastings algorithm are utilized for the random number generation techniques.

4.2 Heteroscedasticity Model

In Section 4.2, Tanizaki and Zhang (2001) is re-computed using the random number generators discussed in Chapters 2 and 3. Here, we show how to use Bayesian approach in the multiplicative heteroscedasticity model discussed by Harvey (1976). The Gibbs sampler and the Metropolis-Hastings (MH) algorithm are applied to the multiplicative heteroscedasticity model, where some sampling densities are considered in the MH algorithm. We carry out Monte Carlo study to examine the properties of the estimates via Bayesian approach and the traditional counterparts such as the modified two-step estimator (M2SE) and the maximum likelihood estimator (MLE). The results of Monte Carlo study show that the sampling density chosen here is suitable, and Bayesian approach shows better performance than the traditional counterparts in the criterion of the root mean square error (RMSE) and the interquartile range (IR).

4.2.1 Introduction

For the heteroscedasticity model, we have to estimate both the regression coefficients and the heteroscedasticity parameters. In the literature of heteroscedasticity, traditional estimation techniques include the two-step estimator (2SE) and the maximum likelihood estimator (MLE). Harvey (1976) showed that the 2SE has an inconsistent element in the heteroscedasticity parameters and furthermore derived the consistent estimator based on the 2SE, which is called the modified two-step estimator (M2SE). These traditional estimators are also examined in Amemiya (1985), Judge, Hill, Griffiths and Lee (1980) and Greene (1997).

Ohtani (1982) derived the Bayesian estimator (BE) for a heteroscedasticity linear model. Using a Monte Carlo experiment, Ohtani (1982) found that among the Bayesian estimator (BE) and some traditional estimators, the Bayesian estimator (BE) shows the best properties in the mean square error (MSE) criterion. Because Ohtani (1982) obtained the Bayesian estimator by numerical integration, it is not easy to

extend to the multi-dimensional cases of both the regression coefficient and the heteroscedasticity parameter.

Recently, Boscardin and Gelman (1996) developed a Bayesian approach in which a Gibbs sampler and the Metropolis-Hastings (MH) algorithm are used to estimate the parameters of heteroscedasticity in the linear model. They argued that through this kind of Bayesian approach, we can average over our uncertainty in the model parameters instead of using a point estimate via the traditional estimation techniques. Their modeling for the heteroscedasticity, however, is very simple and limited. Their choice of the heteroscedasticity is $V(y_i) = \sigma^2 w_i^{-\theta}$, where w_i are known "weights" for the problem and θ is an unknown parameter. In addition, they took only one candidate for the sampling density used in the MH algorithm and compared it with 2SE.

In Section 4.2, we also consider Harvey's (1976) model of multiplicative heteroscedasticity. This modeling is very flexible, general, and includes most of the useful formulations for heteroscedasticity as special cases. The Bayesian approach discussed by Ohtani (1982) and Boscardin and Gelman (1996) can be extended to the multi-dimensional and more complicated cases, using the model introduced here. The Bayesian approach discussed here includes the MH within Gibbs algorithm, where through Monte Carlo studies we examine two kinds of candidates for the sampling density in the MH algorithm and compare the Bayesian approach with the two traditional estimators, i.e., M2SE and MLE, in the criterion of the root mean square error (RMSE) and the interquartile range (IR). We obtain the results that the Bayesian estimator significantly has smaller RMSE and IR than M2SE and MLE at least for the heteroscedasticity parameters. Thus, the results of the Monte Carlo study show that the Bayesian approach performs better than the traditional estimators.

4.2.2 Multiplicative Heteroscedasticity Regression Model

The multiplicative heteroscedasticity model discussed by Harvey (1976) can be shown as follows:

$$y_t = X_t\beta + u_t, \qquad u_t \sim N(0, \sigma_t^2), \tag{4.1}$$

$$\sigma_t^2 = \sigma^2 \exp(q_t\alpha), \tag{4.2}$$

for $t = 1, 2, \cdots, n$, where y_t is the tth observation, X_t and q_t are the tth $1 \times k$ and $1 \times (J-1)$ vectors of explanatory variables, respectively. β and α are vectors of unknown parameters.

The model given by equations (4.1) and (4.2) includes several special cases such as the model in Boscardin and Gelman (1996), in which $q_t = \log w_t$ and $\theta = -\alpha$. As shown in Greene (1997), there is a useful simplification of the formulation. Let $z_t = (1, q_t)$ and $\gamma = (\log\sigma^2, \alpha')'$, where z_t and γ denote $1 \times J$ and $J \times 1$ vectors. Then, we can simply rewrite equation (4.2) as:

$$\sigma_t^2 = \exp(z_t\gamma). \tag{4.3}$$

Note that $\exp(\gamma_1)$ provides σ^2, where γ_1 denotes the first element of γ. As for the variance of u_t, hereafter we use (4.3), rather than (4.2).

The generalized least squares (GLS) estimator of β, denoted by $\hat{\beta}_{GLS}$, is given by:

$$\hat{\beta}_{GLS} = \Big(\sum_{t=1}^{n} \exp(-z_t\gamma)X_t'X_t\Big)^{-1} \sum_{t=1}^{n} \exp(-z_t\gamma)X_t'y_t, \tag{4.4}$$

where $\hat{\beta}_{GLS}$ depends on γ, which is the unknown parameter vector. To obtain the feasible GLS estimator, we need to replace γ by its consistent estimate. We have two traditional consistent estimators of γ, i.e., M2SE and MLE, which are briefly described as follows.

Modified Two-Step Estimator (M2SE): First, define the ordinary least squares (OLS) residual by $e_t = y_t - X_t\hat{\beta}_{OLS}$, where $\hat{\beta}_{OLS}$ represents the OLS estimator, i.e., $\hat{\beta}_{OLS} = (\sum_{t=1}^{n} X_t'X_t)^{-1} \sum_{t=1}^{n} X_t'y_t$. For 2SE of γ, we may form the following regression:

$$\log e_t^2 = z_t\gamma + v_t.$$

The OLS estimator of γ applied to the above equation leads to the 2SE of γ, because e_t is obtained by OLS in the first step. Thus, the OLS estimator of γ gives us 2SE, denoted by $\hat{\gamma}_{2SE}$, which is given by:

$$\hat{\gamma}_{2SE} = (\sum_{t=1}^{n} z_t'z_t)^{-1} \sum_{t=1}^{n} z_t' \log e_t^2.$$

A problem with this estimator is that v_t, $t = 1, 2, \cdots, n$, have non-zero means and are heteroscedastic. If e_t converges in distribution to u_t, the v_t will be asymptotically independent with mean $\mathrm{E}(v_t) = -1.2704$ and variance $\mathrm{V}(v_t) = 4.9348$, which are shown in Harvey (1976). Then, we have the following mean and variance of $\hat{\gamma}_{2SE}$:

$$\begin{aligned} \mathrm{E}(\hat{\gamma}_{2SE}) &= \gamma - 1.2704(\sum_{t=1}^{n} z_t'z_t)^{-1} \sum_{t=1}^{n} z_t', \\ \mathrm{V}(\hat{\gamma}_{2SE}) &= 4.9348(\sum_{t=1}^{n} z_t'z_t)^{-1}. \end{aligned} \tag{4.5}$$

For the second term in equation (4.5), the first element is equal to -1.2704 and the remaining elements are zero, which can be obtained by simple calculation. Therefore, the first element of $\hat{\gamma}_{2SE}$ is biased but the remaining elements are still unbiased. To obtain a consistent estimator of γ_1, we consider M2SE of γ, denoted by $\hat{\gamma}_{M2SE}$, which is given by:

$$\hat{\gamma}_{M2SE} = \hat{\gamma}_{2SE} + 1.2704(\sum_{t=1}^{n} z_t'z_t)^{-1} \sum_{t=1}^{n} z_t'.$$

Let Σ_{M2SE} be the variance of $\hat{\gamma}_{M2SE}$. Then, Σ_{M2SE} is represented by:

$$\Sigma_{M2SE} \equiv \mathrm{V}(\hat{\gamma}_{M2SE}) = \mathrm{V}(\hat{\gamma}_{2SE}) = 4.9348(\sum_{t=1}^{n} z_t' z_t)^{-1}.$$

The first element of $\hat{\gamma}_{2SE}$ and $\hat{\gamma}_{M2SE}$ corresponds to the estimate of σ^2, which value does not influence $\hat{\beta}_{GLS}$. Since the remaining elements of $\hat{\gamma}_{2SE}$ are equal to those of $\hat{\gamma}_{M2SE}$, $\hat{\beta}_{2SE}$ is equivalent to $\hat{\beta}_{M2SE}$, where $\hat{\beta}_{2SE}$ and $\hat{\beta}_{M2SE}$ denote 2SE and M2SE of β, respectively. Note that $\hat{\beta}_{2SE}$ and $\hat{\beta}_{M2SE}$ can be obtained by substituting $\hat{\gamma}_{2SE}$ and $\hat{\gamma}_{M2SE}$ into γ in (4.4).

Maximum Likelihood Estimator (MLE): The density of $Y_n = (y_1, y_2, \cdots, y_n)$ based on (4.1) and (4.3) is:

$$f(Y_n|\beta, \gamma) \propto \exp\left(-\frac{1}{2}\sum_{t=1}^{n}\Big(\exp(-z_t\gamma)(y_t - X_t\beta)^2 + z_t\gamma\Big)\right), \tag{4.6}$$

which is maximized with respect to β and γ, using the method of scoring. That is, given values for $\beta^{(j)}$ and $\gamma^{(j)}$, the method of scoring is implemented by the following iterative procedure:

$$\beta^{(j)} = \Big(\sum_{t=1}^{n} \exp(-z_t\gamma^{(j-1)})X_t'X_t\Big)^{-1} \sum_{t=1}^{n} \exp(-z_t\gamma^{(j-1)})X_t'y_t,$$

$$\gamma^{(j)} = \gamma^{(j-1)} + 2(\sum_{t=1}^{n} z_t'z_t)^{-1}\frac{1}{2}\sum_{t=1}^{n} z_t'\Big(\exp(-z_t\gamma^{(j-1)})e_t^2 - 1\Big),$$

for $j = 1, 2, \cdots$, where $e_t = y_t - X_t\beta^{(j-1)}$. The starting value for the above iteration may be taken as $(\beta^{(0)}, \gamma^{(0)}) = (\hat{\beta}_{OLS}, \hat{\gamma}_{2SE})$, $(\hat{\beta}_{2SE}, \hat{\gamma}_{2SE})$ or $(\hat{\beta}_{M2SE}, \hat{\gamma}_{M2SE})$.

Let $\theta = (\beta, \gamma)$. The limit of $\theta^{(j)} = (\beta^{(j)}, \gamma^{(j)})$ gives us the MLE of θ, which is denoted by $\hat{\theta}_{MLE} = (\hat{\beta}_{MLE}, \hat{\gamma}_{MLE})$. Based on the information matrix, the asymptotic covariance matrix of $\hat{\theta}_{MLE}$ is represented by:

$$\begin{aligned}\mathrm{V}(\hat{\theta}_{MLE}) &= \left(-\mathrm{E}\left(\frac{\partial^2 \log f(Y_n|\theta)}{\partial\theta\partial\theta'}\right)\right)^{-1} \\ &= \begin{pmatrix} \big(\sum_{t=1}^{n} \exp(-z_t\gamma)X_t'X_t\big)^{-1} & 0 \\ 0 & 2(\sum_{t=1}^{n} z_t'z_t)^{-1} \end{pmatrix}. \end{aligned} \tag{4.7}$$

Thus, from (4.7), asymptotically there is no correlation between $\hat{\beta}_{MLE}$ and $\hat{\gamma}_{MLE}$, and furthermore the asymptotic variance of $\hat{\gamma}_{MLE}$ is represented by: $\Sigma_{MLE} \equiv \mathrm{V}(\hat{\gamma}_{MLE}) = 2(\sum_{t=1}^{n} z_t'z_t)^{-1}$, which implies that $\hat{\gamma}_{M2SE}$ is asymptotically inefficient because $\Sigma_{M2SE} - \Sigma_{MLE}$ is positive definite. Remember that the variance of $\hat{\gamma}_{M2SE}$ is given by: $\mathrm{V}(\hat{\gamma}_{M2SE}) = 4.9348(\sum_{t=1}^{n} z_t'z_t)^{-1}$.

4.2.3 Bayesian Estimation

We assume that the prior distributions of the parameters β and γ are noninformative, which are represented by:

$$f_\beta(\beta) = \text{constant}, \qquad f_\gamma(\gamma) = \text{constant}. \tag{4.8}$$

Combining the prior distributions (4.8) and the likelihood function (4.6), the posterior distribution $f_{\beta\gamma}(\beta, \gamma|y)$ is obtained as follows:

$$f_{\beta\gamma}(\beta, \gamma|Y_n) \propto \exp\left(-\frac{1}{2}\sum_{t=1}^{n}\Big(\exp(-z_t\gamma)(y_t - X_t\beta)^2 + z_t\gamma\Big)\right).$$

The posterior means of β and γ are not operationally obtained. Therefore, by generating random draws of β and γ from the posterior density $f_{\beta\gamma}(\beta, \gamma|Y_n)$, we consider evaluating the mathematical expectations as the arithmetic averages based on the random draws.

Now we utilize the Gibbs sampler, which has been introduced in Section 3.6, to sample random draws of β and γ from the posterior distribution. Then, from the posterior density $f_{\beta\gamma}(\beta, \gamma|Y_n)$, we can derive the following two conditional densities:

$$f_{\gamma|\beta}(\gamma|\beta, Y_n) \propto \exp\left(-\frac{1}{2}\sum_{t=1}^{n}\Big(\exp(-z_t\gamma)(y_t - X_t\beta)^2 + z_t\gamma\Big)\right), \tag{4.9}$$

$$f_{\beta|\gamma}(\beta|\gamma, Y_n) = N(B_1, H_1), \tag{4.10}$$

where

$$H_1^{-1} = \sum_{t=1}^{n}\exp(-z_t\gamma)X_t'X_t, \qquad B_1 = H_1\sum_{t=1}^{n}\exp(-z_t\gamma)X_t'y_t.$$

Sampling from (4.10) is simple since it is a k-variate normal distribution with mean B_1 and variance H_1. However, since the J-variate distribution (4.9) does not take the form of any standard density, it is not easy to sample from (4.9). In this case, the MH algorithm discussed in Section 3.4 can be used within the Gibbs sampler. See Tierney (1994) and Chib and Greeberg (1995) for a general discussion.

Let γ_{i-1} be the $(i-1)$th random draw of γ and γ^* be a candidate of the ith random draw of γ. The MH algorithm utilizes another appropriate distribution function $f_*(\gamma|\gamma_i)$, which is called the sampling density or the proposal density. Let us define the acceptance rate $\omega(\gamma_{i-1}, \gamma^*)$ as:

$$\omega(\gamma_{i-1}, \gamma^*) = \min\left(\frac{f_{\gamma|\beta}(\gamma^*|\beta_{i-1}, Y_n)/\,f_*(\gamma^*|\gamma_{i-1})}{f_{\gamma|\beta}(\gamma_{i-1}|\beta_{i-1}, Y_n)/\,f_*(\gamma_{i-1}|\gamma^*)},\ 1\right).$$

The sampling procedure based on the MH algorithm within Gibbs sampling is as follows:

(i) Set the initial value β_{-M}, which may be taken as $\hat{\beta}_{M2SE}$ or $\hat{\beta}_{MLE}$.

(ii) Given β_{i-1}, generate a random draw of γ, denoted by γ_i, from the conditional density $f_{\gamma|\beta}(\gamma|\beta_{i-1}, Y_n)$, where the MH algorithm is utilized for random number generation because it is not easy to generate random draws of γ from (4.9). The Metropolis-Hastings algorithm is implemented as follows:

  (a) Given γ_{i-1}, generate a random draw γ^* from $f_*(\cdot|\gamma_{i-1})$ and compute the acceptance rate $\omega(\gamma_{i-1}, \gamma^*)$. We will discuss later about the sampling density $f_*(\gamma|\gamma_{i-1})$.
  (b) Set $\gamma_i = \gamma^*$ with probability $\omega(\gamma_{i-1}, \gamma^*)$ and $\gamma_i = \gamma_{i-1}$ otherwise,

(iii) Given γ_i, generate a random draw of β, denoted by β_i, from the conditional density $f_{\beta|\gamma}(\beta|\gamma_i, Y_n)$, which is $\beta|\gamma_i, Y_n \sim N(B_1, H_1)$ as shown in (4.10).

(iv) Repeat (ii) and (iii) for $i = -M+1, -M+2, \cdots, N$.

Note that the iteration of Steps (ii) and (iii) corresponds to the Gibbs sampler, which iteration yields random draws of β and γ from the joint density $f_{\beta\gamma}(\beta, \gamma|Y_n)$ when i is large enough. It is well known that convergence of the Gibbs sampler is slow when β is highly correlated with γ. That is, a large number of random draws have to be generated in this case. Therefore, depending on the underlying joint density, we have the case where the Gibbs sampler does not work at all. For example, see Chib and Greenberg (1995) for convergence of the Gibbs sampler. In the model represented by (4.1) and (4.2), however, there is asymptotically no correlation between $\hat{\beta}_{MLE}$ and $\hat{\gamma}_{MLE}$, as shown in (4.7). It might be expected that correlation between $\hat{\beta}_{MLE}$ and $\hat{\gamma}_{MLE}$ is not too high even in the small sample. Therefore, it might be appropriate to consider that the Gibbs sampler works well in this model.

In Step (ii), the sampling density $f_*(\gamma|\gamma_{i-1})$ is utilized. We consider the multivariate normal density function for the sampling distribution, which is discussed as follows.

Choice of the Sampling Density in Step (ii): Several generic choices of the sampling density are discussed by Tierney (1994) and Chib and Greenberg (1995). Here, we take $f_*(\gamma|\gamma_{i-1}) = f_*(\gamma)$ as the sampling density, which is called the independence chain because the sampling density is not a function of γ_{i-1}. We consider taking the multivariate normal sampling density in the independence MH algorithm, because of its simplicity. Therefore, $f_*(\gamma)$ is taken as follows:

$$f_*(\gamma) = N(\gamma^+, c^2\Sigma^+), \tag{4.11}$$

which represents the J-variate normal distribution with mean γ^+ and variance $c^2\Sigma^+$. The tuning parameter c is introduced into the sampling density (4.11). In Section 3.7.5, we have mentioned that for the independence chain (Sampling Density I) the sampling density with the variance which gives us the maximum acceptance probability is not necessarily the best choice. From some Monte Carlo experiments in Section 3.7.5, we have obtained the result that the sampling density with the 1.5 – 2.5

times larger standard error is better than that with the standard error which maximizes the acceptance probability. Therefore, $c = 2$ is taken in the next section, and it is the larger value than the c which gives us the maximum acceptance probability. This detail discussion is given in Section 4.2.4.

Thus, the sampling density of γ is normally distributed with mean γ^+ and variance $c^2\Sigma^+$. As for (γ^+, Σ^+), in the next section we choose one of $(\hat{\gamma}_{M2SE}, \Sigma_{M2SE})$ and $(\hat{\gamma}_{MLE}, \Sigma_{MLE})$ from the criterion of the acceptance rate. As shown in Section 2, both of the two estimators $\hat{\gamma}_{M2SE}$ and $\hat{\gamma}_{MLE}$ are consistent estimates of γ. Therefore, it might be very plausible to consider that the sampling density is distributed around the consistent estimates.

Bayesian Estimator: From the convergence theory of the Gibbs sampler and the MH algorithm, as i goes to infinity we can regard γ_i and β_i as random draws from the target density $f_{\beta\gamma}(\beta, \gamma|Y_n)$. Let M be a sufficiently large number. γ_i and β_i for $i = 1, 2, \cdots, N$ are taken as the random draws from the posterior density $f_{\beta\gamma}(\beta, \gamma|Y_n)$. Therefore, the Bayesian estimators $\hat{\gamma}_{BZZ}$ and $\hat{\beta}_{BZZ}$ are given by:

$$\hat{\gamma}_{BZZ} = \frac{1}{N}\sum_{i=1}^{N}\gamma_i, \qquad \hat{\beta}_{BZZ} = \frac{1}{N}\sum_{i=1}^{N}\beta_i,$$

where we read the subscript BZZ as the Bayesian estimator which uses the multivariate normal sampling density with mean $\hat{\gamma}_{ZZ}$ and variance Σ_{ZZ}. ZZ takes M2SE or MLE. We consider two kinds of candidates of the sampling density for the Bayesian estimator, which are denoted by BM2SE and BMLE. Thus, in Section 4.2.4, we compare the two Bayesian estimators (i.e, BM2SE and BMLE) with the two traditional estimators (i.e., M2SE and MLE).

4.2.4 Monte Carlo Study

4.2.4.1 Setup of the Model

In the Monte Carlo study, we consider using the artificially simulated data, in which the true data generating process (DGP) is presented in Judge, Hill, Griffiths and Lee (1980, p.156). The DGP is defined as:

$$y_t = \beta_1 + \beta_2 x_{2,t} + \beta_3 x_{3,t} + u_t, \tag{4.12}$$

where u_t, $t = 1, 2, \cdots, n$, are normally and independently distributed with $\mathrm{E}(u_t) = 0$, $\mathrm{E}(u_t^2) = \sigma_t^2$ and,

$$\sigma_t^2 = \exp(\gamma_1 + \gamma_2 x_{2,t}), \qquad \text{for } t = 1, 2, \cdots, n. \tag{4.13}$$

As it is discussed in Judge, Hill, Griffiths and Lee (1980), the parameter values are set to be $(\beta_1, \beta_2, \beta_3, \gamma_1, \gamma_2) = (10, 1, 1, -2, 0.25)$. From (4.12) and (4.13), Judge, Hill,

Table 4.1: The Exogenous Variables $x_{1,t}$ and $x_{2,t}$

t	1	2	3	4	5	6	7	8	9	10
$x_{2,t}$	14.53	15.30	15.92	17.41	18.37	18.83	18.84	19.71	20.01	20.26
$x_{3,t}$	16.74	16.81	19.50	22.12	22.34	17.47	20.24	20.37	12.71	22.98
t	11	12	13	14	15	16	17	18	19	20
$x_{2,t}$	20.77	21.17	21.34	22.91	22.96	23.69	24.82	25.54	25.63	28.73
$x_{3,t}$	19.33	17.04	16.74	19.81	31.92	26.31	25.93	21.96	24.05	25.66

Griffiths and Lee (1980, pp.160 – 165) generated one hundred samples of y with $n = 20$. In the Monte Carlo study, we utilize $x_{2,t}$ and $x_{3,t}$ given in Judge, Hill, Griffiths and Lee (1980, pp.156), which is shown in Table 4.1, and generate G samples of y_t given the X_t for $t = 1, 2, \cdots, n$. That is, we perform G simulation runs for each estimator, where $G = 10^4$ is taken.

The simulation procedure is as follows:

(i) Given γ and $x_{2,t}$ for $t = 1, 2, \cdots, n$, generate random numbers of u_t for $t = 1, 2, \cdots, n$, based on the assumptions: $u_t \sim N(0, \sigma_t^2)$, where $(\gamma_1, \gamma_2) = (-2, 0.25)$ and $\sigma_t^2 = \exp(\gamma_1 + \gamma_2 x_{2,t})$ are taken.

(ii) Given β, $(x_{2,t}, x_{3,t})$ and u_t for $t = 1, 2, \cdots, n$, we obtain a set of data y_t, $t = 1, 2, \cdots, n$, from equation (4.12), where $(\beta_1, \beta_2, \beta_3) = (10, 1, 1)$ is assumed.

(iii) Given (y_t, X_t) for $t = 1, 2, \cdots, n$, perform M2SE, MLE, BM2SE and BMLE discussed in Sections 4.2.2 and 4.2.3 in order to obtain the estimates of $\theta = (\beta, \gamma)$, denoted by $\hat{\theta}$. Note that $\hat{\theta}$ takes $\hat{\theta}_{M2SE}$, $\hat{\theta}_{MLE}$, $\hat{\theta}_{BM2SE}$ and $\hat{\theta}_{BMLE}$.

(iv) Repeat (i) – (iii) G times, where $G = 10^4$ is taken as mentioned above.

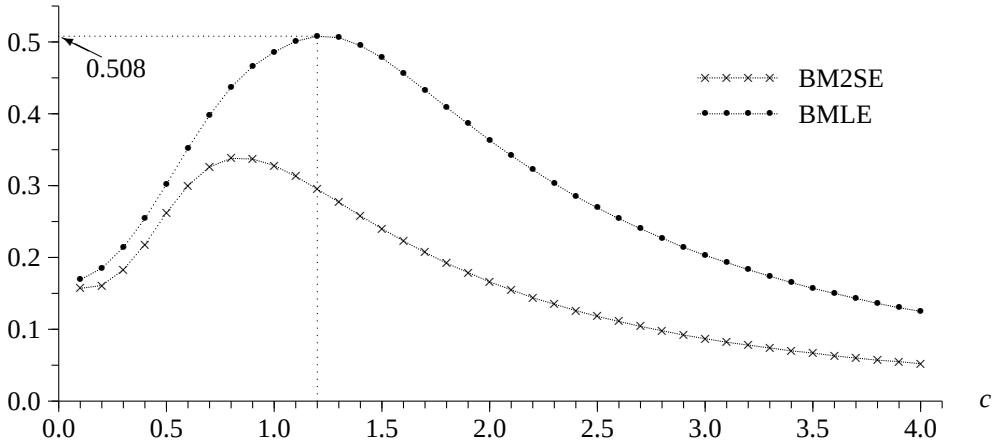
(v) From G estimates of θ, compute the arithmetic average (AVE), the root mean square error (RMSE), the first quartile (25%), the median (50%), the third quartile (75%) and the interquartile range (IR) for each estimator. AVE and RMSE are obtained as follows:

$$\text{AVE} = \frac{1}{G}\sum_{g=1}^{G}\hat{\theta}_j^{(g)}, \qquad \text{RMSE} = \Big(\frac{1}{G}\sum_{g=1}^{G}(\hat{\theta}_j^{(g)} - \theta_j)^2\Big)^{1/2},$$

for $j = 1, 2, \cdots, 5$, where θ_j denotes the jth element of θ and $\hat{\theta}_j^{(g)}$ represents the j-element of $\hat{\theta}$ in the gth simulation run. As mentioned above, $\hat{\theta}$ denotes the estimate of θ, where $\hat{\theta}$ takes $\hat{\theta}_{M2SE}$, $\hat{\theta}_{MLE}$, $\hat{\theta}_{BM2SE}$ and $\hat{\theta}_{BMLE}$.

Choice of (γ^+, Σ^+) and c: For the Bayesian approach, depending on (γ^+, Σ^+) we have BM2SE and BMLE, which denote the Bayesian estimators using the multivariate normal sampling density whose mean and covariance matrix are calibrated on

Figure 4.1: Acceptance Rates in Average: $M = 5000$ and $N = 10^4$



the basis of M2SE or MLE. We consider the following sampling density: $f_*(\gamma) = N(\gamma^+, c^2\Sigma^+)$, where c denotes the tuning parameter and (γ^+, Σ^+) takes $(\gamma_{M2SE}, \Sigma_{M2SE})$ or $(\gamma_{MLE}, \Sigma_{MLE})$. Generally, for choice of the sampling density, the sampling density should not have too large variance and too small variance. Chib and Greenberg (1995) pointed out that if standard deviation of the sampling density is too low, the Metropolis steps are too short and move too slowly within the target distribution; if it is too high, the algorithm almost always rejects and stays in the same place. The sampling density should be chosen so that the chain travels over the support of the target density.

First, we consider choosing (γ^+, Σ^+) and c which maximizes the arithmetic average of the acceptance rates obtained from G simulation runs. The results are in Figure 4.1, where $n = 20$, $M = 5000$, $N = 10^4$, $G = 10^4$ and $c = 0.1, 0.2, \cdots, 4.0$ are taken (choice of N and M is discussed in Appendix of Section 4.2.6). In the case of $(\gamma^+, \Sigma^+) = (\gamma_{MLE}, \Sigma_{MLE})$ and $c = 1.2$, the acceptance rate in average is 0.5078, which gives us the largest one. It is important to reduce positive correlation between γ_i and γ_{i-1} and keep randomness. Therefore, $(\gamma^+, \Sigma^+) = (\gamma_{MLE}, \Sigma_{MLE})$ is adopted, rather than $(\gamma^+, \Sigma^+) = (\gamma_{M2SE}, \Sigma_{M2SE})$, because BMLE has a larger acceptance probability than BM2SE for all c (see Figure 4.1).

As discussed in Section 3.7.5, however, the sampling density with the largest acceptance probability is not necessarily the best choice. We have the result that the optimal standard error should be 1.5 – 2.5 times larger than the standard error which gives us the largest acceptance probability. Here, $(\gamma^+, \Sigma^+) = (\gamma_{MLE}, \Sigma_{MLE})$ and $c = 2$ are taken. When c is larger than 2, both the estimates and their standard errors become stable although here we do not show these facts. Therefore, in this Monte Carlo study, $f_*(\gamma) = N(\gamma_{MLE}, 2^2\Sigma_{MLE})$ is chosen for the sampling density. Hereafter, we compare BMLE with M2SE and MLE (i.e., we do not consider BM2SE anymore).

As for computational CPU time, the case of $n = 20$, $M = 5000$, $N = 10^4$ and

$G = 10^4$ takes about 76 minutes for each of $c = 0.1, 0.2, \cdots, 4.0$ and each of BM2SE and BMLE, where Dual Pentium III 1GHz CPU, Microsoft Windows 2000 Professional Operating System and Open Watcom FORTRAN 77/32 Optimizing Compiler (Version 1.0) are utilized. Note that WATCOM Fortran 77 Compiler is downloaded from `http://www.openwatcom.org/`.

4.2.4.2 Results and Discussion

Through Monte Carlo simulation studies, the Bayesian estimator (i.e., BMLE) is compared with the traditional estimators (i.e., M2SE and MLE).

The arithmetic mean (AVE) and the root mean square error (RMSE) have been usually used in Monte Carlo study. Moreover, for comparison with the standard normal distribution, Skewness and Kurtosis are also computed. Moments of the parameters are needed in the calculation of AVE, RMSE, Skewness and Kurtosis. However, we cannot assure that these moments actually exist. Therefore, in addition to AVE and RMSE, we also present values for quartiles, i.e., the first quartile (25%), median (50%), the third quartile (75%) and the interquartile range (IR). Thus, for each estimator, AVE, RMSE, Skewness, Kurtosis, 25%, 50%, 75% and IR are computed from G simulation runs. The results are given in Table 4.2, where BMLE is compared with M2SE and MLE. The case of $n = 20$, $M = 5000$ and $N = 10^4$ is examined in Table 4.2. A discussion on choice of M and N is given in Appendix 4.2.6, where we examine whether $M = 5000$ and $N = 10^4$ are sufficient.

First, we compare the two traditional estimators, i.e., M2SE and MLE. Judge, Hill, Griffiths and Lee (1980, pp.141–142) indicated that 2SE of γ_1 is inconsistent although 2SE of the other parameters is consistent but asymptotically inefficient. For M2SE, the estimate of γ_1 is modified to be consistent. But M2SE is still asymptotically inefficient while MLE is consistent and asymptotically efficient. Therefore, for γ, MLE should have better performance than M2SE in the sense of efficiency. In Table 4.2, for all the parameters except for IR of β_3, RMSE and IR of MLE are smaller than those of M2SE. For both M2SE and MLE, AVEs of β are close to the true parameter values. Therefore, it might be concluded that M2SE and MLE are unbiased for β even in the case of small sample. However, the estimates of γ are different from the true values for both M2SE and MLE. That is, AVE and 50% of γ_1 are -0.988 and -0.934 for M2SE, and -2.753 and -2.710 for MLE, which are far from the true value -2.0. Similarly, AVE and 50% of γ_2 are 0.199 and 0.200 for M2SE, which are different from the true value 0.25. But 0.272 and 0.273 for MLE are slightly larger than 0.25 and they are close to 0.25. Thus, the traditional estimators work well for the regression coefficients β but not for the heteroscedasticity parameters γ.

Next, the Bayesian estimator (i.e., BMLE) is compared with the traditional ones (i.e., M2SE and MLE). For all the parameters of β, we can find from Table 4.2 that BMLE shows better performance in RMSE and IR than the traditional estimators, because RMSE and IR of BMLE are smaller than those of M2SE and MLE. Furthermore, from AVEs of BMLE, we can see that the heteroscedasticity parameters as well

Table 4.2: The AVE, RMSE and Quartiles: $n = 20$

		β_1	β_2	β_3	γ_1	γ_2
	True Value	10	1	1	−2	0.25
M2SE	AVE	10.064	0.995	1.002	−0.988	0.199
	RMSE	7.537	0.418	0.333	3.059	0.146
	Skewness	0.062	−0.013	−0.010	−0.101	−0.086
	Kurtosis	4.005	3.941	2.988	3.519	3.572
	25%	5.208	0.728	0.778	−2.807	0.113
	50%	10.044	0.995	1.003	−0.934	0.200
	75%	14.958	1.261	1.227	0.889	0.287
	IR	9.751	0.534	0.449	3.697	0.175
MLE	AVE	10.029	0.997	1.002	−2.753	0.272
	RMSE	7.044	0.386	0.332	2.999	0.139
	Skewness	0.081	−0.023	−0.014	0.006	−0.160
	Kurtosis	4.062	3.621	2.965	4.620	4.801
	25%	5.323	0.741	0.775	−4.514	0.189
	50%	10.066	0.998	1.002	−2.710	0.273
	75%	14.641	1.249	1.229	−0.958	0.355
	IR	9.318	0.509	0.454	3.556	0.165
BMLE	AVE	10.034	0.996	1.002	−2.011	0.250
	RMSE	6.799	0.380	0.328	2.492	0.117
	Skewness	0.055	−0.016	−0.013	−0.016	−0.155
	Kurtosis	3.451	3.340	2.962	3.805	3.897
	25%	5.413	0.745	0.778	−3.584	0.176
	50%	10.041	0.996	1.002	−1.993	0.252
	75%	14.538	1.246	1.226	−0.407	0.325
	IR	9.125	0.501	0.448	3.177	0.150

$c = 2.0$, $M = 5000$ and $N = 10^4$ are chosen for BMLE

as the regression coefficients are unbiased in the small sample. Thus, Table 4.2 also shows the evidence that for both β and γ, AVE and 50% of BMLE are very close to the true parameter values. The values of RMSE and IR also indicate that the estimates are concentrated around the AVE and 50%, which are vary close to the true parameter values.

For the regression coefficient β, all of the three estimators are very close to the true parameter values. However, for the heteroscedasticity parameter γ, BMLE shows a good performance but M2SE and MLE are poor.

The larger values of RMSE for the traditional counterparts may be due to "outliers" encountered with the Monte Carlo experiments. This problem is also indicated in Zellner (1971, pp.281). Compared with the traditional counterparts, the Bayesian approach is not characterized by extreme values for posterior modal values.

Now we compare empirical distributions for M2SE, MLE and BMLE in Figures 4.2 – 4.6. For the posterior densities of β_1 (Figure 4.2), β_2 (Figure 4.3), β_3 (Figure 4.4) and γ_1 (Figure 4.5), all of M2SE, MLE and BMLE are almost symmetric (also, see Skewness in Table 4.2). For the posterior density of γ_2 (Figure 4.6), both MLE and

Figure 4.2: Empirical Distributions of β_1

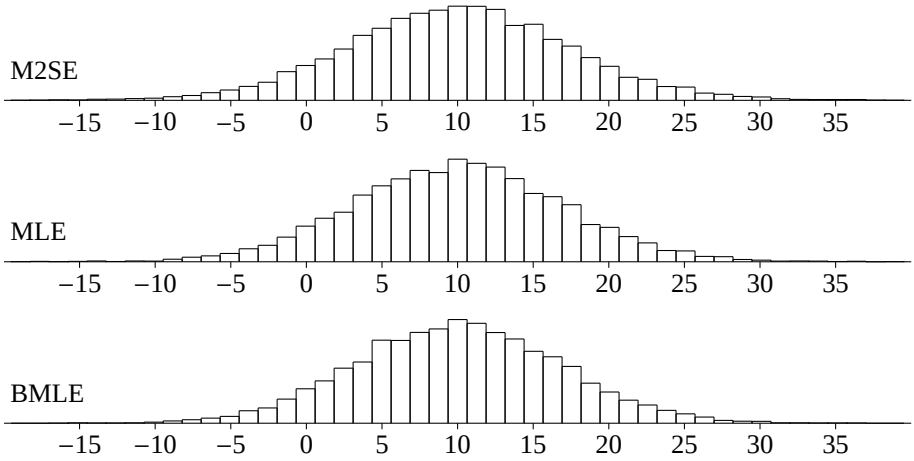


Figure 4.3: Empirical Distributions of β_2

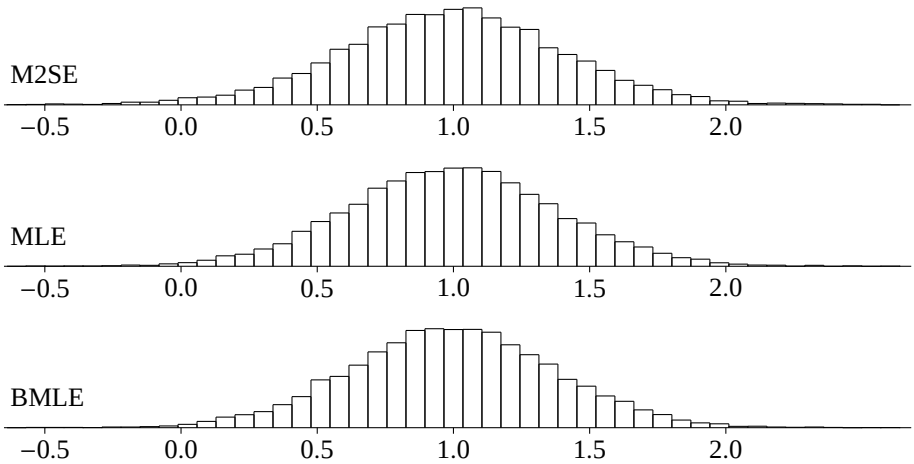


Figure 4.4: Empirical Distributions of β_3

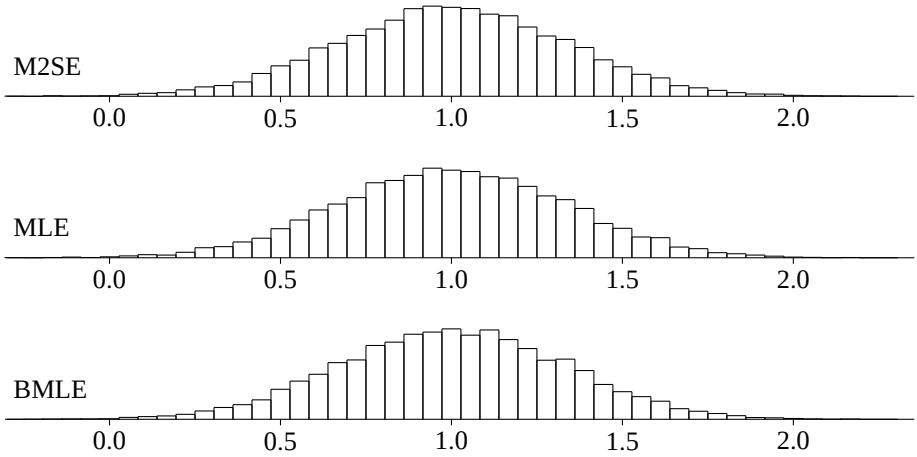


Figure 4.5: Empirical Distributions of γ_1

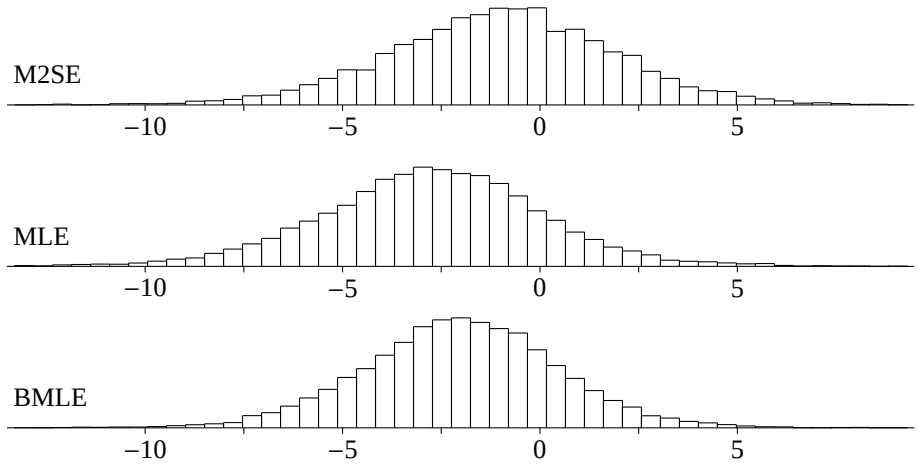
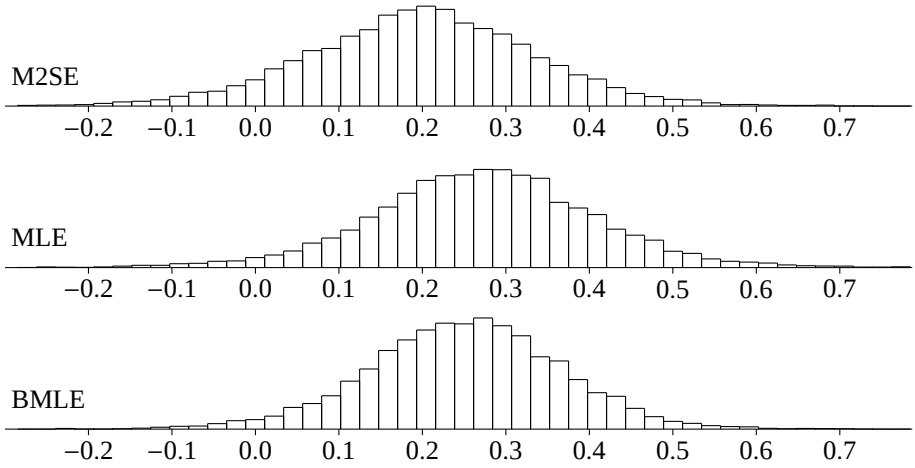


Figure 4.6: Empirical Distributions of γ_2



BMLE are slightly skewed to the left because Skewness of γ_2 in Table 4.2 is negative, while M2SE is almost symmetric. As for Kurtosis, all the empirical distributions except for β_3 have a sharp kurtosis and fat tails, compared with the normal distribution. Especially, for the heteroscedasticity parameters γ_1 and γ_2, MLE has the largest kurtosis of the three. For all figures, location of the empirical distributions indicates whether the estimators are unbiased or not. For β_1 in Figure 4.2, β_2 in Figure 4.3 and β_3 in Figure 4.4, M2SE is biased while MLE and BMLE are distributed around the true value. For γ_1 in Figure 4.5 and γ_2 in Figure 4.6, the empirical distributions of M2SE, MLE and BMLE are quite different. For γ_1 in Figure 4.5, M2SE is located in the right-hand side of the true parameter value, MLE is in the left-hand side, and BMLE is also slightly in the left-hand side. Moreover, for γ_2 in Figure 4.6, M2SE is downward-biased, MLE is overestimated, and BMLE is distributed around the true parameter value.

On the Sample Size *n*: Finally, we examine how the sample size n influences precision of the parameter estimates. Since we utilize the exogenous variable X shown in Judge, Hill, Griffiths and Lee (1980), we cannot examine the case where n is greater than 20. In order to see the effect of the sample size n, here the case of $n = 15$ is compared with that of $n = 20$.

The case $n = 15$ of BMLE is shown in Table 4.3, which should be compared with BMLE in Table 4.2. As a result, all the AVEs are very close to the corresponding true parameter values. Therefore, we can conclude from Tables 4.2 and 4.3 that the Bayesian estimator is unbiased even in the small sample such as $n = 15, 20$. However, RMSE and IR become large as n decreases. That is, for example, RMSEs of β_1, β_2, β_3, γ_1 and γ_2 are given by 6.799, 0.380, 0.328, 2.492 and 0.117 in Table 4.2, and 8.715, 0.455, 0.350, 4.449 and 0.228 in Table 4.3. Thus, we can see that RMSE and

Table 4.3: BMLE: $n = 15$, $c = 2.0$, $M = 5000$ and $N = 10^4$

	β_1	β_2	β_3	γ_1	γ_2
True Value	10	1	1	−2	0.25
AVE	10.060	0.995	1.002	−2.086	0.252
RMSE	8.715	0.455	0.350	4.449	0.228
Skewness	0.014	0.033	−0.064	−0.460	0.308
Kurtosis	3.960	3.667	3.140	4.714	4.604
25%	4.420	0.702	0.772	−4.725	0.107
50%	10.053	0.995	1.004	−1.832	0.245
75%	15.505	1.284	1.237	0.821	0.391
IR	11.085	0.581	0.465	5.547	0.284

IR decrease as n is large.

4.2.5 Summary

In Section 4.2, we have examined the multiplicative heteroscedasticity model discussed by Harvey (1976), where the two traditional estimators are compared with the Bayesian estimator. For the Bayesian approach, we have evaluated the posterior mean by generating random draws from the posterior density, where the Markov chain Monte Carlo methods (i.e., the MH within Gibbs algorithm) are utilized. In the MH algorithm, the sampling density has to be specified. We examine the multivariate normal sampling density, which is the independence chain in the MH algorithm. For mean and variance in the sampling density, we consider using the mean and variance estimated by the two traditional estimators (i.e., M2SE and MLE). The Bayesian estimators with M2SE and MLE are called BM2SE and BMLE in Section 4.2. Through the Monte Carlo studies, the results are summarized as follows:

(i) We compare BM2SE and BMLE with respect to the acceptance rates in the MH algorithm. In this case, BMLE shows higher acceptance rates than BM2SE for all c, which is shown in Figure 4.1. For the sampling density, we utilize the independence chain through Section 4.2. The high acceptance rate implies that the chain travels over the support of the target density. For the Bayesian estimator, therefore, BMLE is preferred to BM2SE. However, note as follows. The sampling density which yields the highest acceptance rate is not necessarily the best choice and the tuning parameter c should be larger than the value which gives us the maximum acceptance rate. Therefore, we have focused on BMLE with $c = 2$ (remember that BMLE with $c = 1.2$ yields the maximum acceptance rate).

(ii) For the traditional estimators (i.e., M2SE and MLE), we have obtained the result that MLE has smaller RMSE than M2SE for all the parameters, because for one reason the M2SE is asymptotically less efficient than the MLE. Furthermore,

Table 4.4: BMLE: $n = 20$ and $c = 2.0$

		β_1	β_2	β_3	γ_1	γ_2
	True Value	10	1	1	−2	0.25
$M = 1000$ $N = 10^4$	AVE	10.028	0.997	1.002	−2.008	0.250
	RMSE	6.807	0.380	0.328	2.495	0.117
	Skewness	0.041	−0.007	−0.012	0.017	−0.186
	Kurtosis	3.542	3.358	2.963	3.950	4.042
	25%	5.413	0.745	0.778	−3.592	0.176
	50%	10.027	0.996	1.002	−1.998	0.252
	75%	14.539	1.245	1.226	−0.405	0.326
	IR	9.127	0.500	0.448	3.187	0.150
$M = 5000$ $N = 5000$	AVE	10.033	0.996	1.002	−2.010	0.250
	RMSE	6.799	0.380	0.328	2.491	0.117
	Skewness	0.059	−0.016	−0.011	−0.024	−0.146
	Kurtosis	3.498	3.347	2.961	3.764	3.840
	25%	5.431	0.747	0.778	−3.586	0.176
	50%	10.044	0.995	1.002	−1.997	0.252
	75%	14.532	1.246	1.225	−0.406	0.326
	IR	9.101	0.499	0.447	3.180	0.149

for M2SE, the estimates of β are unbiased but those of γ are different from the true parameter values (see Table 4.2).

(iii) From Table 4.2, BMLE performs better than the two traditional estimators in the sense of RMSE and IR, because RMSE and IR of BMLE are smaller than those of the traditional ones for all the cases.

(iv) Each empirical distribution is displayed in Figures 4.2 – 4.6. The posterior densities of almost all the estimates are distributed to be symmetric (γ_2 is slightly skewed to the left), but the posterior densities of both the regression coefficients (except for β_3) and the heteroscedasticity parameters have fat tails. Also, see Table 4.2 for skewness and kurtosis.

(v) As for BMLE, the case of $n = 15$ is compared with $n = 20$. The case $n = 20$ has smaller RMSE and IR than $n = 15$, while AVE and 50% are close to the true parameter values for β and γ. Therefore, it might be expected that the estimates of BMLE go to the true parameter values as n is large.

4.2.6 Appendix: Are $M = 5000$ and $N = 10^4$ Sufficient?

In Section 4.2.4, only the case of $(M, N) = (5000, 10^4)$ is examined. In this appendix, we check whether $M = 5000$ and $N = 10^4$ are sufficient. For the burn-in period M, there are some diagnostic tests, which are discussed in Geweke (1992) and Mengersen, Robert and Guihenneuc-Jouyaux (1999). However, since their tests are applicable in the case of one sample path, we cannot utilize them. Because G simulation runs are implemented in Section 4.2.4 (see p.260 for the simulation procedure), we have G test

statistics if we apply the tests. It is difficult to evaluate G testing results at the same time. Therefore, we consider using the alternative approach to see if $M = 5000$ and $N = 10^4$ are sufficient.

For choice of M and N, we consider the following two issues.

(i) Given fixed $M = 5000$, compare $N = 5000$ and $N = 10^4$.

(ii) Given fixed $N = 10^4$, compare $M = 1000$ and $M = 5000$.

(i) examines whether $N = 5000$ is sufficiently large, while (ii) checks whether $M = 1000$ is large enough. If the case of $(M, N) = (5000, 5000)$ is close to that of $(M, N) = (5000, 10^4)$, we can conclude that $N = 5000$ is sufficiently large. Similarly, if the case of $(M, N) = (1000, 10^4)$ is not too different from that of $(M, N) = (5000, 10^4)$, it might be concluded that $M = 1000$ is also sufficient.

The results are in Table 4.4, where AVE, RMSE, Skewness, Kurtosis, 25%, 50%, 75% and IR are shown for each of the regression coefficients and the heteroscedasticity parameters. BMLE in Table 4.2 should be compared with Table 4.4. From Tables 4.2 and 4.4, the three cases, i.e., $(M, N) = (5000, 10^4)$, $(1000, 10^4)$, $(5000, 5000)$, are very close to each other. Therefore, we can conclude that both $M = 1000$ and $N = 5000$ are large enough in the simulation study shown in Section 4.2.4. We take the case of $M = 5000$ and $N = 10^4$ for safety in Section 4.2.4, although we obtain the results that both $M = 1000$ and $N = 5000$ are large enough.

4.3 Autocorrelation Model

In the previous section, we have considered estimating the regression model with the heteroscedastic error term, where the traditional estimators such as MLE and M2SE are compared with the Bayesian estimators. In this section, using both the maximum likelihood estimator and the Bayes estimator, we consider the regression model with the first order autocorrelated error term, where the initial distribution of the autocorrelated error is taken into account. As for the autocorrelated error term, the stationary case is assumed, i.e., the autocorrelation coefficient is assumed to be less than one in absolute value. The traditional estimator (i.e., MLE) is compared with the Bayesian estimator. Utilizing the Gibbs sampler, Chib (1993) discussed the regression model with the autocorrelated error term in a Bayesian framework, where the initial condition of the autoregressive process is not taken into account. In this section, taking into account the initial density, we compare the maximum likelihood estimator and the Bayesian estimator. For the Bayes estimator, the Gibbs sampler and the Metropolis-Hastings algorithm are utilized to obtain random draws of the parameters. As a result, the Bayes estimator is less biased and more efficient than the maximum likelihood estimator. Especially, for the autocorrelation coefficient, the Bayes estimate is much less biased than the maximum likelihood estimate. Accordingly, for the standard error of the estimated regression coefficient, the Bayes estimate is more plausible than the maximum likelihood estimate.

4.3.1 Introduction

In Section 4.3, we consider the regression model with the first order autocorrelated error term, where the error term is assumed to be stationary, i.e., the autocorrelation coefficient is assumed to be less than one in absolute value. The traditional estimator, i.e., the maximum likelihood estimator (MLE), is compared with the Bayes estimator (BE). Utilizing the Gibbs sampler, Chib (1993) and Chib and Greenberg (1994) discussed the regression model with the autocorrelated error term in a Bayesian framework, where the initial condition of the autoregressive process is ignored. Here, taking into account the initial density, we compare MLE and BE, where the Gibbs sampler and the Metropolis-Hastings (MH) algorithm are utilized in BE. As for MLE, it is well known that the autocorrelation coefficient is underestimated in small sample and therefore that variance of the estimated regression coefficient is also biased. See, for example, Andrews (1993) and Tanizaki (2000, 2001). Under this situation, inference on the regression coefficient is not appropriate, because variance of the estimated regression coefficient depends on the estimated autocorrelation coefficient. We show in Section 4.3 that BE is superior to MLE because BEs of both the autocorrelation coefficient and the variance of the error term are closer to the true values, compared with MLEs.

4.3.2 Setup of the Model

Let X_t be a $1 \times k$ vector of exogenous variables and β be a $k \times 1$ parameter vector. Consider the following regression model:

$$y_t = X_t\beta + u_t, \qquad u_t = \rho u_{t-1} + \epsilon_t, \qquad \epsilon_t \sim N(0, \sigma_\epsilon^2),$$

for $t = 1, 2, \cdots, n$, where ϵ_1, ϵ_2, $\cdots$, ϵ_n are assumed to be mutually independently distributed. In this model, the parameter to be estimated is given by $\theta = (\beta, \rho, \sigma_\epsilon^2)$.

The unconditional density of y_t is:

$$f(y_t|\beta, \rho, \sigma_\epsilon^2) = \frac{1}{\sqrt{2\pi\sigma_\epsilon^2/(1-\rho^2)}} \exp\Big(-\frac{1}{2\sigma_\epsilon^2/(1-\rho^2)}(y_t - X_t\beta)^2\Big).$$

Let Y_t be the information set up to time t, i.e., $Y_t = \{y_t, y_{t-1}, \cdots, y_1\}$. The conditional density of y_t given Y_{t-1} is:

$$\begin{aligned} f(y_t|Y_{t-1}, \beta, \rho, \sigma_\epsilon^2) &= f(y_t|y_{t-1}, \beta, \rho, \sigma_\epsilon^2) \\ &= \frac{1}{\sqrt{2\pi\sigma_\epsilon^2}} \exp\Big(-\frac{1}{2\sigma_\epsilon^2}((y_t - \rho y_{t-1}) - (X_t - \rho X_{t-1})\beta)^2\Big). \end{aligned}$$

Therefore, the joint density of Y_n, i.e., the likelihood function, is given by :

$$\begin{aligned} f(Y_n|\beta, \rho, \sigma_\epsilon^2) &= f(y_1|\beta, \rho, \sigma_\epsilon^2) \prod_{t=2}^{n} f(y_t|Y_{t-1}, \beta, \rho, \sigma_\epsilon^2) \\ &= (2\pi\sigma_\epsilon^2)^{-n/2}(1-\rho^2)^{1/2} \exp\Big(-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^{n}(y_t^* - X_t^*\beta)^2\Big), \end{aligned} \tag{4.14}$$

where y_t^* and X_t^* represent the following transformed variables:

$$y_t^* = y_t^*(\rho) = \begin{cases} \sqrt{1-\rho^2}y_t, & \text{for } t = 1, \\ y_t - \rho y_{t-1}, & \text{for } t = 2, 3, \cdots, n, \end{cases}$$

$$X_t^* = X_t^*(\rho) = \begin{cases} \sqrt{1-\rho^2}X_t, & \text{for } t = 1, \\ X_t - \rho X_{t-1}, & \text{for } t = 2, 3, \cdots, n, \end{cases}$$

which depend on the autocorrelation coefficient ρ.

Maximum Likelihood Estimator: We have shown above that the likelihood function is given by equation (4.14). Maximizing equation (4.14) with respect to β and σ_ϵ^2, we obtain the following expressions:

$$\begin{aligned} \hat{\beta} \equiv \hat{\beta}(\rho) &= (\sum_{t=1}^{n} X_t^{*\prime} X_t^*)^{-1} \sum_{t=1}^{n} X_t^{*\prime} y_t^*, \\ \hat{\sigma}_\epsilon^2 \equiv \hat{\sigma}_\epsilon^2(\rho) &= \frac{1}{n}\sum_{t=1}^{n}(y_t^* - X_t^*\hat{\beta})^2. \end{aligned} \tag{4.15}$$

By substituting $\hat{\beta}$ and $\hat{\sigma}_\epsilon^2$ into β and σ_ϵ^2 in equation (4.14), we have the concentrated likelihood function:

$$f(Y_n|\hat{\beta}, \rho, \hat{\sigma}_\epsilon^2) = \left(2\pi\hat{\sigma}_\epsilon^2(\rho)\right)^{-n/2}(1-\rho^2)^{1/2}\exp(-\frac{n}{2}), \tag{4.16}$$

which is a function of ρ. Equation (4.16) has to be maximized with respect to ρ. In the next section, we obtain the maximum likelihood estimate of ρ by a simple grid search, in which the concentrated likelihood function (4.16) is maximized by changing the parameter value of ρ by 0.0001 in the interval between -0.9999 and 0.9999. Once the solution of ρ, denoted by $\hat{\rho}$, is obtained, $\hat{\beta}(\hat{\rho})$ and $\hat{\sigma}_\epsilon^2(\hat{\rho})$ lead to the maximum likelihood estimates of β and σ_ϵ^2. Hereafter, $\hat{\beta}$, $\hat{\sigma}_\epsilon^2$ and $\hat{\rho}$ are taken as the maximum likelihood estimates of β, σ_ϵ^2 and ρ, i.e., $\hat{\beta}(\hat{\rho})$ and $\hat{\sigma}_\epsilon^2(\hat{\rho})$ are simply written as $\hat{\beta}$ and $\hat{\sigma}_\epsilon^2$.

Variance of the estimate of $\theta = (\beta', \sigma^2, \rho)'$ is asymptotically given by: $\mathrm{V}(\hat{\theta}) = I^{-1}(\theta)$, where $I(\theta)$ denotes the information matrix, which is represented as:

$$I(\theta) = -\mathrm{E}\left(\frac{\partial^2 \log f(Y_n|\theta)}{\partial\theta\partial\theta'}\right).$$

Therefore, variance of $\hat{\beta}$ is given by $\mathrm{V}(\hat{\beta}) = \sigma^2(\sum_{t=1}^{n} X_t^{*\prime} X_t^*)^{-1}$ in large sample, where ρ in X_t^* is replaced by $\hat{\rho}$, i.e., $X_t^* = X_t^*(\hat{\rho})$. For example, suppose that X_t^* has a tendency to rise over time t and that we have $\rho > 0$. If ρ is underestimated, then $\mathrm{V}(\hat{\beta})$ is also underestimated, which yields incorrect inference on the regression coefficient β. Thus, unless ρ is properly estimated, the estimate of $\mathrm{V}(\hat{\beta})$ is also biased. In large sample, $\hat{\rho}$ is

a consistent estimator of ρ and therefore $V(\hat{\beta})$ is not biased. However, in small sample, since it is known that $\hat{\rho}$ is underestimated (see, for example, Andrews (1993), Tanizaki (2000, 2001)), clearly $V(\hat{\beta})$ is also underestimated. In addition to $\hat{\rho}$, the estimate of σ^2 also influences inference of β, because we have $V(\hat{\beta}) = \sigma^2(\sum_{t=1}^n X_t^{*\prime} X_t^*)^{-1}$ as mentioned above. If σ^2 is underestimated, the estimated variance of β is also underestimated. $\hat{\sigma}^2$ is a consistent estimator of σ^2 in large sample, but it is appropriate to consider that $\hat{\sigma}^2$ is biased in small sample, because $\hat{\sigma}^2$ is a function of $\hat{\rho}$ as in (4.15). Therefore, the biased estimate of ρ gives us the serious problem on inference of β.

Bayesian Estimator: We assume that the prior density functions of β, ρ and σ_ϵ^2 are the following noninformative priors:

$$f_\beta(\beta) \propto \text{constant}, \qquad \text{for } -\infty < \beta < \infty, \tag{4.17}$$

$$f_\rho(\rho) \propto \text{constant}, \qquad \text{for } -1 < \rho < 1, \tag{4.18}$$

$$f_{\sigma_\epsilon}(\sigma_\epsilon^2) \propto \frac{1}{\sigma_\epsilon^2}, \qquad \text{for } 0 < \sigma_\epsilon^2 < \infty. \tag{4.19}$$

In equation (4.18), theoretically we should have $-1 < \rho < 1$. As for the prior density of σ_ϵ^2, since we consider that $\log \sigma_\epsilon^2$ has the flat prior for $-\infty < \log \sigma_\epsilon^2 < \infty$, we obtain $f_{\sigma_\epsilon}(\sigma_\epsilon^2) \propto 1/\sigma_\epsilon^2$. Note that in Section 4.2 the first element of the heteroscedasticity parameter γ is also assumed to be diffuse, where it is formulated as the logarithm of variance of the error term, i.e., $\log \sigma_\epsilon^2$.

Combining the four densities (4.14) and (4.17) – (4.19), the posterior density function of β, ρ and σ_ϵ^2, denoted by $f_{\beta\rho\sigma_\epsilon}(\beta, \rho, \sigma_\epsilon^2|Y_n)$, is represented as follows:

$$\begin{aligned} &f_{\beta\rho\sigma_\epsilon}(\beta, \rho, \sigma_\epsilon^2|Y_n) \\ &\quad \propto f(Y_n|\beta, \rho, \sigma_\epsilon^2) f_\beta(\beta) f_\rho(\rho) f_{\sigma_\epsilon}(\sigma_\epsilon^2) \\ &\quad \propto (\sigma_\epsilon^2)^{-(n/2+1)}(1-\rho^2)^{1/2} \exp\Big(-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^n (y_t^* - X_t^*\beta)^2\Big). \end{aligned} \tag{4.20}$$

We want to have random draws of β, ρ and σ_ϵ^2 given Y_n. However, it is not easy to generate random draws of β, ρ and σ_ϵ^2 from $f_{\beta\rho\sigma_\epsilon}(\beta, \rho, \sigma_\epsilon^2|Y_n)$. Therefore, we perform the Gibbs sampler in this problem. According to the Gibbs sampler, we can sample from the posterior density function (4.20), using the three conditional distributions $f_{\beta|\rho\sigma_\epsilon}(\beta|\rho, \sigma_\epsilon^2, Y_n)$, $f_{\rho|\beta\sigma_\epsilon}(\rho|\beta, \sigma_\epsilon^2, Y_n)$ and $f_{\sigma_\epsilon|\beta\rho}(\sigma_\epsilon^2|\beta, \rho, Y_n)$, which are proportional to $f_{\beta\rho\sigma}(\beta, \rho, \sigma^2|Y_n)$ and are obtained as follows:

- $f_{\beta|\rho\sigma_\epsilon}(\beta|\rho, \sigma_\epsilon^2, Y_n)$ is given by:

$$\begin{aligned} &f_{\beta|\rho\sigma_\epsilon}(\beta|\rho, \sigma_\epsilon^2, Y_n) \\ &\quad \propto f_{\beta\rho\sigma_\epsilon}(\beta, \rho, \sigma_\epsilon^2|Y_n) \propto \exp\Big(-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^n (y_t^* - X_t^*\beta)^2\Big) \end{aligned}$$

$$= \exp\Big(-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^{n}\Big((y_t^* - X_t^*\hat{\beta}) - X_t(\beta-\hat{\beta})\Big)^2\Big)$$

$$= \exp\Big(-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^{n}(y_t^* - X_t^*\hat{\beta})^2 - \frac{1}{2\sigma_\epsilon^2}(\beta-\hat{\beta})'(\sum_{t=1}^{n}X_t^{*\prime}X_t^*)(\beta-\hat{\beta})\Big)$$

$$\propto \exp\Big(-\frac{1}{2}(\beta-\hat{\beta})'(\frac{1}{\sigma_\epsilon^2}\sum_{t=1}^{n}X_t^{*\prime}X_t^*)(\beta-\hat{\beta})\Big), \tag{4.21}$$

which indicates that $\beta \sim N(\hat{\beta}, \sigma_\epsilon^2(\sum_{t=1}^n X_t^{*\prime}X_t^*)^{-1})$, where $\hat{\beta}$ represents the OLS estimate, i.e., $\hat{\beta} = (\sum_{t=1}^n X_t^{*\prime}X_t^*)^{-1}(\sum_{t=1}^n X_t^{*\prime}y_t^*)$. Thus, (4.21) implies that β can be sampled from the multivariate normal distribution with mean $\hat{\beta}$ and variance $\sigma_\epsilon^2(\sum_{t=1}^n X_t^{*\prime}X_t^*)^{-1}$.

- $f_{\rho|\beta\sigma_\epsilon}(\rho|\beta,\sigma_\epsilon^2,Y_n)$ is obtained as:

$$f_{\rho|\beta\sigma_\epsilon}(\rho|\beta,\sigma_\epsilon^2,Y_n) \propto f_{\beta\rho\sigma_\epsilon}(\beta,\rho,\sigma_\epsilon^2|Y_n)$$
$$\propto (1-\rho^2)^{1/2}\exp\Big(-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^{n}\Big(y_t^* - X_t^*\beta\Big)^2\Big), \tag{4.22}$$

for $-1 < \rho < 1$, which cannot be represented in a known distribution. Note that $y_t^* = y_t^*(\rho)$ and $X_t^* = X_t^*(\rho)$. Sampling from (4.22) is implemented by the MH algorithm. A detail discussion on sampling will be given later.

- $f_{\sigma_\epsilon|\beta\rho}(\sigma_\epsilon^2|\beta,\rho,Y_n)$ is represented as:

$$f_{\sigma_\epsilon|\beta\rho}(\sigma_\epsilon^2|\beta,\rho,Y_n) \propto f_{\beta\rho\sigma_\epsilon}(\beta,\rho,\sigma_\epsilon^2|Y_n)$$
$$\propto \frac{1}{(\sigma_\epsilon^2)^{n/2+1}}\exp\Big(-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^{n}(y_t^* - X_t^*\beta)^2\Big), \tag{4.23}$$

which is written as follows: $\sigma_\epsilon^2 \sim IG(n/2,\ 2/\sum_{t=1}^n \epsilon_t^2)$, or equivalently, $1/\sigma_\epsilon^2 \sim G(n/2,\ 2/\sum_{t=1}^n \epsilon_t^2)$, where $\epsilon_t = y_t^* - X_t^*\beta$. See Section 2.2.6 (p.97) for the inverse gamma distribution.

Thus, in order to generate random draws of β, ρ and σ_ϵ^2 from the posterior density $f_{\beta\rho\sigma_\epsilon}(\beta,\rho,\sigma_\epsilon^2|Y_n)$, the following procedures have to be taken:

(i) Let β_i, ρ_i and $\sigma_{\epsilon,i}^2$ be the ith random draws of β, ρ and σ_ϵ^2. Take the initial values of $(\beta,\ \rho,\ \sigma_\epsilon^2)$ as $(\beta_{-M},\ \rho_{-M},\ \sigma_{\epsilon,-M}^2)$.

(ii) From equation (4.21), generate β_i given ρ_{i-1}, $\sigma_{\epsilon,i-1}^2$ and Y_n, using $\beta \sim N(\hat{\beta},\ \sigma_{\epsilon,i-1}^2(\sum_{t=1}^n X_t^{*\prime}X_t^*)^{-1})$, where $\hat{\beta} = (\sum_{t=1}^n X_t^{*\prime}X_t^*)^{-1}(\sum_{t=1}^n X_t^{*\prime}y_t^*)$, $y_t^* = y_t^*(\rho_{i-1})$ and $X_t^* = X_t^*(\rho_{i-1})$.

(iii) From equation (4.22), generate ρ_i given β_i, $\sigma_{\epsilon,i-1}^2$ and Y_n. Since it is not easy to generate random draws from (4.21), the Metropolis-Hastings algorithm is utilized, which is implemented as follows:

(a) Generate ρ^* from the uniform distribution between -1 and 1, which implies that the sampling density of ρ is given by $f_*(\rho|\rho_{i-1}) = 1/2$ for $-1 < \rho < 1$. Compute the acceptance probability $\omega(\rho_{i-1}, \rho^*)$, which is defined as:

$$\omega(\rho_{i-1}, \rho^*) = \min\left(\frac{f_{\rho|\beta\sigma_\epsilon}(\rho^*|\beta_i, \sigma^2_{\epsilon,i-1}, Y_n)/\, f_*(\rho^*|\rho_{i-1})}{f_{\rho|\beta\sigma_\epsilon}(\rho_{i-1}|\beta_i, \sigma^2_{\epsilon,i-1}, Y_n)/\, f_*(\rho_{i-1}|\rho^*)},\ 1 \right)$$
$$= \min\left(\frac{f_{\rho|\beta\sigma_\epsilon}(\rho^*|\beta_i, \sigma^2_{\epsilon,i-1}, Y_n)}{f_{\rho|\beta\sigma_\epsilon}(\rho_{i-1}|\beta_i, \sigma^2_{\epsilon,i-1}, Y_n)},\ 1 \right).$$

(b) Set $\rho_i = \rho^*$ with probability $\omega(\rho_{i-1}, \rho^*)$ and $\rho_i = \rho_{i-1}$ otherwise.

(iv) From equation (4.23), generate $\sigma^2_{\epsilon,i}$ given β_i, ρ_i and Y_n, using $1/\sigma^2_\epsilon \sim G(n/2,\ 2/\sum_{t=1}^n u_t^2)$, where $u_t = y_t^* - X_t^*\beta$, $y_t^* = y_t^*(\rho_i)$ and $X_t^* = X_t^*(\rho_i)$.

(v) Repeat Steps (ii) – (iv) for $i = -M+1, -M+2, \cdots, N$, where M indicates the burn-in period.

Repetition of Steps (ii) – (iv) corresponds to the Gibbs sampler. For sufficiently large M, we have the following results:

$$\frac{1}{N}\sum_{i=1}^N g(\beta_i) \longrightarrow \mathrm{E}(g(\beta)),$$
$$\frac{1}{N}\sum_{i=1}^N g(\rho_i) \longrightarrow \mathrm{E}(g(\rho)),$$
$$\frac{1}{N}\sum_{i=1}^N g(\sigma^2_{\epsilon,i}) \longrightarrow \mathrm{E}(g(\sigma^2_\epsilon)),$$

where $g(\cdot)$ is a function, typically $g(x) = x$ or $g(x) = x^2$. We define the Bayesian estimates of β, ρ and σ^2_ϵ as $\widetilde{\beta} \equiv (1/N)\sum_{i=1}^N \beta_i$, $\widetilde{\rho} \equiv (1/N)\sum_{i=1}^N \rho_i$ and $\widetilde{\sigma}^2_\epsilon \equiv (1/N)\sum_{i=1}^N \sigma^2_{\epsilon,i}$, respectively.

Thus, using both the Gibbs sampler and the MH algorithm, we have shown that we can sample from $f_{\beta\rho\sigma_\epsilon}(\beta, \rho, \sigma^2_\epsilon|Y_n)$. See, for example, Bernardo and Smith (1994), Carlin and Louis (1996), Chen, Shao and Ibrahim (2000), Gamerman (1997), Robert and Casella (1999) and Smith and Roberts (1993) for the Gibbs sampler and the MH algorithm.

4.3.3 Monte Carlo Experiments

For the exogenous variables, again we take the data used in Section 4.2, in which the true data generating process (DGP) is presented in Judge, Hill, Griffiths and Lee (1980, p.156). As in equation (4.12), the DGP is defined as:

$$y_t = \beta_1 + \beta_2 x_{2,t} + \beta_3 x_{3,t} + u_t, \qquad u_t = \rho u_{t-1} + \epsilon_t, \tag{4.24}$$

where ϵ_t, $t = 1, 2, \cdots, n$, are normally and independently distributed with $\mathrm{E}(\epsilon_t) = 0$ and $\mathrm{E}(\epsilon_t^2) = \sigma_\epsilon^2$. As in Judge, Hill, Griffiths and Lee (1980), the parameter values are set to be $(\beta_1, \beta_2, \beta_3) = (10, 1, 1)$. We utilize $x_{2,t}$ and $x_{3,t}$ given in Judge, Hill, Griffiths and Lee (1980, pp.156), which is shown in Table 4.1, and generate G samples of y_t given the X_t for $t = 1, 2, \cdots, n$. That is, we perform G simulation runs for each estimator, where $G = 10^4$ is taken.

The simulation procedure is as follows:

(i) Given ρ, generate random numbers of u_t for $t = 1, 2, \cdots, n$, based on the assumptions: $u_t = \rho u_{t-1} + \epsilon_t$ and $\epsilon_t \sim N(0, 1)$.

(ii) Given β, $(x_{2,t}, x_{3,t})$ and u_t for $t = 1, 2, \cdots, n$, we obtain a set of data y_t, $t = 1, 2, \cdots, n$, from equation (4.24), where $(\beta_1, \beta_2, \beta_3) = (10, 1, 1)$ is assumed.

(iii) Given (y_t, X_t) for $t = 1, 2, \cdots, n$, obtain the estimates of $\theta = (\beta, \rho, \sigma_\epsilon^2)$ by the maximum likelihood estimation (MLE) and the Bayesian estimation (BE) discussed in Sections 4.3.2, which are denoted by $\hat{\theta}$ and $\widetilde{\theta}$, respectively.

(iv) Repeat (i) – (iii) G times, where $G = 10^4$ is taken.

(v) From G estimates of θ, compute the arithmetic average (AVE), the standard error (SER), the root mean square error (RMSE), the skewness (Skewness), the kurtosis (Kurtosis), and the 5, 10, 25, 50, 75, 90 and 95 percent points (5%, 10%, 25%, 50%, 75%, 90% and 95%) for each estimator. For the maximum likelihood estimator (MLE), we compute:

$$\mathrm{AVE} = \frac{1}{G}\sum_{g=1}^{G} \hat{\theta}_j^{(g)}, \qquad \mathrm{RMSE} = \Big(\frac{1}{G}\sum_{g=1}^{G} (\hat{\theta}_j^{(g)} - \theta_j)^2\Big)^{1/2},$$

for $j = 1, 2, \cdots, 5$, where θ_j denotes the jth element of θ and $\hat{\theta}_j^{(g)}$ represents the jth element of $\hat{\theta}$ in the gth simulation run. For the Bayesian estimator (BE), $\hat{\theta}$ in the above equations is replaced by $\widetilde{\theta}$, and AVE and RMSE are obtained.

(vi) Repeat (i) – (v) for $\rho = -0.99, -0.98, \cdots, 0.99$.

Thus, in Section 4.3.3, we compare the Bayesian estimator (BE) with the maximum likelihood estimator (MLE) through Monte Carlo studies.

In Figures 4.7 and 4.8, we focus on the estimates of the autocorrelation coefficient ρ. In Figure 4.7 we draw the relationship between ρ and $\hat{\rho}$, where $\hat{\rho}$ denotes the arithmetic average of the 10^4 MLEs, while in Figure 4.8 we display the relationship between ρ and $\widetilde{\rho}$, where $\widetilde{\rho}$ indicates the arithmetic average of the 10^4 BEs. In the two figures the cases of $n = 10, 15, 20$ are shown, and $(M, N) = (5000, 10^4)$ is taken in Figure 4.8 (we will discuss later about M and N). If the relationship between ρ and $\hat{\rho}$ (or $\widetilde{\rho}$) lies on the 45° degree line, we can conclude that MLE (or BE) of ρ is unbiased. However, from the two figures, both estimators are biased. Take an example of $\rho = 0.9$ in Figures 4.7 and 4.8. When the true value is $\rho = 0.9$, the arithmetic averages of 10^4 MLEs are given by 0.142 for $n = 10$, 0.422 for $n = 15$ and 0.559 for $n = 20$ (see

Figure 4.7: The Arithmetic Average from the 10^4 MLE's of AR(1) Coeff.

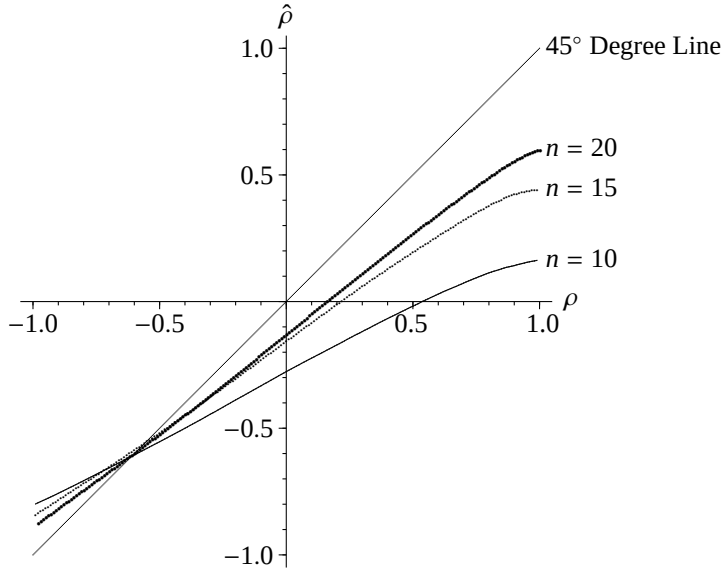


Figure 4.8: The Arithmetic Average from the 10^4 BE's of AR(1) Coeff.
——— $M = 5000$ and $N = 10^4$ ———

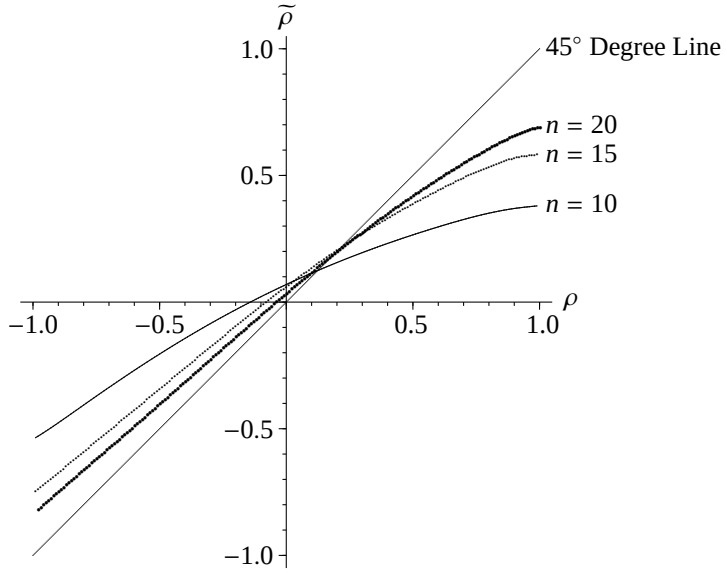


Table 4.5: MLE: $n = 20$ and $\rho = 0.9$

Parameter	β_1	β_2	β_3	ρ	σ_ϵ^2
True Value	10	1	1	0.9	1
AVE	10.012	0.999	1.000	0.559	0.752
SER	3.025	0.171	0.053	0.240	0.276
RMSE	3.025	0.171	0.053	0.417	0.372
Skewness	0.034	−0.045	−0.008	−1.002	0.736
Kurtosis	2.979	3.093	3.046	4.013	3.812
5%	5.096	0.718	0.914	0.095	0.363
10%	6.120	0.785	0.933	0.227	0.426
25%	7.935	0.883	0.965	0.426	0.550
50%	10.004	0.999	1.001	0.604	0.723
75%	12.051	1.115	1.036	0.740	0.913
90%	13.913	1.217	1.068	0.825	1.120
95%	15.036	1.274	1.087	0.863	1.255

Table 4.6: BE with $M = 5000$ and $N = 10^4$: $n = 20$ and $\rho = 0.9$

Parameter	β_1	β_2	β_3	ρ	σ_ϵ^2
True Value	10	1	1	0.9	1
AVE	10.010	0.999	1.000	0.661	1.051
SER	2.782	0.160	0.051	0.188	0.380
RMSE	2.782	0.160	0.051	0.304	0.384
Skewness	0.008	−0.029	−0.022	−1.389	0.725
Kurtosis	3.018	3.049	2.942	5.391	3.783
5%	5.498	0.736	0.915	0.285	0.515
10%	6.411	0.798	0.934	0.405	0.601
25%	8.108	0.891	0.966	0.572	0.776
50%	10.018	1.000	1.001	0.707	1.011
75%	11.888	1.107	1.036	0.799	1.275
90%	13.578	1.205	1.067	0.852	1.555
95%	14.588	1.258	1.085	0.875	1.750

Table 4.7: BE with $M = 5000$ and $N = 5000$: $n = 20$ and $\rho = 0.9$

Parameter	β_1	β_2	β_3	ρ	σ_ϵ^2
True Value	10	1	1	0.9	1
AVE	10.011	0.999	1.000	0.661	1.051
SER	2.785	0.160	0.051	0.189	0.380
RMSE	2.785	0.160	0.052	0.305	0.384
Skewness	0.004	−0.027	−0.022	−1.390	0.723
Kurtosis	3.028	3.056	2.938	5.403	3.776
5%	5.500	0.736	0.915	0.285	0.514
10%	6.402	0.797	0.934	0.405	0.603
25%	8.117	0.891	0.966	0.572	0.775
50%	10.015	1.000	1.001	0.707	1.011
75%	11.898	1.107	1.036	0.799	1.277
90%	13.612	1.205	1.066	0.852	1.559
95%	14.600	1.257	1.085	0.876	1.747

Table 4.8: BE with $M = 1000$ and $N = 10^4$: $n = 20$ and $\rho = 0.9$

Parameter	β_1	β_2	β_3	ρ	σ_ϵ^2
True Value	10	1	1	0.9	1
AVE	10.010	0.999	1.000	0.661	1.051
SER	2.783	0.160	0.051	0.188	0.380
RMSE	2.783	0.160	0.051	0.304	0.384
Skewness	0.008	−0.029	−0.021	−1.391	0.723
Kurtosis	3.031	3.055	2.938	5.404	3.774
5%	5.495	0.736	0.915	0.284	0.514
10%	6.412	0.797	0.935	0.404	0.602
25%	8.116	0.891	0.966	0.573	0.774
50%	10.014	1.000	1.001	0.706	1.011
75%	11.897	1.107	1.036	0.799	1.275
90%	13.587	1.204	1.067	0.852	1.558
95%	14.588	1.257	1.085	0.876	1.746

Figure 4.7), while those of 10^4 BEs are 0.369 for $n = 10$, 0.568 for $n = 15$ and 0.661 for $n = 20$ (see Figure 4.8). As n increases the estimators are less biased, because it is shown that MLE gives us the consistent estimators. Comparing BE and MLE, BE is less biased than MLE in the small sample, because BE is closer to the 45° degree line than MLE. Especially, as ρ goes to one, the difference between BE and MLE becomes quite large.

Tables 4.5 – 4.8 represent the basic statistics such as arithmetic average, standard error, root mean square error, skewness, kurtosis and percent points, which are computed from $G = 10^4$ simulation runs, where the case of $n = 20$ and $\rho = 0.9$ is examined. Table 4.5 is based on the MLEs while Tables 4.6 – 4.8 are obtained from the BEs. To check whether M and N are enough large, Tables 4.6 – 4.8 are shown for BE. Comparison between Tables 4.6 and 4.7 shows whether $N = 5000$ is large enough and we can see from Tables 4.6 and 4.8 whether the burn-in period $M = 1000$ is large enough. We can conclude that $N = 5000$ is enough if Table 4.6 is very close to Table 4.7 and that $M = 1000$ is enough if Table 4.6 is close to Table 4.8. The difference between Tables 4.6 and 4.7 is at most 0.034 (see 90% in β_1) and that between Tables 4.6 and 4.8 is less than or equal to 0.013 (see Kurtosis in β_1). Thus, all the three tables are very close to each other. Therefore, we can conclude that $(M, N) = (1000, 5000)$ is enough. For safety, hereafter we focus on the case of $(M, N) = (5000, 10^4)$.

We compare Tables 4.5 and 4.6. Both MLE and BE give us the unbiased estimators of regression coefficients β_1, β_2 and β_3, because the arithmetic averages from the 10^4 estimates of β_1, β_2 and β_3, (i.e., AVE in the tables) are very close to the true parameter values, which are set to be $(\beta_1, \beta_2, \beta_3) = (10, 1, 1)$. However, in the SER and RMSE criteria, BE is better than MLE, because SER and RMSE of BE are smaller than those of MLE. From Skewness and Kurtosis in the two tables, we can see that the empirical distributions of MLE and BE of $(\beta_1, \beta_2, \beta_3)$ are very close to the normal distribution. Remember that the skewness and kurtosis of the normal distribution are given by zero

Figure 4.9: Empirical Distributions of β_1

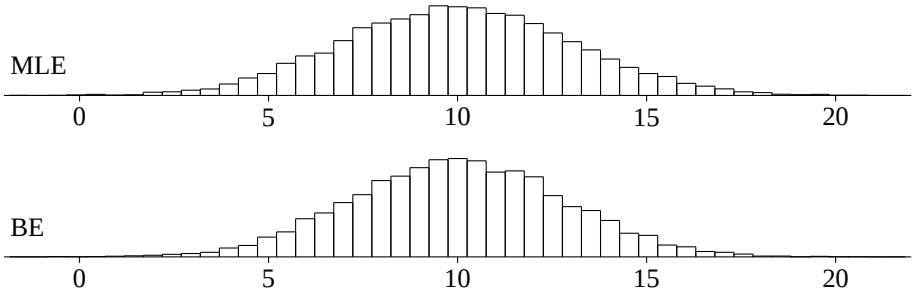


Figure 4.10: Empirical Distributions of β_2

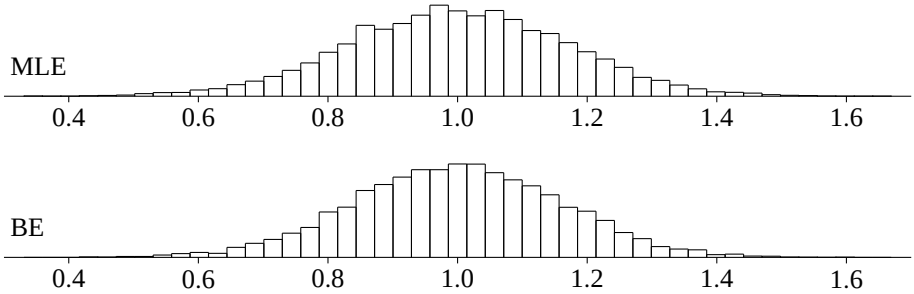


Figure 4.11: Empirical Distributions of β_3

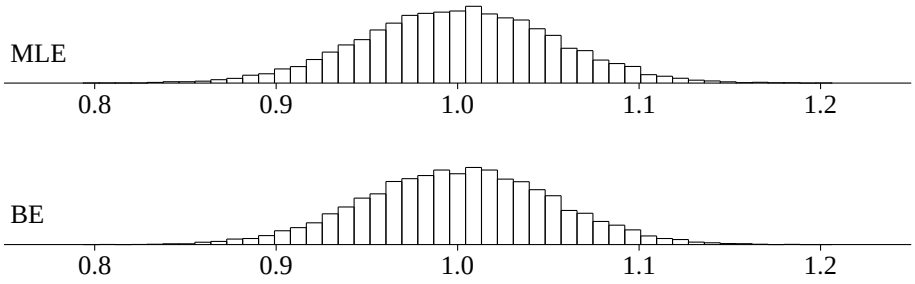


Figure 4.12: Empirical Distributions of ρ

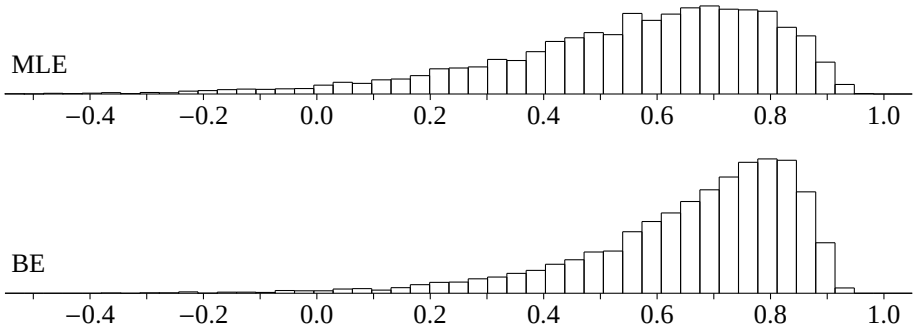
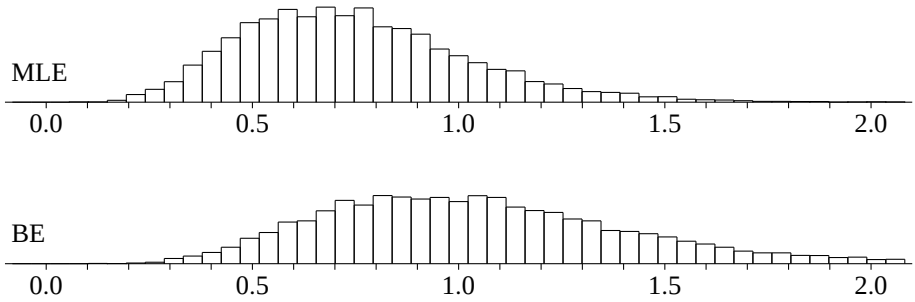


Figure 4.13: Empirical Distributions of σ_ϵ^2



and three, respectively.

As for σ_ϵ^2, AVE of BE is closer to the true value than that of MLE, because AVE of MLE is 0.752 (see Table 4.5) and that of BE is 1.051 (see Table 4.6). However, in the SER and RMSE criteria, MLE is superior to BE, since SER and RMSE of MLE are given by 0.276 and 0.372 (see Table 4.5) while those of BE are 0.380 and 0.384 (see Table 4.6). The empirical distribution obtained from 10^4 estimates of σ_ϵ^2 is skewed to the right (Skewness is positive for both MLE and BE) and has a larger kurtosis than the normal distribution because Kurtosis is greater than three for both tables.

For ρ, AVE of MLE is 0.559 (Table 4.5) and that of BE is given by 0.661 (Table 4.6). As it is also seen in Figures 4.7 and 4.8, BE is less biased than MLE from the AVE criterion. Moreover, SER and RMSE of MLE are 0.240 and 0.417, while those of BE are 0.188 and 0.304. Therefore, BE is more efficient than MLE. Thus, in the AVE, SER and RMSE criteria, BE is superior to MLE with respect to ρ. The empirical distributions of MLE and BE of ρ are skewed to the left because Skewness is negative, which value is given by -1.002 in Table 4.5 and -1.389 in Table 4.6. We can see that MLE is less skewed than BE. For Kurtosis, both MLE and BE of ρ are

greater than three and therefore the empirical distributions of the estimates of ρ have fat tails, compared with the normal distribution. Since Kurtosis in Table 4.6 is 5.391 and that in Table 4.5 is 4.013, the empirical distribution of BE has more kurtosis than that of MLE.

Figures 4.9 – 4.13 correspond to the empirical distributions for each parameter, which are constructed from the G estimates used in Tables 4.5 and 4.6. As we can see from Skewness and Kurtosis in Tables 4.5 and 4.6, $\hat{\beta}_i$ and $\widetilde{\beta}_i$, $i = 1, 2, 3$, are very similar to normal distributions in Figures 4.9 – 4.11. For β_i, $i = 1, 2, 3$, the empirical distributions of MLE have the almost same centers as those of BE, but the empirical distributions of MLE are more widely distributed than those of BE. We can also observe these facts from AVEs and SERs in Tables 4.5 and 4.6. In Figure 4.12, the empirical distribution of $\hat{\rho}$ is quite different from that of $\widetilde{\rho}$. $\widetilde{\rho}$ is more skewed to the left than $\hat{\rho}$ and $\widetilde{\rho}$ has a larger kurtosis than $\hat{\rho}$. Since the true value of ρ is 0.9, BE is distributed at the nearer place to the true value than MLE. Figure 4.13 displays the empirical distributions of σ_ϵ^2. MLE $\hat{\sigma}_\epsilon^2$ is biased and underestimated, but it has a smaller variance than BE $\widetilde{\sigma}_\epsilon^2$. In addition, we can see that BE $\widetilde{\sigma}_\epsilon^2$ is distributed around the true value.

4.3.4 Summary

In Section 4.3, we have compared MLE with BE, using the regression model with the autocorrelated error term. Chib (1993) applied the Gibbs sampler to the autocorrelation model, where the initial density of the error term is ignored. Under this setup, the posterior distribution of ρ reduces to the normal distribution. Therefore, random draws of ρ given β, σ_ϵ^2 and (y_t, X_t) can be easily generated. However, when the initial density of the error term is taken into account, the posterior distribution of ρ is not normal and it cannot be represented in an explicit functional form. Accordingly, in Section 4.3, the Metropolis-Hastings algorithm have been applied to generate random draws of ρ from its posterior density.

The obtained results are summarized as follows. Given $\beta' = (10, 1, 1)$ and $\sigma^2 = 1$, in Figure 4.7 we have the relationship between ρ and $\hat{\rho}$, and $\widetilde{\rho}$ corresponding to ρ is drawn in Figure 4.8. In the two figures, we can observe: (i) both MLE and BE approach the true parameter value as n is large, and (ii) BE is closer to the 45° degree line than MLE and accordingly BE is superior to MLE.

Moreover, we have compared MLE with BE in Tables 4.5 and 4.6, where $\beta' = (10, 1, 1)$, $\rho = 0.9$ and $\sigma^2 = 1$ are taken as the true values. As for the regression coefficient β, both MLE and BE gives us the unbiased estimators. However, we have obtained the result that BE of β is more efficient than MLE. For estimation of σ^2, BE is less biased than MLE. In addition, BE of the autocorrelation coefficient ρ is also less biased than MLE. Therefore, as for inference on β, BE is superior to MLE, because it is plausible to consider that the estimated variance of $\hat{\beta}$ is biased much more than that of $\widetilde{\beta}$. Remember that variance of $\hat{\beta}$ depends on both ρ and σ^2. Thus, from the simulation studies, we can conclude that BE performs much better than MLE.

References

Amemiya, T., 1985, *Advanced Econometrics*, Cambridge:Harvard University Press.

Andrews, D.W.K., 1993, "Exactly Median-Unbiased Estimation of First Order Autoregressive / Unit Root Models," *Econometrica*, Vol.61, No.1, pp.139 – 165.

Bernardo, J.M. and Smith, A.F.M., 1994, *Bayesian Theory*, John Wiley & Sons.

Boscardin, W.J. and Gelman, A., 1996, "Bayesian Computation for parametric Models of Heteroscedasticity in the Linear Model," in *Advances in Econometrics, Vol.11 (Part A)*, edited by Hill, R.C., pp.87 – 109, Connecticut:JAI Press Inc.

Carlin, B.P. and Louis, T.A., 1996, *Bayes and Empirical Bayes Methods for Data Analysis*, Chapman & Hall.

Chen, M.H., Shao, Q.M. and Ibrahim, J.G., 2000, *Monte Carlo Methods in Bayesian Computation*, Springer-Verlag.

Chib, S., 1993, "Bayes Regression with Autoregressive Errors: A Gibbs Sampling Approach," *Journal of Econometrics*, Vol.58, No.3, pp.275 – 294.

Chib, S. and Greenberg, E., 1994, "Bayes Inference in Regression Models with ARMA(p, q) Errors," *Journal of Econometrics*, Vol.64, No.1&2, pp.183 – 206.

Chib, S. and Greenberg, E., 1995, "Understanding the Metropolis-Hastings Algorithm," *The American Statistician*, Vol.49, No.4, pp.327 – 335.

Gamerman, D., 1997, *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference*, Chapman & Hall.

Geweke, J., 1992, "Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments," in *Bayesian Statistics, Vol.4*, edited by Bernardo, J.M., Berger, J.O., Dawid, A.P. and Smith, A.F.M., pp.169 – 193 (with discussion), Oxford University Press.

Greene, W.H., 1997, *Econometric Analysis* (Third Edition), Prentice-Hall.

Harvey, A.C., 1976, "Estimating Regression Models with Multiplicative Heteroscedasticity," *Econometrica*, Vol.44, No.3, pp.461 – 465.

Hogg, R.V. and Craig, A.T., 1995, *Introduction to Mathematical Statistics* (Fifth Edition), Prentice Hall.

Judge, G., Hill, C., Griffiths, W. and Lee, T., 1980, *The Theory and Practice of Econometrics*, John Wiley & Sons.

Mengersen, K.L., Robert, C.P. and Guihenneuc-Jouyaux, C., 1999, "MCMC Convergence Diagnostics: A Reviewww," in *Bayesian Statistics, Vol.6*, edited by Bernardo, J.M., Berger, J.O., Dawid, A.P. and Smith, A.F.M., pp.514 – 440 (with discussion), Oxford University Press.

O'Hagan, A., 1994, *Kendall's Advanced Theory of Statistics*, Vol.2B (Bayesian Inference), Edward Arnold.

Ohtani, K., 1982, "Small Sample Properties of the Two-step and Three-step Estimators in a Heteroscedastic Linear Regression Model and the Bayesian Alternative," *Economics Letters*, Vol.10, pp.293 – 298.

Robert, C.P. and Casella, G., 1999, *Monte Carlo Statistical Methods*, Springer-Verlag.

Smith, A.F.M. and Roberts, G.O., 1993, "Bayesian Computation via Gibbs Sampler and Related Markov Chain Monte Carlo Methods," *Journal of the Royal Statistical Society*, Ser.B, Vol.55, No.1, pp.3 – 23.

Tanizaki, H., 2000, "Bias Correction of OLSE in the Regression Model with Lagged Dependent Variables," *Computational Statistics and Data Analysis*, Vol.34, No.4, pp.495 – 511.

Tanizaki, H., 2001, "On Least-Squares Bias in the AR(p) Models: Bias Correction Using the Bootstrap Methods," Unpublished Manuscript.

Tanizaki, H. and Zhang, X., 2001, "Posterior Analysis of the Multiplicative Heteroscedasticity Model," *Communications in Statistics, Theory and Methods*, Vol.30, No.2, pp.855 – 874.

Tierney, L., 1994, "Markov Chains for Exploring Posterior Distributions," *The Annals of Statistics*, Vol.22, No.4, pp.1701 – 1762.

Zellner, A., 1971, *An Introduction to Bayesian Inference in Econometrics*, John Wiley & Sons.

Chapter 5

Bias Correction of OLSE in AR Models

This chapter is based on Tanizaki (2000, 2001). In the case where the lagged dependent variables are included in the regression model, it is known that the ordinary least squares estimates (OLSE) are biased in small sample and that bias increases as the number of the irrelevant variables increases. In this chapter, based on the bootstrap methods, an attempt is made to obtain the unbiased estimates in autoregressive and non-Gaussian cases. We introduce the residual-based bootstrap method in this chapter. See Efron and Tibshirani (1993) for the bootstrap methods. Some simulation studies are performed to examine whether the estimation procedure discussed in this chapter works well or not. We obtain the results that it is possible to recover the true parameter values based on OLSE and that the discussed procedure gives us the less biased estimators than OLSE.

5.1 Introduction

In the case where the lagged dependent variables are included in the regression model, it is known that the OLSEs of autoregressive (AR) models are biased in small sample.

Hurwicz (1950), Marriott and Pope (1954), Kendall (1954) and White (1961) discussed the mean-bias of the OLSE. Quenouille (1956) introduced the jackknife estimator of the AR parameter which is median-unbiased to order $1/n$ as the sample size n goes to infinity, where the trend term is not taken into account. Orcutt and Winokur (1969) constructed approximately mean-unbiased estimates of the AR parameter in stationary models. Sawa (1978), Tanaka (1983) and Tsui and Ali (1994) also examined the AR(1) models, where the exact moments of OLSE are discussed. Shaman and Stine (1988) established the mean-bias of the OLSE to order $1/n$ in stationary AR(p) (also see Maekawa (1987) for the AR(p) models). Grubb and Symons (1987) gave an expression to order $1/n$ for bias to the estimated coefficient on a lagged dependent variable when all other regressors are exogenous (also see Tse (1982) and Maekawa

(1983) for the AR models including the exogenous variables). Peters (1989) studied the finite sample sensitivity of OLSE of the AR(1) term with nonnormal errors. In Abadir (1993), an analytical formula was derived to approximate the finite sample bias of OLSE of the AR(1) term when the underlying process has a unit root.

Moreover, in the case where the true model is the first order AR model, Andrews (1993) examined the cases where the estimated models are the AR(1), the AR(1) with a constant term and the AR(1) with constant ane trend terms, where the exact median-unbiased estimator of the first order autoregressive model is derived by utilizing the Imhof (1961) algorithm. Andrews and Chen (1994) obtained the approximately median-unbiased estimator of autoregressive models, where Andrews (1993) is applied by transforming AR(p) models into AR(1) and taking the iterative procedure.

Thus, the AR models have been studied with respect to various aspects, i.e., (i) a stationary model or a unit root model, (ii) the first order autoregressive model or the higher order autoregressive models, (iii) an autoregressive model with or without exogenous variables, and (iv) a normal error or a nonnormal error. Tanizaki (2000) proposed the median- and mean-unbiased estimators using simulation techniques, where the underlying assumption is that the error term is normal. Furthermore, in more general formulation which can be applied to all the cases (i) – (iv), using the bootstrap methods Tanizaki (2001) derived the unbiased estimates of the regression coefficients in small sample. In this chapter, Tanizaki (2000, 2001) is introduced, some simulation studies are performed and an empirical study is shown as an example.

5.2 OLSE Bias

We take the autoregressive model which may include the exogenous variables, say X_t. That is, consider the following simple regression model:

$$y_t = X_t\beta + \sum_{j=1}^{p} \alpha_j y_{t-j} + u_t = z_t\theta + u_t, \tag{5.1}$$

for $t = p+1, p+2, \cdots, n$, where X_t and β are a $1 \times k$ vector and a $k \times 1$ vector, respectively. θ and z_t are given by $\theta = (\beta', \alpha')'$ and $z_t = (X_t, y_{t-1}, y_{t-2}, \cdots, y_{t-p})$, where $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_p)'$. u_t is assumed to be distributed with mean zero and variance σ^2. We will discuss later for the distribution function of the error term u_t. The initial values $y_p, y_{p-1}, \cdots, y_1$ are assumed to be constant for simplicity of discussion. Thus, it is assumed that y_t follows the pth order autoregressive model.

First of all, in this section we examine by Monte Carlo simulations how large the OLSE bias is. Let Model A be the case of $k = 0$, Model B be the case of $k = 1$ and $X_t = 1$ and Model C be the case of $k = 2$ and $X_t = (1,\ x_{1t})$, i.e,

- Model A: $y_t = \sum_{j=1}^{p} \alpha_j y_{t-j} + u_t,$

- Model B: $y_t = \beta_1 + \sum_{j=1}^{p} \alpha_j y_{t-j} + u_t$,
- Model C: $y_t = \beta_1 + \beta_2 x_{1t} + \sum_{j=1}^{p} \alpha_j y_{t-j} + u_t$,

for $t = p+1, p+2, \cdots, n$, given the initial condition $y_1 = y_2 = \cdots = y_p = 0$. In Model C, we take x_{1t} as the trend term, i.e., $x_{1t} = t$.

We focus only on the case of $p = 1$, i.e., the AR(1) model. Suppose that the true model is represented by Model A with $p = 1$. When $x_{1t} = t$ (time trend) is taken in Model C, it is known that OLSE of α_1 from Model C gives us the largest bias while OLSE of α_1 from Model A yields the smallest one (see, for example, Andrews (1993)). Figures 5.1 – 5.3 show the relationship between the true autoregressive coefficient (i.e., α_1) and the arithmetic mean of OLSEs from 10^4 simulation runs (i.e., $\hat{\alpha}_1$). In order to draw the relationship between the true parameter value of α_1 and its OLSE, we take the following simulation procedure.

(i) Given α_1, $u_t \sim N(0,1)$ and $y_1 = 0$, generate y_2, y_3, $\cdots$, y_n by Model A, where $n = 10, 15, 20$.

(ii) Compute OLSE of α_1 by estimating Model A, OLSE of (β_1, α_1) by estimating Model B, and OLSE of $(\beta_1, \beta_2, \alpha_1)$ by estimating Model C. That is, we compute the above three kinds of OLSEs. Note in Model C that $x_{1t} = t$ (time trend) is taken in this simulation study.

(iii) Repeat (i) and (ii) 10^4 times.

(iv) Obtain the arithmetic mean from the 10^4 OLSEs of α_1 (we focus on the OLSE of α_1).

(v) Given the exactly same random draws for u_t (i.e., $10^4 \times (n-p)$ random draws for $n = 20$ and $p = 1$), repeat (i) – (iv) for $\alpha_1 = -1.20, -1.19. - 1.18, \cdots, 1.20$. We do not assume stationarity of the data (y_t, X_t).

Thus, we have the arithmetic mean from the 10^4 OLSEs corresponding to the true parameter value for each model. In Figures 5.1 – 5.3, the true model is given by Model A and it is estimated by Models A – C. The horizontal line implies the true parameter value of the AR(1) coefficient and the vertical line indicates the OLSE corresponding to the true parameter value. Unless the OLSE is biased, the 45° degree line represents the relationship between the true parameter value and the OLSE.

In Figures 5.1 – 5.3, each line indicates the arithmetic mean of the 10^4 OLSEs. There is the largest bias around $\alpha_1 = 1$ for all the Models A – C. From Figures 5.1 – 5.3, bias drastically increases as number of the exogenous variables increases. That is, in the case where α_1 is positive, OLSE of Model C has the largest downward-bias and OLSE of Model A represents the smallest downward-bias, which implies that inclusion of more extra variables results in larger bias of OLSE. Moreover, we can

Figure 5.1: The Arithmetic Average from the 10^4 OLSEs of AR(1) Coeff.
— $N(0, 1)$ Error and $n = 10$ —

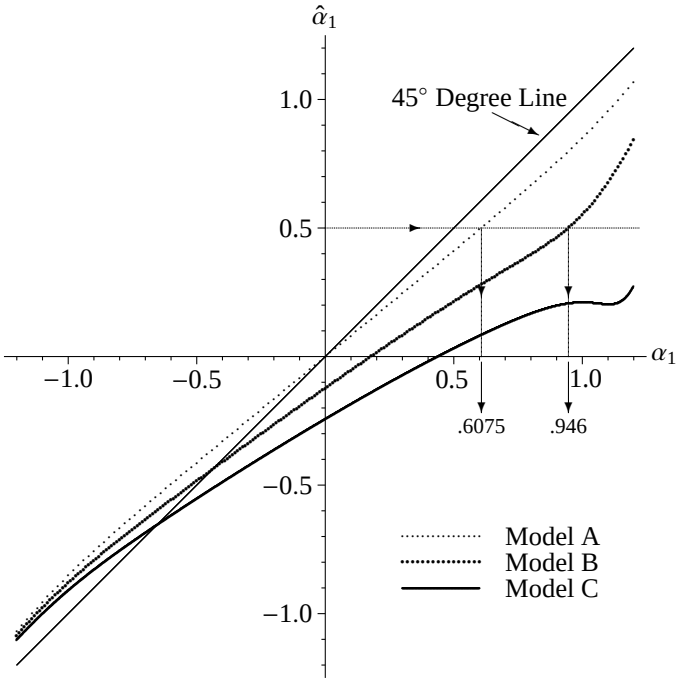


Figure 5.2: The Arithmetic Average from the 10^4 OLSEs of AR(1) Coeff.
— $N(0, 1)$ Error and $n = 15$ —

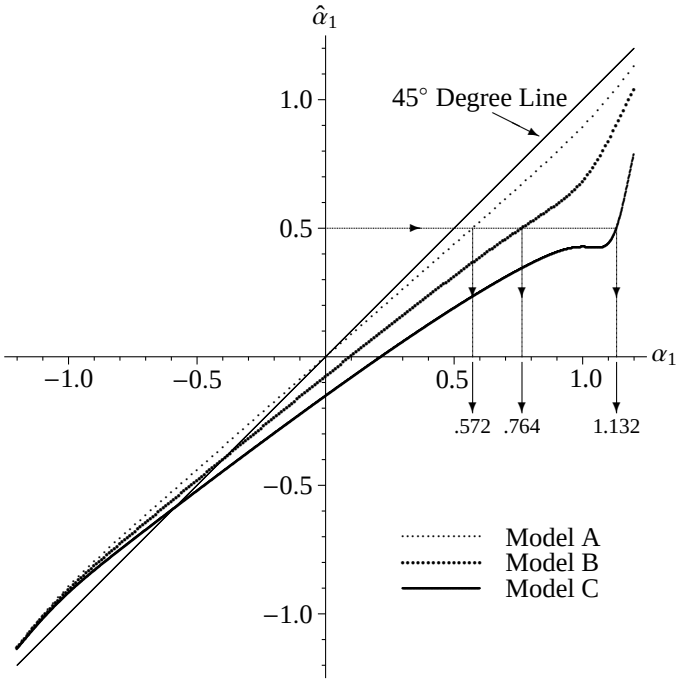
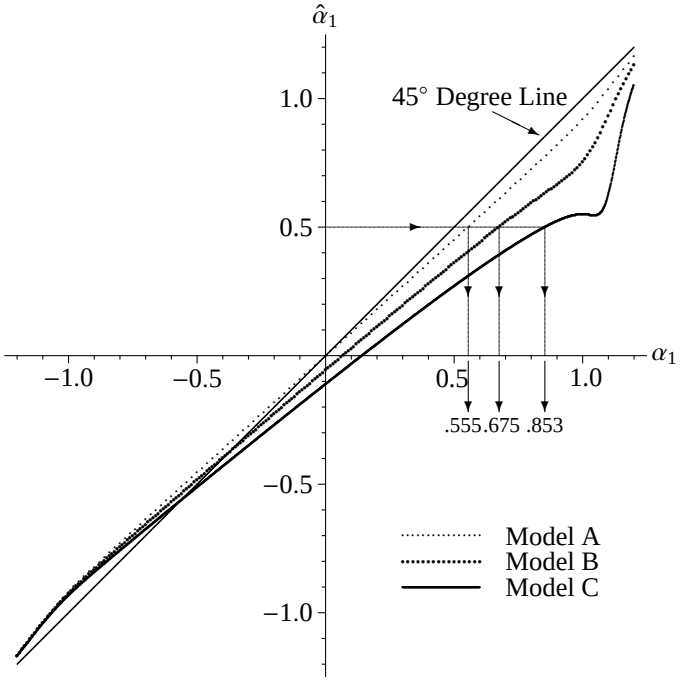


Figure 5.3: The Arithmetic Average from the 10^4 OLSEs of AR(1) Coeff.
— $N(0, 1)$ Error and $n = 20$ —



see from the three figures that the OLS bias decreases as the sample size n increases, because the three lines are close to the 45° line as n is large.

Thus, from Figures 5.1 – 5.3 we can see how large the OLSE bias is. That is, discrepancy between the 45° degree line and the other lines increases as number of the extra variables increases. Now we consider improving the OLSE bias. In Figures 5.1 – 5.3, we see the arithmetic mean from the 10^4 OLSEs given the true coefficient, respectively. It is also possible to read the figures reversely. For example, in Figure 5.3, when OLSE is obtained as $\hat{\alpha}_1 = 0.5$ from actually observed data, the true parameter value α_1 can be estimated as 0.555 for Model A, 0.675 for Model B and 0.853 for Model C. For the estimator discussed in this chapter, we consider shifting the distribution of OLSE toward the distribution around the true value in the sense of mean.

In practice, no one knows the true model, including the underlying distribution of the error term. What we can do is to estimate the model based on the underlying assumptions. Figures 5.1 – 5.3 indicate that inclusion of more extra variables possibly yields serious biased OLSE and furthermore that the true parameter values can be recovered from the estimated model even if we do not know the true model. In Section 5.3, based on this idea, we obtain the unbiased estimator, which can be applied to any case of the higher order autoregressive models, the nonnormal error term and inclusion of the exogenous variables other than the constant and trend terms. Here, we take the

constant term and the time trend as $X_t = (1, x_{1t})$, although any exogenous variables can be included in the model.

5.3 Bias Correction Method

Since it is well known that OLSE of the autoregressive coefficient vector in the AR(p) model is biased in small sample (see, for example, Andrews (1993), Andrews and Chen (1994), Diebold and Rudebusch (1991), Hurwicz (1950), Kendall (1954), Marriott and Pope (1954), Quenouille (1956) and so on), OLSE of θ, $\hat{\theta} = (\hat{\beta}', \hat{\alpha}')'$, is clearly biased.

To obtain the unbiased estimator of θ, the underlying idea in this chapter is described as follows. Let θ be an unknown parameter and $\hat{\theta}$ be the OLS estimate of θ. Suppose that the distribution function of $\hat{\theta}$ is given by $f_{\hat{\theta}}(\cdot)$, which is not obtained analytically in the case where the lagged dependent variables are included in the explanatory variables. Since $\hat{\theta}$ is biased, we have $\theta \neq \mathrm{E}(\hat{\theta})$, where the expectation $\mathrm{E}(\hat{\theta})$ is defined as follows:

$$\mathrm{E}(\hat{\theta}) \equiv \int_{-\infty}^{+\infty} x f_{\hat{\theta}}(x)\, \mathrm{d}x. \tag{5.2}$$

To obtain the relationship between $\hat{\theta}$ and θ, let $\{\hat{\theta}_1^*, \hat{\theta}_2^*, \cdots, \hat{\theta}_N^*\}$ be a sequence of the biased estimates of θ, which are taken as the random draws generated from $f_{\hat{\theta}}(\cdot)$. Note that $\hat{\theta}$ implies the OLSE obtained from the actual data while $\hat{\theta}_i^*$ denotes the ith OLSE based on the simulated data given the true parameter value θ. Therefore, $\hat{\theta}_i^*$ depends on θ, i.e., $\hat{\theta}_i^* = \hat{\theta}_i^*(\theta)$ for all $i = 1, 2, \cdots, N$. Suppose that given θ we can generate the N random draws from $f_{\hat{\theta}}(\cdot)$, which are denoted by $\hat{\theta}_1^*, \hat{\theta}_2^*, \cdots, \hat{\theta}_N^*$. Using the N random draws, the integration in equation (5.2) can be written as follows:

$$\mathrm{E}(\hat{\theta}) = \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} \hat{\theta}_i^*(\theta). \tag{5.3}$$

Let us define the unbiased estimator of θ as $\bar{\theta}$. Equation (5.3) implies that $\bar{\theta}$ is given by the θ which satisfies the following equation:

$$\hat{\theta} = \frac{1}{N} \sum_{i=1}^{N} \hat{\theta}_i^*(\theta) \equiv g(\theta), \tag{5.4}$$

where $\hat{\theta}$ in the left-hand side represents the OLSE of θ based on the original data y_t and X_t. $g(\cdot)$ denotes a $(k + p) \times 1$ vector function. The solution of θ obtained from equation (5.4) are denoted by $\bar{\theta}$, which corresponds to the unbiased estimator of θ. Conventionally, it is impossible to obtain an explicit functional form of $g(\cdot)$ in equation (5.4). Therefore, equation (5.4) is practically solved by an iterative procedure or a simple grid search. In this chapter, the computational implementation is shown as the following iterative procedure.

1. Given the actual time series data (i.e., y_t and X_t), estimate θ and σ^2 in (5.1) by OLS, which are denoted by $\hat{\theta}$ and $\hat{\sigma}^2$.
2. Let u_t^* be the random draw with mean zero and variance $\hat{\sigma}^2$. Suppose for now that the random draws $u_{p+1}^*, u_{p+2}^*, \cdots, u_n^*$ are available. We will discuss later for the random number generation method of u_t^*.
3. Let $\theta^{(j)}$ be the jth iteration of θ and y_t^* be the random draw of y_t. Given the initial values $\{y_p^*, y_{p-1}^*, \cdots, y_1^*\}$, the exogenous variable X_t for $t = p+1, p+2, \cdots, n$ and $\theta^{(j)}$, we obtain y_t^* substituting u_t^* into u_t in equation (5.1), where the initial values $\{y_p^*, y_{p-1}^*, \cdots, y_1^*\}$ may be taken as the actual data $\{y_p, y_{p-1}, \cdots, y_1\}$. That is, y_t^* is generated from $z_t^*\theta^{(j)} + u_t^*$ given $\theta^{(j)}$ and the random draw u_t^*, where $z_t^* = (X_t, y_{t-1}^*, y_{t-2}^*, \cdots, y_{t-p}^*)$. For $j = 1$ (the first iteration), we may set $\theta^{(1)} = \hat{\theta}$.
4. Given y_t^* and X_t for $t = 1, 2, \cdots, n$, compute the OLSE of $\theta^{(j)}$, which is denoted by $\hat{\theta}^*$.
5. Repeat Steps 2 – 4 N times, where $N = 10^4$ is taken in this chapter. Then, the N OLSEs based on $\theta^{(j)}$ are obtained, which correspond to $\hat{\theta}_i^*(\theta^{(j)})$, $i = 1, 2, \cdots, N$, in equation (5.4). Based on the N OLSEs obtained from $\theta^{(j)}$, compute the arithmetic mean for each element of θ, i.e., compute $g(\theta^{(j)}) \equiv (1/N)\sum_{i=1}^N \hat{\theta}_i^*(\theta^{(j)})$.
6. As in equation (5.4), $\hat{\theta}$ should be equal to the arithmetic average $g(\theta^{(j)}) \equiv (1/N)\sum_{i=1}^N \hat{\theta}_i^*(\theta^{(j)})$. For each element of θ, therefore, $\theta^{(j+1)}$ should be smaller than $\theta^{(j)}$ if $\hat{\theta}$ is less than the arithmetic average, and it should be larger than $\theta^{(j)}$ otherwise. Here, we consider that each element of $g(\theta)$ is a monotone increasing function of the corresponding element of θ (see Andrews (1993)). Thus, $\theta^{(j)}$ is updated to $\theta^{(j+1)}$. An example of the optimization procedure is described in the next section.
7. Repeat Steps 2 – 6 until $\theta^{(j+1)}$ is stable, where the limit of $\theta^{(j+1)}$ with respect to j is taken as $\bar{\theta}$. Note that the same random draws of u_t, generated in Step 2, should be utilized for all j, i.e., $N \times (n-p)$ random draws have to be stored.

Now, in Step 2 we need to consider generating the random draw of u_t, i.e., u_t^*, which is assumed to be distributed with mean zero and variance $\hat{\sigma}^2$. In the regression model (5.1), the underlying distribution of u_t is conventionally unknown. Therefore, we take the following four types of random draws for u_t, i.e., the normal error (N), the chi-squared error (X), the uniform error (U) and the residual-based error (R), which are represented as follows.

(N) $u_t^* = \hat{\sigma}\epsilon_t$, where $\epsilon_t \sim N(0,1)$ and $\hat{\sigma}$ denotes the standard error of regression by OLS, i.e., $\hat{\sigma}$ is obtained by $\hat{\sigma}^2 = \sum_{t=p+1}^n (y_t - z_t\hat{\theta})^2/(n-p-k)$.

(X) $u_t^* = \hat{\sigma}\epsilon_t$, where $\epsilon_t = \dfrac{v_t - 1}{\sqrt{2}}$ and $v_t \sim \chi^2(1)$.

(U) $u_t^* = \hat{\sigma}\epsilon_t$, where $\epsilon_t \sim U(-\sqrt{3}, \sqrt{3})$.

(R) u_t^* is resampled from $\{ce_1, ce_2, \cdots, ce_n\}$ with equal probability $1/n$, where e_t denotes the OLS residual at time t, i.e., $e_t = y_t - z_t\hat{\theta}$, and $c = \sqrt{n/(n-p-k)}$ is taken (see Wu (1986) for c). This sampling method is called the **bootstrap method**.

Note that ϵ_t in (N), (X) and (U) is normalized to the random variable with mean zero and variance one. Thus, repeating the above sampling procedures for $t = p+1, p+2, \cdots, n$, we can generate the random draws of u_t, i.e., $\{u_{p+1}^*, u_{p+2}^*, \cdots, u_n^*\}$, in Step 2. In practice we often have the case where the underlying distribution of the true data series is different from that of the simulated one, because the distribution of u_t is not known. (R) does not assume any distribution for u_t. Therefore, it might be expected that (R) is more robust compared with (N), (X) and (U).

In Section 5.5, (N), (X), (U) and (R) are examined for each of the three models, i.e., Model A – C.

5.3.1 Optimization Procedure

In Step 6, we need to obtain $\theta^{(j)}$ which satisfies $\hat{\theta} = g(\theta^{(j)})$. In order to solve $\hat{\theta} = g(\theta^{(j)})$ with respect to $\theta^{(j)}$, in this section we take an example of an iterative procedure, where the numerical optimization procedure is utilized.

Using equation (5.4), we update the parameter θ as follows:

$$\theta^{(j+1)} = \theta^{(j)} + \gamma^{(j)}\left(\hat{\theta} - g(\theta^{(j)})\right), \tag{5.5}$$

where j denotes the jth iteration. In the first iteration, OLSE of θ is taken for $\theta^{(1)}$, i.e., $\theta^{(1)} = \hat{\theta}$. $\gamma^{(j)}$ is a scalar, which may depend on the number of iteration, i.e., j.

Equation (5.5) is justified as follows. It might be appropriate for an interpretation of $\gamma^{(j)}$ to consider that the Newton-Raphson optimization procedure is taken. Approximating $\hat{\theta} - g(\theta)$ about $\theta = \theta^{(j)}$, we have:

$$\begin{aligned} 0 &= \hat{\theta} - g(\theta) \\ &\approx \hat{\theta} - g(\theta^{(j)}) - \frac{\partial g(\theta^{(j)})}{\partial \theta'}(\theta - \theta^{(j)}). \end{aligned}$$

Then, we can rewrite as:

$$\theta = \theta^{(j)} + \left(\frac{\partial g(\theta^{(j)})}{\partial \theta'}\right)^{-1}\left(\hat{\theta} - g(\theta^{(j)})\right).$$

Replacing θ by $\theta^{(j+1)}$, the following equation is derived:

$$\theta^{(j+1)} = \theta^{(j)} + \left(\frac{\partial g(\theta^{(j)})}{\partial \theta'}\right)^{-1}\left(\hat{\theta} - g(\theta^{(j)})\right),$$

which is equivalent to equation (5.5) with the following condition:

$$\left(\frac{\partial g(\theta^{(j)})}{\partial \theta'}\right)^{-1} = \gamma^{(j)} I_{k+p},$$

where I_{k+p} denotes a $(k+p)\times(k+p)$ identity matrix. Since $g(\theta)$ cannot be explicitly specified, we take the first derivative of $g(\theta)$ as the diagonal matrix $\gamma^{(j)} I_{k+p}$. Moreover, taking into account the convergence speed, $\gamma^{(j)} = c^{j-1}$ is used in this chapter, where $c = 0.9$.

Thus, using equation (5.5), the unbiased estimator can be obtained. When $\theta^{(j+1)}$ is stable, we take it as the unbiased estimate of θ, i.e., $\overline{\theta}$. As for convergence criterion, when each element of $\theta^{(j+1)} - \theta^{(j)}$ is less than 0.001 in absolute value, we consider that $\theta^{(j+1)}$ is stable.

5.3.2 Standard Error, Confidence Interval and Etc

In the above procedure, $\overline{\theta}$ is obtained. For statistical inference, we need the distribution of $\overline{\theta}$. However, it is impossible to know the distribution explicitly. Therefore, we consider obtaining the distribution numerically.

When we judge that $\theta^{(j+1)}$ is stable in Step 7, N OLSEs of θ are available in Step 5. For each of the N OLSEs of θ, we may compute the unbiased estimate suggested in this chapter. That is, we may perform Steps 1 – 7 N times. Thus, we can obtain the N unbiased estimates based on the N OLSEs. From the N unbiased estimates, we can compute the percentiles by sorting them, the standard errors and etc. In order to obtain one unbiased estimate, we have to compute N OLSEs. Therefore, we need $N \times N$ OLSEs for the N unbiased estimates. This implies that constructing the confidence interval, the standard error of the coefficients and etc takes an extremely lot of time computationally. Once we have the N unbiased estimates, we can compute the standard error, skewness, kurtosis and so on using the N unbiased estimates shown above.

5.3.3 Standard Error of Regression

Once $\overline{\theta}$ is computed by Steps 1 – 7, $\overline{\sigma}^2$ is derived by using the following formula:

$$\overline{\sigma}^2 = \frac{1}{n-p-k}\sum_{t=p+1}^{n}(y_t - X_t\overline{\beta} - \sum_{j=1}^{p}\overline{\alpha}_j y_{t-p})^2. \tag{5.6}$$

5.4 Monte Carlo Experiments

In the previous section, we have derived the unbiased estimator of θ when the lagged dependent variables are included in the explanatory variables. Using the first, second and third order autoregressive models, we examine whether the suggested procedure works well.

5.4.1 AR(1) Models

Let Model A be the case of $k = 0$, Model B be the case of $k = 1$ and $X_t = 1$ and Model C be the case of $k = 2$ and $X_t = (1,\ x_{1t})$, i.e,

$$\text{Model A:} \quad y_t = \sum_{j=1}^{p} \alpha_j y_{t-j} + u_t,$$

$$\text{Model B:} \quad y_t = \beta_1 + \sum_{j=1}^{p} \alpha_j y_{t-j} + u_t,$$

$$\text{Model C:} \quad y_t = \beta_1 + \beta_2 x_{1t} + \sum_{j=1}^{p} \alpha_j y_{t-j} + u_t,$$

for $t = p+1, p+2, \ldots, n$, given the initial condition $y_1 = y_2 = \cdots = y_p = 0$. In Model C, we take x_{1t} as the trend term, i.e., $x_{1t} = t$.

The true distribution of the error term u_t is assumed to be normal in Table 5.1, chi-squared in Table 5.2 and uniform in Table 5.3. For all the tables, mean and variance of the error term are normalized to be zero and one, respectively. Since the true distribution of the error term is not known in practice, we examine (N), (X), (U) and (R) for all the estimated models.

In Tables 5.1 – 5.3, the case of $p = 1$ is examined although the suggested procedure would be applied to any p. The sample size is $n = 20, 40, 60$. For the parameter values, $\alpha_1 = 0.6, 0.9, 1.0, \beta_1 = 0.0, 1.0, 2.0$ and $\beta_2 = 0.0$ are taken. We perform 10^4 simulation runs. The arithmetic averages from the 10^4 estimates of α_1 are shown in Tables 5.1 – 5.3. The values in the parentheses denote the root mean square errors from the 10^4 estimates. In Tables 5.1 – 5.3, (O) represents OLS, while (N), (X), (U) and (R) are discussed in Section 5.3. (1) – (5) in each table denote as follows:

(1) The true model is $\alpha_1 = 0.6, 0.9, 1.0$ and $(\beta_1, \beta_2) = (0, 0)$, i.e., Model A, and the estimated model is Model A.

(2) The true model is $\alpha_1 = 0.6, 0.9, 1.0$ and $(\beta_1, \beta_2) = (0, 0)$, i.e., Model A, and the estimated model is Model B.

(3) The true model is $\alpha_1 = 0.6, 0.9, 1.0$ and $(\beta_1, \beta_2) = (1, 0)$, i.e., Model B, and the estimated model is Model B.

(4) The true model is $\alpha_1 = 0.6, 0.9, 1.0$ and $(\beta_1, \beta_2) = (2, 0)$, i.e., Model B, and the estimated model is Model B.

(5) The true model is $\alpha_1 = 0.6, 0.9, 1.0$ and $(\beta_1, \beta_2) = (0, 0)$, i.e., Model A, and the estimated model is Model C.

First, when the true model is represented by Model A with $p = 1$, we consider estimating Model A by Models A – C. Note that Models A, B and C correspond to $k = 0$, $k = 1$ and $k = 2$, respectively, where k denotes the number of exogenous variables. It is known that the OLSE of α_1 estimated by Model C gives us the largest

Table 5.1: Estimates of α_1 (Case: $\beta_2 = 0$) — $N(0,1)$ Error

n	Estimated Model		(1) A	(2) B	(3) B	(4) B	(5) C
		$\alpha_1 \setminus \beta_1$	0.0	0.0	1.0	2.0	0.0
20	(O)	0.6	0.541 (.206)	0.441 (.269)	0.479 (.224)	0.530 (.158)	0.341 (.345)
	(N)		0.596 (.217)	0.601 (.267)	0.599 (.221)	0.598 (.154)	0.601 (.321)
	(X)		0.568 (.222)	0.580 (.262)	0.583 (.222)	0.588 (.153)	0.574 (.312)
	(U)		0.596 (.218)	0.596 (.269)	0.595 (.222)	0.597 (.155)	0.593 (.323)
	(R)		0.596 (.217)	0.599 (.267)	0.598 (.221)	0.598 (.154)	0.598 (.321)
	(O)	0.9	0.819 (.180)	0.665 (.307)	0.839 (.116)	0.883 (.050)	0.523 (.438)
	(N)		0.893 (.168)	0.842 (.231)	0.901 (.104)	0.900 (.047)	0.805 (.303)
	(X)		0.876 (.178)	0.823 (.239)	0.894 (.104)	0.897 (.046)	0.778 (.311)
	(U)		0.894 (.169)	0.841 (.234)	0.903 (.105)	0.901 (.047)	0.802 (.307)
	(R)		0.893 (.169)	0.841 (.232)	0.902 (.105)	0.900 (.047)	0.804 (.304)
	(O)	1.0	0.919 (.167)	0.754 (.310)	0.983 (.048)	0.996 (.021)	0.550 (.503)
	(N)		0.991 (.146)	0.902 (.217)	1.006 (.051)	1.000 (.021)	0.827 (.329)
	(X)		0.980 (.155)	0.888 (.228)	1.000 (.048)	0.999 (.021)	0.800 (.344)
	(U)		0.992 (.147)	0.902 (.219)	1.008 (.052)	1.000 (.021)	0.824 (.333)
	(R)		0.991 (.147)	0.902 (.218)	1.007 (.052)	1.000 (.021)	0.825 (.330)
40	(O)	0.6	0.569 (.138)	0.523 (.163)	0.533 (.148)	0.553 (.119)	0.476 (.195)
	(N)		0.600 (.142)	0.600 (.159)	0.599 (.143)	0.599 (.116)	0.600 (.177)
	(X)		0.586 (.142)	0.593 (.157)	0.593 (.142)	0.595 (.116)	0.593 (.174)
	(U)		0.597 (.142)	0.597 (.160)	0.597 (.144)	0.598 (.116)	0.598 (.178)
	(R)		0.598 (.142)	0.598 (.159)	0.598 (.143)	0.598 (.116)	0.599 (.177)
	(O)	0.9	0.856 (.105)	0.785 (.165)	0.863 (.071)	0.888 (.033)	0.713 (.230)
	(N)		0.899 (.098)	0.889 (.130)	0.901 (.063)	0.900 (.031)	0.878 (.165)
	(X)		0.890 (.102)	0.881 (.132)	0.899 (.063)	0.900 (.031)	0.869 (.167)
	(U)		0.898 (.099)	0.888 (.131)	0.901 (.063)	0.900 (.031)	0.877 (.166)
	(R)		0.898 (.099)	0.888 (.130)	0.900 (.063)	0.900 (.031)	0.877 (.165)
	(O)	1.0	0.957 (.088)	0.870 (.166)	0.996 (.015)	0.999 (.007)	0.759 (.274)
	(N)		0.996 (.074)	0.952 (.112)	1.001 (.016)	1.000 (.007)	0.916 (.170)
	(X)		0.991 (.077)	0.947 (.116)	1.000 (.015)	1.000 (.007)	0.907 (.176)
	(U)		0.995 (.074)	0.951 (.112)	1.001 (.016)	1.000 (.007)	0.915 (.171)
	(R)		0.995 (.074)	0.951 (.112)	1.001 (.016)	1.000 (.007)	0.915 (.171)
60	(O)	0.6	0.579 (.110)	0.550 (.123)	0.555 (.116)	0.565 (.099)	0.519 (.143)
	(N)		0.600 (.112)	0.601 (.120)	0.601 (.113)	0.601 (.097)	0.600 (.130)
	(X)		0.591 (.112)	0.595 (.119)	0.596 (.112)	0.597 (.096)	0.595 (.129)
	(U)		0.599 (.112)	0.600 (.120)	0.600 (.113)	0.599 (.097)	0.598 (.130)
	(R)		0.598 (.112)	0.599 (.120)	0.599 (.113)	0.599 (.097)	0.599 (.129)
	(O)	0.9	0.870 (.078)	0.826 (.113)	0.870 (.057)	0.890 (.028)	0.780 (.154)
	(N)		0.900 (.073)	0.899 (.093)	0.900 (.051)	0.900 (.026)	0.896 (.116)
	(X)		0.894 (.075)	0.893 (.094)	0.899 (.051)	0.899 (.026)	0.890 (.117)
	(U)		0.900 (.074)	0.899 (.094)	0.900 (.051)	0.900 (.026)	0.895 (.117)
	(R)		0.898 (.074)	0.898 (.094)	0.900 (.051)	0.900 (.026)	0.895 (.117)
	(O)	1.0	0.971 (.060)	0.913 (.112)	0.998 (.008)	1.000 (.004)	0.836 (.187)
	(N)		0.997 (.049)	0.969 (.074)	1.000 (.008)	1.000 (.004)	0.946 (.113)
	(X)		0.995 (.051)	0.966 (.076)	1.000 (.008)	1.000 (.004)	0.941 (.117)
	(U)		0.997 (.049)	0.969 (.074)	1.000 (.008)	1.000 (.004)	0.945 (.114)
	(R)		0.996 (.049)	0.969 (.074)	1.000 (.008)	1.000 (.004)	0.945 (.114)

Table 5.2: Estimates of α_1 (Case: $\beta_2 = 0$) — $(\chi^2(1) - 1)/\sqrt{2}$ Error

n	Estimated Model		(1) A	(2) B	(3) B	(4) B	(5) C
		$\alpha_1 \setminus \beta_1$	0.0	0.0	1.0	2.0	0.0
20	(O)	0.6	0.572 (.194)	0.456 (.239)	0.494 (.204)	0.538 (.157)	0.354 (.321)
	(N)		0.630 (.211)	0.616 (.233)	0.608 (.200)	0.600 (.156)	0.613 (.287)
	(X)		0.603 (.214)	0.593 (.227)	0.592 (.200)	0.591 (.155)	0.584 (.278)
	(U)		0.630 (.212)	0.610 (.234)	0.605 (.202)	0.598 (.157)	0.605 (.288)
	(R)		0.609 (.214)	0.601 (.228)	0.599 (.200)	0.595 (.156)	0.597 (.281)
	(O)	0.9	0.839 (.164)	0.691 (.275)	0.849 (.120)	0.885 (.053)	0.544 (.416)
	(N)		0.913 (.158)	0.866 (.199)	0.901 (.117)	0.899 (.052)	0.830 (.280)
	(X)		0.897 (.165)	0.847 (.206)	0.894 (.115)	0.897 (.051)	0.800 (.289)
	(U)		0.914 (.158)	0.864 (.201)	0.902 (.118)	0.900 (.052)	0.827 (.284)
	(R)		0.903 (.162)	0.856 (.201)	0.898 (.116)	0.899 (.051)	0.820 (.283)
	(O)	1.0	0.933 (.153)	0.777 (.287)	0.988 (.045)	0.997 (.021)	0.568 (.483)
	(N)		1.004 (.136)	0.920 (.195)	1.005 (.050)	1.000 (.021)	0.852 (.307)
	(X)		0.994 (.143)	0.908 (.206)	1.000 (.047)	0.999 (.021)	0.824 (.322)
	(U)		1.005 (.136)	0.920 (.197)	1.006 (.050)	1.000 (.021)	0.850 (.311)
	(R)		0.997 (.140)	0.916 (.199)	1.002 (.048)	1.000 (.021)	0.845 (.311)
40	(O)	0.6	0.583 (.127)	0.533 (.142)	0.542 (.133)	0.558 (.118)	0.484 (.176)
	(N)		0.615 (.133)	0.611 (.139)	0.607 (.129)	0.602 (.116)	0.609 (.157)
	(X)		0.602 (.132)	0.603 (.137)	0.601 (.128)	0.598 (.116)	0.602 (.154)
	(U)		0.612 (.133)	0.608 (.139)	0.605 (.130)	0.601 (.116)	0.607 (.158)
	(R)		0.603 (.133)	0.604 (.137)	0.602 (.129)	0.599 (.116)	0.603 (.155)
	(O)	0.9	0.864 (.097)	0.795 (.149)	0.864 (.080)	0.888 (.037)	0.724 (.214)
	(N)		0.908 (.093)	0.899 (.115)	0.898 (.074)	0.899 (.036)	0.890 (.150)
	(X)		0.899 (.095)	0.891 (.117)	0.897 (.074)	0.899 (.036)	0.880 (.151)
	(U)		0.907 (.093)	0.898 (.116)	0.899 (.074)	0.899 (.036)	0.890 (.150)
	(R)		0.900 (.095)	0.893 (.116)	0.897 (.074)	0.899 (.036)	0.884 (.150)
	(O)	1.0	0.961 (.083)	0.877 (.158)	0.996 (.015)	0.999 (.007)	0.769 (.261)
	(N)		1.000 (.070)	0.957 (.104)	1.001 (.016)	1.000 (.007)	0.928 (.157)
	(X)		0.995 (.073)	0.952 (.108)	1.000 (.015)	1.000 (.007)	0.919 (.162)
	(U)		0.999 (.070)	0.957 (.104)	1.001 (.016)	1.000 (.007)	0.927 (.158)
	(R)		0.996 (.072)	0.954 (.106)	1.000 (.015)	1.000 (.007)	0.923 (.159)
60	(O)	0.6	0.588 (.101)	0.556 (.110)	0.560 (.104)	0.569 (.094)	0.525 (.129)
	(N)		0.610 (.104)	0.607 (.108)	0.607 (.102)	0.604 (.093)	0.606 (.116)
	(X)		0.601 (.104)	0.602 (.107)	0.601 (.101)	0.600 (.093)	0.601 (.115)
	(U)		0.609 (.104)	0.606 (.108)	0.605 (.102)	0.603 (.093)	0.604 (.116)
	(R)		0.602 (.104)	0.602 (.107)	0.602 (.102)	0.601 (.093)	0.602 (.115)
	(O)	0.9	0.876 (.072)	0.832 (.104)	0.872 (.061)	0.890 (.031)	0.786 (.144)
	(N)		0.906 (.069)	0.905 (.086)	0.900 (.056)	0.900 (.030)	0.903 (.107)
	(X)		0.900 (.071)	0.899 (.087)	0.899 (.056)	0.899 (.030)	0.896 (.107)
	(U)		0.906 (.070)	0.905 (.087)	0.900 (.056)	0.900 (.030)	0.902 (.108)
	(R)		0.901 (.071)	0.900 (.086)	0.900 (.056)	0.900 (.030)	0.898 (.107)
	(O)	1.0	0.973 (.056)	0.917 (.107)	0.998 (.008)	1.000 (.004)	0.841 (.181)
	(N)		1.000 (.046)	0.972 (.068)	1.000 (.008)	1.000 (.004)	0.951 (.107)
	(X)		0.997 (.048)	0.970 (.071)	1.000 (.008)	1.000 (.004)	0.946 (.110)
	(U)		0.999 (.046)	0.973 (.069)	1.000 (.008)	1.000 (.004)	0.951 (.107)
	(R)		0.997 (.048)	0.970 (.070)	1.000 (.008)	1.000 (.004)	0.948 (.109)

Table 5.3: Estimates of α_1 (Case: $\beta_2 = 0$) — $U(-\sqrt{3}, \sqrt{3})$ Error

n	Estimated Model		(1) A	(2) B	(3) B	(4) B	(5) C
		$\alpha_1 \setminus \beta_1$	0.0	0.0	1.0	2.0	0.0
20	(O)	0.6	0.538 (.212)	0.439 (.273)	0.477 (.227)	0.528 (.159)	0.343 (.345)
	(N)		0.593 (.222)	0.599 (.271)	0.598 (.224)	0.598 (.155)	0.603 (.321)
	(X)		0.565 (.228)	0.578 (.266)	0.582 (.225)	0.588 (.154)	0.577 (.313)
	(U)		0.592 (.224)	0.594 (.273)	0.595 (.226)	0.596 (.156)	0.595 (.323)
	(R)		0.595 (.223)	0.599 (.271)	0.599 (.225)	0.599 (.155)	0.602 (.322)
	(O)	0.9	0.814 (.185)	0.663 (.310)	0.838 (.117)	0.883 (.049)	0.525 (.438)
	(N)		0.888 (.173)	0.840 (.234)	0.902 (.105)	0.900 (.047)	0.806 (.302)
	(X)		0.872 (.183)	0.822 (.243)	0.894 (.105)	0.898 (.046)	0.779 (.311)
	(U)		0.889 (.174)	0.838 (.237)	0.903 (.106)	0.901 (.047)	0.803 (.306)
	(R)		0.891 (.173)	0.840 (.235)	0.904 (.106)	0.900 (.047)	0.807 (.303)
	(O)	1.0	0.915 (.171)	0.752 (.313)	0.983 (.048)	0.996 (.021)	0.550 (.504)
	(N)		0.986 (.150)	0.900 (.218)	1.007 (.052)	1.000 (.021)	0.826 (.329)
	(X)		0.975 (.159)	0.886 (.229)	1.000 (.049)	0.999 (.021)	0.799 (.344)
	(U)		0.988 (.151)	0.900 (.220)	1.008 (.052)	1.000 (.021)	0.823 (.334)
	(R)		0.988 (.150)	0.901 (.218)	1.007 (.052)	1.000 (.021)	0.826 (.330)
40	(O)	0.6	0.568 (.140)	0.522 (.164)	0.532 (.149)	0.551 (.120)	0.476 (.195)
	(N)		0.599 (.143)	0.599 (.160)	0.598 (.144)	0.598 (.116)	0.600 (.177)
	(X)		0.585 (.143)	0.592 (.158)	0.592 (.144)	0.594 (.116)	0.593 (.174)
	(U)		0.596 (.143)	0.596 (.161)	0.597 (.145)	0.597 (.116)	0.598 (.178)
	(R)		0.598 (.143)	0.598 (.160)	0.598 (.145)	0.598 (.116)	0.600 (.177)
	(O)	0.9	0.855 (.107)	0.784 (.167)	0.862 (.071)	0.888 (.033)	0.713 (.230)
	(N)		0.898 (.100)	0.887 (.132)	0.900 (.062)	0.900 (.030)	0.877 (.163)
	(X)		0.889 (.103)	0.879 (.134)	0.899 (.063)	0.899 (.030)	0.867 (.165)
	(U)		0.897 (.100)	0.886 (.133)	0.901 (.063)	0.900 (.030)	0.876 (.164)
	(R)		0.897 (.100)	0.887 (.132)	0.900 (.063)	0.900 (.030)	0.877 (.163)
	(O)	1.0	0.957 (.089)	0.871 (.165)	0.996 (.015)	0.999 (.007)	0.759 (.275)
	(N)		0.996 (.075)	0.953 (.110)	1.001 (.016)	1.000 (.007)	0.915 (.170)
	(X)		0.991 (.078)	0.948 (.114)	1.000 (.015)	1.000 (.007)	0.907 (.176)
	(U)		0.995 (.075)	0.953 (.110)	1.001 (.016)	1.000 (.007)	0.915 (.171)
	(R)		0.995 (.075)	0.953 (.110)	1.001 (.016)	1.000 (.007)	0.915 (.170)
60	(O)	0.6	0.580 (.110)	0.551 (.123)	0.556 (.114)	0.566 (.098)	0.520 (.141)
	(N)		0.601 (.111)	0.602 (.120)	0.603 (.112)	0.603 (.096)	0.601 (.128)
	(X)		0.593 (.111)	0.596 (.119)	0.597 (.111)	0.598 (.095)	0.596 (.127)
	(U)		0.600 (.112)	0.601 (.120)	0.601 (.112)	0.601 (.096)	0.599 (.128)
	(R)		0.600 (.111)	0.600 (.120)	0.601 (.112)	0.601 (.096)	0.600 (.128)
	(O)	0.9	0.871 (.077)	0.827 (.113)	0.871 (.056)	0.890 (.028)	0.781 (.153)
	(N)		0.901 (.073)	0.900 (.094)	0.901 (.050)	0.900 (.026)	0.897 (.115)
	(X)		0.895 (.075)	0.894 (.095)	0.900 (.050)	0.900 (.026)	0.891 (.115)
	(U)		0.901 (.073)	0.900 (.094)	0.901 (.050)	0.900 (.026)	0.896 (.115)
	(R)		0.900 (.073)	0.899 (.094)	0.901 (.050)	0.900 (.026)	0.896 (.115)
	(O)	1.0	0.971 (.058)	0.913 (.111)	0.998 (.008)	1.000 (.004)	0.837 (.186)
	(N)		0.998 (.048)	0.969 (.072)	1.000 (.008)	1.000 (.004)	0.947 (.112)
	(X)		0.995 (.050)	0.966 (.075)	1.000 (.008)	1.000 (.004)	0.942 (.115)
	(U)		0.998 (.048)	0.970 (.072)	1.000 (.008)	1.000 (.004)	0.946 (.112)
	(R)		0.997 (.048)	0.969 (.073)	1.000 (.008)	1.000 (.004)	0.946 (.112)

bias and the OLSE of α_1 estimated by Model A yields the smallest one (see, for example, Andrews (1993)). That is, OLSE bias of the AR(1) coefficient increases as the number of exogenous variables increases. In order to check this fact, we compare (1), (2) and (5) with respect to (O). Note that (O) represents the arithmetic average and the root mean square error from the 10^4 OLSEs. For (O) in Table 5.1, in the case of $n = 20$ and $\alpha_1 = 0.6$, (1) is 0.541, (2) is 0.441 and (5) is 0.341. For all the cases of $n = 20, 40, 60$ and $\alpha_1 = 0.6, 0.9, 1.0$ in Tables 5.1 – 5.3, bias of (O) increases as the number of exogenous variables increases.

Next, we compare (2) – (4), taking the case $n = 20$ in Table 5.1, where the true model is Model A or B while the estimated model is Model B. We examine whether the intercept influences precision of OLSE. The results are as follows. When the intercept increases, the OLSE approaches the true parameter value and in addition the root mean square error of the OLSE becomes small. Taking an example of $n = 20$ and $\alpha_1 = 0.6$ in (O) of Table 5.1, the arithmetic averages are given by 0.441 for (2), 0.479 for (3) and 0.530 for (4), and the root mean squared errors are 0.269 for (2), 0.224 for (3) and 0.158 for (4). Thus, as the intercept increases, the better OLSE is obtained. The same results are obtained for (O) in both Tables 5.2 and 5.3.

The error term is assumed to be normal in Table 5.1, chi-squared in Table 5.2 and uniform in Table 5.3. OLSE is distribution-free, but it is observed from Tables 5.1 – 5.3 that the bias of OLSE depends on the underlying distribution of the error term. That is, the OLSE with the chi-squared error yields the smallest bias and the smallest RMSE of the three. The normal error (Table 5.1) is similar to the uniform error (Table 5.3), but both give us larger bias and larger RMSE than the chi-squared error. See (O) in each table.

We have focused only on (O) in Tables 5.1 – 5.3. Now, we compare (N), (X), (U) and (R). In Table 5.1, under the condition that the true distribution of the error term is normal, we compute the unbiased estimate of the AR(1) coefficient assuming that the error term follows the normal distribution (N), the chi-square distribution (X), the uniform distribution (U) and the residual-based distribution (R). Accordingly, it might be expected that (N) shows the best performance, because the estimated model is consistent with the underlying true one. Similarly, it is expected that (X) is better than any other procedures in Table 5.2 and (U) shows the best performance in Table 5.3. That is, summarizing the expected results, the best estimator should be (N) in Table 5.1, (X) in Table 5.2 and (U) in Table 5.3. In Table 5.1, (N), (U) and (R) are very similar to each other, but (X) indicates the poor estimator. In Table 5.2, (N) and (U) are over-estimated in the case of (1). Remember that (X) should be the best because the underlying assumption of the error is chi-squared. In Table 5.3, (N) and (R) are slightly better than (U), but (X) is the worst estimator. In any case, we can see from the tables that (O) is biased while all of (N), (X), (U) and (R) are bias-corrected. Through Tables 5.1 – 5.3, it might be concluded that (R) is robust for all the cases. That is, (R) shows a relatively good performance for any underlying distribution of the error terms. In other words, even if we do not know the true distribution of the error term (this case is common), (R) indicates quite good results. Thus, we could verify

that the true model is obtained from the estimated model in the case of inclusion of irrelevant variables.

5.4.2 AR(p) Models

Next, we consider the AR(p) models, where $p = 2, 3$ is taken. Assume that the true model is represented by Model A, i.e.,

$$y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \cdots + \alpha_p y_{t-p} + u_t,$$

for $t = p+1, p+2, \cdots, n$, where the initial values are set to be $y_1 = y_2 = \cdots = y_p = 0$. In Section 5.4.1 we have examined the four kinds of distributions for u_t but in this section we consider the normal and uniform errors for the AR(2) model and the normal error for the AR(3) model. The above AR(p) model can be rewritten as:

$$(1 - \lambda_1 L)(1 - \lambda_2 L) \cdots (1 - \lambda_p L) y_t = u_t,$$

where L denotes the lag operator, i.e., $Ly_t = y_{t-1}$. λ_1, λ_2, $\cdots$, λ_p are assumed to be real numbers. (O) is compared with (N) and (R) in this section, i.e., (U) and (X) are omitted to save space. Taking the cases of $p = 2, 3$, we estimate the true model by Model A. However, the true model is not necessarily equivalent to the estimated model. The case of $\lambda_1 \neq 0$ and $\lambda_2 = \cdots = \lambda_p = 0$ implies that the true model is AR(1) but the estimated one is AR(p). The results are in Table 5.4 for AR(2) and Table 5.5 for AR(3), where the arithmetic averages and the root mean squares errors from the 10^4 coefficient estimates of α_1, α_2 and α_3 are shown. As in Tables 5.1 – 5.3, the values in the parentheses in Tables 5.4 and 5.5 denote the root mean square errors from the 10^4 estimates. The sample size is taken as $n = 20, 40, 60$ in Table 5.4 and $n = 20, 40$ in Table 5.5. As shown in Tables 5.1 – 5.3, the cases of $n = 40, 60$ are similar to those of $n = 20$ except that the former cases are less biased than the latter ones. We examine the cases where λ_i takes 0.0, 0.5 or 1.0 for $i = 1, 2, 3$ and $\lambda_1 \geq \lambda_2 \geq \lambda_3$ holds. In Table 5.4, (a), (b), (c), (d), (e) and (f) denote the case of $(\alpha_1, \alpha_2) = (0, 0)$, $(.5, 0)$, $(1, 0)$, $(1, -.25)$, $(1.5, -.5)$ and $(2, -1)$, respectively. Similarly, in Table 5.5, (a), (b), (c), (d), (e), (f), (g), (h), (i) and (j) correspond to the case of $(\alpha_1, \alpha_2, \alpha_3) = (0, 0, 0)$, $(.5, 0, 0)$, $(1, 0, 0)$, $(1, -.25, 0)$, $(1.5, -.5, 0)$, $(2, -1, 0)$, $(1.5, -.75, 1.25)$, $(2, -1.25, .25)$, $(2.5, -2, .5)$ and $(3, -3, 1)$, respectively.

Under the assumption that the error term u_t is normally or uniformly distributed in the true model, we obtain the unbiased estimates using (N) and (R) shown in Section 5.3.

In Table 5.4, the root mean square errors of (N) and (R) are smaller than those of (O) in the case of (f), i.e., $(\alpha_1, \alpha_2) = (2, -1)$ or $\lambda_1 = \lambda_2 = 1$. For all the estimates of α_1 and α_2, the arithmetic averages of (N) and (R) are closer to the true parameter values than those of (O). Therefore, it might be concluded that the OLSE bias is corrected by the suggested estimators. Thus, in the case of the AR(2) models, we obtain the same

Table 5.4: AR(2) Model: $N(0,1)$ and $U(-\sqrt{3}, \sqrt{3})$ Errors for $n = 20, 40, 60$

	$n = 20$		$n = 40$		$n = 60$	
	Est. of α_1	Est. of α_2	Est. of α_1	Est. of α_2	Est. of α_1	Est. of α_2
***N*(0, 1) Error**						
(O)	−0.003 (.251)	−0.061 (.247)	−0.001 (.168)	−0.030 (.166)	0.000 (.135)	−0.021 (.133)
(a) (N)	−0.004 (.265)	−0.007 (.278)	0.000 (.173)	−0.002 (.177)	0.000 (.138)	−0.001 (.138)
(R)	−0.005 (.265)	−0.009 (.278)	−0.001 (.172)	−0.005 (.176)	0.001 (.138)	−0.004 (.138)
(O)	0.471 (.254)	−0.059 (.248)	0.486 (.169)	−0.029 (.167)	0.491 (.136)	−0.020 (.134)
(b) (N)	0.496 (.266)	−0.006 (.279)	0.501 (.173)	−0.002 (.177)	0.500 (.138)	−0.001 (.140)
(R)	0.495 (.266)	−0.007 (.279)	0.499 (.173)	−0.003 (.177)	0.500 (.138)	−0.004 (.139)
(O)	0.938 (.258)	−0.031 (.256)	0.970 (.172)	−0.015 (.169)	0.980 (.137)	−0.011 (.136)
(c) (N)	0.994 (.271)	−0.009 (.289)	1.001 (.176)	−0.006 (.179)	1.001 (.140)	−0.004 (.141)
(R)	0.995 (.272)	−0.009 (.289)	1.000 (.176)	−0.005 (.179)	1.001 (.140)	−0.005 (.141)
(O)	0.943 (.251)	−0.267 (.236)	0.973 (.165)	−0.258 (.160)	0.982 (.132)	−0.256 (.129)
(d) (N)	0.995 (.258)	−0.253 (.270)	1.000 (.167)	−0.251 (.172)	1.000 (.133)	−0.250 (.135)
(R)	0.995 (.258)	−0.254 (.270)	0.998 (.167)	−0.251 (.172)	1.000 (.133)	−0.252 (.135)
(O)	1.399 (.253)	−0.447 (.247)	1.450 (.159)	−0.473 (.154)	1.467 (.125)	−0.482 (.121)
(e) (N)	1.492 (.250)	−0.500 (.269)	1.501 (.157)	−0.504 (.161)	1.501 (.124)	−0.503 (.125)
(R)	1.493 (.251)	−0.501 (.270)	1.500 (.157)	−0.503 (.161)	1.500 (.123)	−0.502 (.125)
(O)	1.861 (.247)	−0.863 (.265)	1.928 (.127)	−0.928 (.131)	1.950 (.087)	−0.950 (.089)
(f) (N)	1.982 (.208)	−0.983 (.235)	1.995 (.101)	−0.995 (.106)	1.996 (.067)	−0.996 (.070)
(R)	1.983 (.208)	−0.984 (.235)	1.995 (.101)	−0.995 (.107)	1.996 (.068)	−0.996 (.070)
***U*(−√3, √3) Error**						
(O)	−0.001 (.257)	−0.058 (.249)	−0.001 (.169)	−0.028 (.167)	−0.001 (.135)	−0.017 (.133)
(a) (N)	−0.002 (.271)	−0.004 (.282)	0.000 (.173)	0.000 (.178)	−0.001 (.138)	0.003 (.139)
(R)	−0.002 (.271)	−0.002 (.283)	−0.001 (.173)	−0.001 (.178)	−0.001 (.138)	0.000 (.139)
(O)	0.472 (.260)	−0.058 (.250)	0.486 (.170)	−0.028 (.166)	0.490 (.136)	−0.016 (.134)
(b) (N)	0.497 (.273)	−0.005 (.283)	0.500 (.174)	−0.001 (.177)	0.499 (.138)	0.003 (.140)
(R)	0.498 (.274)	−0.004 (.283)	0.499 (.174)	−0.001 (.177)	0.499 (.138)	0.001 (.140)
(O)	0.939 (.265)	−0.033 (.261)	0.969 (.172)	−0.016 (.169)	0.980 (.137)	−0.009 (.135)
(c) (N)	0.995 (.279)	−0.011 (.294)	1.000 (.177)	−0.007 (.180)	1.000 (.140)	−0.003 (.141)
(R)	0.997 (.280)	−0.012 (.295)	1.000 (.177)	−0.006 (.180)	1.000 (.140)	−0.003 (.141)
(O)	0.945 (.257)	−0.269 (.240)	0.973 (.166)	−0.258 (.158)	0.982 (.133)	−0.253 (.129)
(d) (N)	0.998 (.265)	−0.255 (.274)	1.001 (.168)	−0.252 (.170)	1.000 (.134)	−0.248 (.136)
(R)	1.000 (.266)	−0.256 (.275)	1.000 (.168)	−0.251 (.171)	0.999 (.134)	−0.249 (.136)
(O)	1.400 (.257)	−0.448 (.250)	1.450 (.160)	−0.474 (.154)	1.468 (.125)	−0.483 (.122)
(e) (N)	1.493 (.255)	−0.501 (.273)	1.501 (.158)	−0.505 (.162)	1.502 (.124)	−0.504 (.125)
(R)	1.496 (.256)	−0.505 (.274)	1.501 (.158)	−0.504 (.162)	1.502 (.124)	−0.504 (.125)
(O)	1.861 (.248)	−0.863 (.265)	1.926 (.128)	−0.927 (.131)	1.952 (.086)	−0.952 (.087)
(f) (N)	1.982 (.209)	−0.983 (.236)	1.993 (.100)	−0.993 (.106)	1.997 (.067)	−0.998 (.069)
(R)	1.986 (.208)	−0.987 (.236)	1.994 (.100)	−0.994 (.106)	1.997 (.067)	−0.997 (.070)

Note that the true value of (α_1, α_2) is given by $(0, 0)$ for (a), $(.5, 0)$ for (b), $(1, 0)$ for (c), $(1, -.25)$ for (d), $(1.5, -.5)$ for (e) and $(2, -1)$ for (f).

Table 5.5: AR(3) Models: $N(0,1)$ Error for $n = 20, 40$

		$n = 20$			$n = 40$		
		Est. of α_1	Est. of α_2	Est. of α_3	Est. of α_1	Est. of α_2	Est. of α_3
(a)	(O)	−0.003 (.263)	−0.061 (.264)	−0.003 (.264)	0.000 (.174)	−0.031 (.170)	0.000 (.167)
	(N)	−0.001 (.279)	−0.002 (.299)	−0.003 (.321)	0.002 (.178)	−0.002 (.181)	0.000 (.185)
	(R)	−0.007 (.279)	−0.006 (.299)	−0.006 (.320)	0.000 (.178)	−0.005 (.181)	0.000 (.185)
(b)	(O)	0.469 (.266)	−0.059 (.286)	0.000 (.263)	0.487 (.175)	−0.031 (.189)	0.002 (.168)
	(N)	0.499 (.281)	−0.002 (.327)	−0.005 (.317)	0.502 (.179)	−0.003 (.201)	0.001 (.186)
	(R)	0.493 (.281)	−0.003 (.326)	−0.007 (.317)	0.500 (.179)	−0.005 (.201)	0.002 (.186)
(c)	(O)	0.934 (.279)	−0.054 (.348)	0.031 (.278)	0.971 (.179)	−0.031 (.236)	0.016 (.174)
	(N)	0.996 (.289)	−0.005 (.398)	−0.010 (.328)	1.003 (.182)	−0.005 (.252)	−0.004 (.189)
	(R)	0.991 (.289)	−0.003 (.397)	−0.008 (.327)	1.001 (.182)	−0.006 (.251)	−0.001 (.189)
(d)	(O)	0.942 (.271)	−0.269 (.335)	0.003 (.266)	0.974 (.176)	−0.262 (.229)	0.003 (.170)
	(N)	0.999 (.284)	−0.252 (.393)	−0.006 (.318)	1.002 (.180)	−0.254 (.248)	0.001 (.187)
	(R)	0.994 (.284)	−0.249 (.391)	−0.007 (.318)	1.000 (.180)	−0.254 (.247)	0.003 (.187)
(e)	(O)	1.403 (.287)	−0.480 (.416)	0.030 (.286)	1.457 (.181)	−0.496 (.285)	0.016 (.175)
	(N)	1.495 (.291)	−0.503 (.482)	−0.003 (.334)	1.503 (.182)	−0.504 (.307)	−0.002 (.190)
	(R)	1.490 (.291)	−0.499 (.481)	−0.002 (.333)	1.501 (.182)	−0.503 (.306)	−0.001 (.189)
(f)	(O)	1.862 (.305)	−0.875 (.511)	0.011 (.313)	1.936 (.187)	−0.947 (.340)	0.011 (.184)
	(N)	1.989 (.300)	−0.995 (.585)	0.003 (.364)	2.001 (.185)	−1.009 (.364)	0.007 (.198)
	(R)	1.983 (.300)	−0.988 (.583)	0.004 (.363)	1.999 (.185)	−1.005 (.363)	0.006 (.198)
(g)	(O)	1.408 (.279)	−0.690 (.400)	0.103 (.274)	1.458 (.177)	−0.723 (.271)	0.115 (.170)
	(N)	1.499 (.285)	−0.750 (.462)	0.120 (.322)	1.501 (.179)	−0.753 (.292)	0.126 (.187)
	(R)	1.493 (.285)	−0.745 (.461)	0.119 (.322)	1.500 (.179)	−0.752 (.292)	0.126 (.187)
(h)	(O)	1.865 (.297)	−1.108 (.490)	0.217 (.294)	1.939 (.182)	−1.189 (.318)	0.238 (.172)
	(N)	1.993 (.289)	−1.247 (.542)	0.249 (.338)	2.002 (.178)	−1.253 (.334)	0.250 (.187)
	(R)	1.988 (.290)	−1.242 (.542)	0.249 (.337)	2.000 (.178)	−1.251 (.333)	0.249 (.186)
(i)	(O)	2.311 (.324)	−1.692 (.609)	0.376 (.339)	2.410 (.187)	−1.858 (.356)	0.448 (.182)
	(N)	2.485 (.294)	−1.978 (.615)	0.491 (.367)	2.501 (.172)	−2.005 (.350)	0.504 (.187)
	(R)	2.478 (.294)	−1.967 (.613)	0.488 (.365)	2.498 (.172)	−2.000 (.350)	0.502 (.186)
(j)	(O)	2.757 (.359)	−2.493 (.769)	0.728 (.441)	2.871 (.190)	−2.735 (.395)	0.864 (.208)
	(N)	2.975 (.289)	−2.948 (.654)	0.971 (.395)	2.991 (.143)	−2.982 (.304)	0.990 (.164)
	(R)	2.968 (.289)	−2.932 (.652)	0.963 (.393)	2.988 (.143)	−2.976 (.305)	0.988 (.164)

Note that the true value of $(\alpha_1, \alpha_2, \alpha_3)$ is given by $(0, 0, 0)$ for (a), $(.5, 0, 0)$ for (b), $(1, 0, 0)$ for (c), $(1, -.25, 0)$ for (d), $(1.5, -.5, 0)$ for (e), $(2, -1, 0)$ for (f), $(1.5, -.75, .125)$ for (g), $(2, -1.25, .25)$ for (h), $(2.5, -2, .5)$ for (i) and $(3, -3, 1)$ for (j).

results as in the case of the AR(1) models. In addition, the $N(0, 1)$ error is very similar to the $U(-\sqrt{3}, \sqrt{3})$ error.

Next, we examine the AR(3) models and the results are in Table 5.5. For estimation of zero coefficients, all the three estimators are close to the true parameter value. However, for estimation of non-zero coefficients, the suggested estimators are superior to OLSE, which implies that (N) and (R) are less biased than (O).

Moreover, we can see from both Tables 5.4 and 5.5 that as the sample size increases all the estimates approach the true values and the root mean square errors become small.

For all the cases of AR(p) models, $p = 1, 2, 3$, it is shown from Tables 5.1 – 5.5 that OLSE bias is corrected using the estimators introduced in Section 5.3 even if the data generating process is not known. That is, in the case where the true parameter value is zero, the parameter estimate is close to zero. In Table 5.4, the cases of $\alpha_1 \neq 0$ and $\alpha_2 = 0$, i.e., (a), (b) and (c), imply that the data generating process is AR(1). In Table 5.5, the cases of $\alpha_1 \neq 0$ and $\alpha_2 = \alpha_3 = 0$, i.e., (a) and (b), imply that the data generating process is AR(1), and the case of $\alpha_1 \neq 0$, $\alpha_2 \neq 0$ and $\alpha_3 = 0$, i.e., (c), (d) and (e), imply that the data generating process is AR(2). The estimated models are given by AR(2) in Table 5.4 and AR(3) in Table 5.5. We can see from the two tables that zero coefficients are estimated to be almost zeros while non-zero coefficients are estimated to be their true values. The suggested estimator can recover the true parameter values from the OLS estimates even in the case where the unnecessary variables are included in the regression model. Thus, even if the true model is different from the estimated model, we can obtain the bias-corrected coefficient estimate based on the suggested estimators.

5.5 Empirical Example

In this section, based on actually observed annual data from 1960 to 1998, U.S. and Japanese consumption functions are estimated as an empirical example of the estimator suggested above. We consider the AR(1) model with constant term and exogenous variable x_{1t}, which is specified as follows:

$$y_t = \beta_1 + \beta_2 x_{1t} + \alpha_1 y_{t-1} + u_t,$$

where $u_t \sim N(0, \sigma^2)$ is assumed. y_t and x_{1t} represent consumption and income, respectively. We assume that current consumption depends on past consumption as well as income, where we consider that we have a habit persistence effect in consumption.

As for the initial value of y_t, the actual consumption data of 1960 is used. Therefore, the estimation period is from 1961 to 1998. The following consumption and income data are used for y_t and x_{1t}.

- U.S. Data

y_t = Personal Consumption Expenditures
(Billions of Chained (1992) Dollars)

x_{1t} = Disposable Personal Income
(Billions of Chained (1992) Dollars)

- Japanese Data

y_t = Domestic Final Consumption Expenditures of Households
(Billions of Japanese Yen at Market Prices in 1990)

x_{1t} = National Disposable Income of Households
(Billions of Japanese Yen at Market Prices in 1990)

The data are taken from *Economic Report of the President* (United States Government Printing Office, 2001) for U.S. data and the *Annual Report on National Accounts* (Economic and Social Research Institute, Government of Japan, 2000) for Japanese data. In the economic interpretation, β_2 is known as the marginal propensity to consume. Using U.S. and Japanese data, β_1, β_2 and α_1 are estimated, where OLSE is compared with the estimators suggested in Section 5.3.

Under the above setup, the estimation results of U.S. and Japanese consumption functions are in Tables 5.6 and 5.7. (O) denotes OLSE, while (N), (X), (U) and (R) represent the suggested estimator with normal, chi-squared, uniform and residual-based errors. EST and SER indicate the parameter estimates and the corresponding standard errors. 2.5%, 5%, 25%, 50%, 75%, 95% and 97.5% represent the percentiles obtained from N estimates, where $N = 10^4$ is taken. See Section 5.3.2 for derivation of the standard errors (i.e., SER) and percentiles.

From Tables 5.6 and 5.7, we can see skewness of the distribution by computing the distance between 2.5% and 50% values and that between 50% and 97.5% values (or 5.0%, 50% and 95% values). Figures 5.4 and 5.5 represent the empirical distributions obtained from the suggested estimators with (N) and (R), where the first three empirical densities are related to (N) while the last three distributions are (R). $\overline{\beta}_2$ and $\overline{\alpha}_1$ in (N) and (R) of Figures 5.4 and 5.5 take the same scale in the horizontal line. Therefore, it is possible to compare the empirical distributions with respect to width, height, variance and so on. Comparing $\overline{\beta}_2$ and $\overline{\alpha}_1$ in (N) and (R) of Figures 5.4 and 5.5, $\overline{\beta}_2$ and $\overline{\alpha}_1$ in Figure 5.4 are more widely distributed than those in Figure 5.5. In other words, the coefficient estimates of the U.S. consumption function have larger variance than those of the Japanese consumption function. Judging from (N), (X), (U) and (R) in Tables 5.6 and 5.7 and Figures 5.4 and 5.5, the distribution is skewed to the left for $\overline{\beta}_1$ and $\overline{\alpha}_1$, and to the right for $\overline{\beta}_2$. That is, OLSEs of β_1 and α_1 are underestimated while OLSE of β_2 is overestimated. Moreover, comparing SERs, we can observe that all the standard errors obtained from OLSE are overestimated compared with those from the unbiased estimator suggested in this chapter. The standard error obtained from OLSE is computed by the conventional formula, and the standard error from the suggested estimator is based on the simulation technique. Since OLSE is biased, it might be appropriate to consider that the standard error from OLSE is also biased.

Table 5.6: U.S. Consumption Function: 1961 – 1998

		EST	SER	2.5%	5%	25%	50%	75%	95%	97.5%
(O)	$\hat{\beta}_1$	−25.574	29.615							
	$\hat{\beta}_2$	0.334	0.109							
	$\hat{\alpha}_1$	0.655	0.121							
(N)	$\overline{\beta}_1$	−14.690	31.757	−84.369	−70.759	−34.660	−12.070	8.170	32.579	39.226
	$\overline{\beta}_2$	0.282	0.092	0.110	0.135	0.218	0.277	0.341	0.437	0.469
	$\overline{\alpha}_1$	0.712	0.100	0.505	0.542	0.649	0.719	0.781	0.870	0.899
(X)	$\overline{\beta}_1$	−16.558	30.395	−79.973	−67.977	−34.717	−15.692	3.460	30.212	39.934
	$\overline{\beta}_2$	0.287	0.085	0.134	0.158	0.229	0.280	0.336	0.432	0.466
	$\overline{\alpha}_1$	0.706	0.092	0.511	0.548	0.653	0.714	0.770	0.846	0.871
(U)	$\overline{\beta}_1$	−15.185	32.026	−85.210	−73.403	−35.371	−12.616	7.952	33.221	39.891
	$\overline{\beta}_2$	0.282	0.094	0.102	0.131	0.217	0.277	0.341	0.441	0.472
	$\overline{\alpha}_1$	0.712	0.102	0.503	0.539	0.649	0.717	0.781	0.876	0.906
(R)	$\overline{\beta}_1$	−14.190	32.567	−86.981	−73.727	−33.962	−10.578	9.021	33.533	40.055
	$\overline{\beta}_2$	0.282	0.093	0.110	0.135	0.218	0.277	0.340	0.439	0.473
	$\overline{\alpha}_1$	0.711	0.101	0.504	0.540	0.648	0.717	0.782	0.870	0.897

Table 5.7: Japanese Consumption Function: 1961 – 1998

		EST	SER	2.5%	5%	25%	50%	75%	95%	97.5%
(O)	$\hat{\beta}_1$	3894.2	1693.6							
	$\hat{\beta}_2$	0.171	0.068							
	$\hat{\alpha}_1$	0.803	0.076							
(N)	$\overline{\beta}_1$	4398.4	1507.5	984.0	1719.9	3447.5	4502.8	5445.5	6567.0	6911.6
	$\overline{\beta}_2$	0.133	0.060	0.025	0.042	0.092	0.128	0.168	0.235	0.262
	$\overline{\alpha}_1$	0.847	0.069	0.697	0.728	0.807	0.853	0.895	0.951	0.972
(X)	$\overline{\beta}_1$	4307.4	1425.5	1547.2	2039.5	3371.0	4258.7	5178.8	6614.1	7128.9
	$\overline{\beta}_2$	0.135	0.062	0.035	0.050	0.094	0.127	0.167	0.247	0.282
	$\overline{\alpha}_1$	0.844	0.072	0.673	0.714	0.808	0.854	0.892	0.942	0.959
(U)	$\overline{\beta}_1$	4362.9	1485.3	1109.9	1714.5	3405.9	4442.3	5381.2	6586.2	6913.3
	$\overline{\beta}_2$	0.134	0.060	0.024	0.040	0.092	0.129	0.170	0.236	0.261
	$\overline{\alpha}_1$	0.846	0.070	0.699	0.727	0.804	0.852	0.895	0.954	0.973
(R)	$\overline{\beta}_1$	4388.0	1495.2	982.3	1691.8	3488.6	4491.3	5397.3	6544.8	6886.9
	$\overline{\beta}_2$	0.135	0.059	0.027	0.043	0.094	0.130	0.169	0.236	0.262
	$\overline{\alpha}_1$	0.845	0.068	0.696	0.729	0.806	0.851	0.892	0.950	0.969

Figure 5.4: U.S. Consumption Function

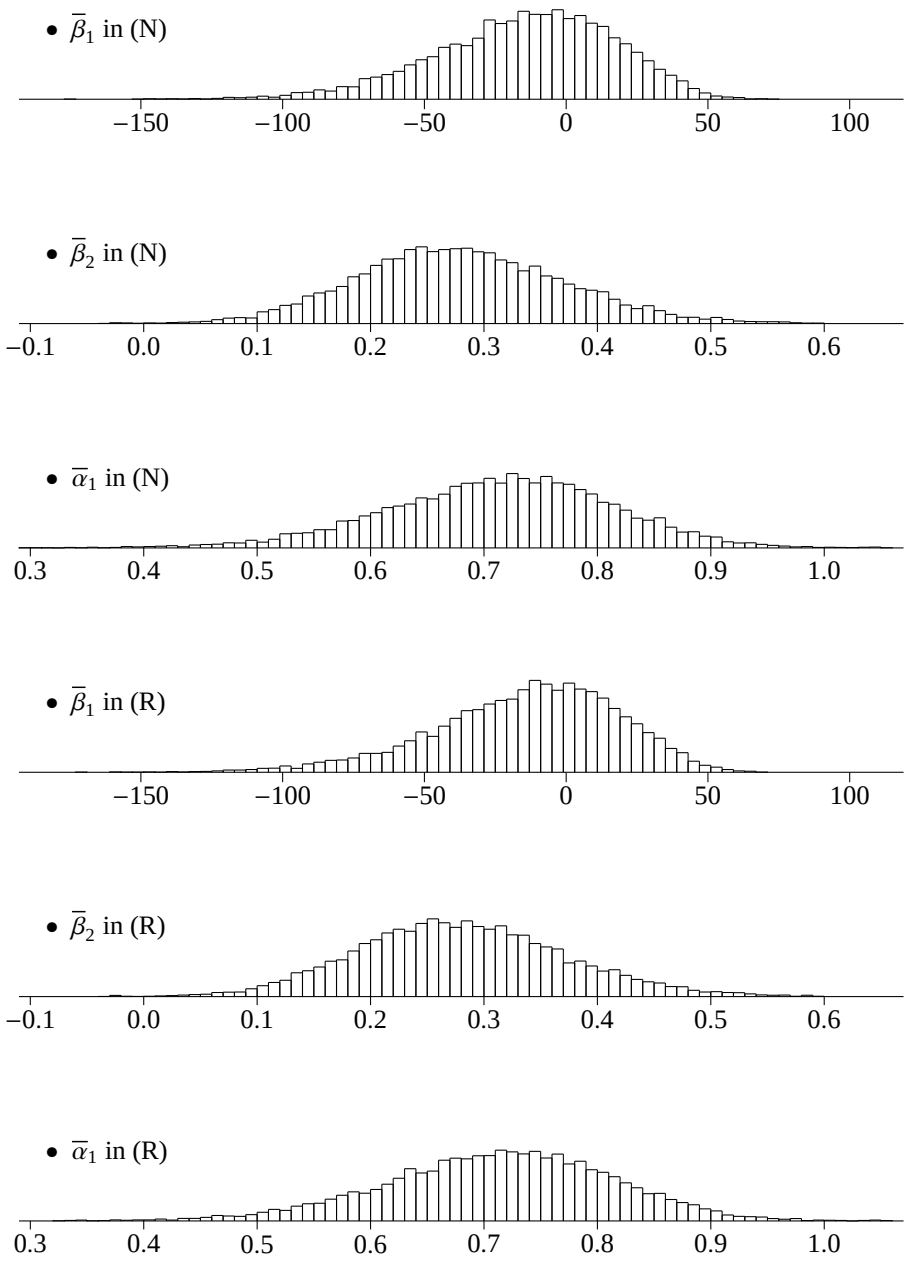
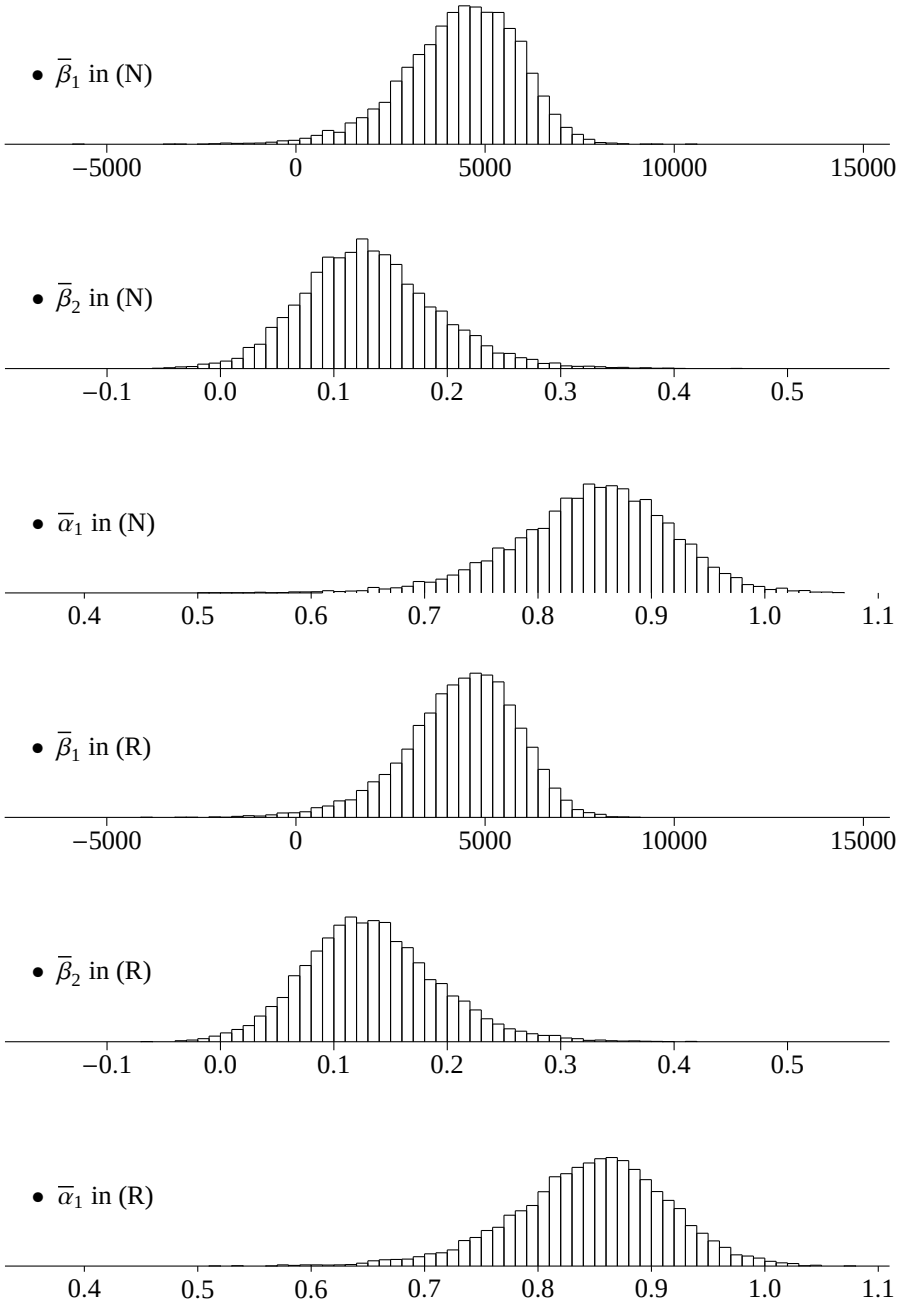


Figure 5.5: Japanese Consumption Function



5.6 Summary

It is well known that the OLS estimates are biased when the autoregressive terms are included in the explanatory variables. In this chapter, we have discussed the bias correction method using the simulation techniques, where the bootstrap methods are applied. We obtain the unbiased estimate of regression coefficient θ, i.e., $\overline{\theta}$, which is the θ such that the OLSE computed from the original data is equal to the arithmetic average of the OLSEs obtained from the simulated data given the unbiased estimate $\overline{\theta}$. When we simulate the data series, we need to assume a distribution of the error term. Since the underlying true distribution of the error term is not known, the four types of errors are examined, i.e., the normal error (N), the chi-squared error (X), the uniform error (U) and the residual-based error (R). Because the residual-based approach is distribution-free, it is easily expected that (R) shows a good performance for any simulation study. From the simulation studies shown in Tables 5.1 – 5.3, (N), (U) and (R) are very similar, but we have the case where (N) and (U) are overestimated (see Table 5.2). Therefore, as it is expected, we can conclude that (R) is the best estimator.

Finally, we have shown estimation of U.S. and Japanese consumption functions as an empirical example. The AR(1) model which includes both a constant term and an exogenous variable x_{1t} has been estimated, where the standard errors and the percentiles are shown. The OLSE applied to the AR model has large variance, compared with the suggested unbiased estimator. In addition, the empirical distributions are not symmetric. Therefore, inference based on OLSE causes us wrong results. Thus, using the unbiased estimator discussed in this chapter, we can obtain the appropriate confidence interval as well as the unbiased estimate of the regression coefficient.

As a final remark, note as follows. In empirical studies, we conventionally assume the normal error, but we never assume the chi-square and uniform errors. However, the true distribution of the error term is not known. Therefore, even the normality assumption on the error term is not plausible. We can see that the empirical results shown in Tables 5.6 and 5.7 are quite different, depending on the underlying distribution on the error term. We have no *ad hoc* assumption in (R), which is the residual-based and distribution-free procedure. Thus, the unbiased estimator with (R) might be preferred in practice. However, (R) has the problem that the residuals are not mutually independent. The procedure introduced in this chapter regards the residuals as the mutually independently distributed random draws.

5.7 Appendix: Source Code

The subroutine used in Sections 5.4.1 and 5.4.2 is shown in this appendix. `l` and `n` denote the sample period, where `l` is the starting point and `n` is the end point. `k` indicates the number of exogenous variables except for the lagged dependent variables such as $y_{t-1}, y_{t-2}, \cdots, y_{t-p}$. `lag` represents the number of the lagged dependent variables, i.e.,

p corresponds to `lag`. We have to assume the random draws u^*_{p+1}, u^*_{p+2}, $\cdots$, u^*_n in Step 2 of the estimation procedure discussed in Section 5.3. In the source code, the distribution of the random draw u^*_t is given by `idist`, where u^*_t is assumed to be (N) when `idist=1`, (X) when `idist=2`, (U) when `idist=3` and (R) when `idist=4` (see Lines 43 – 62). `z(m)` indicates the dependent variable, while `x(m,k+lag)` denotes the explanatory variables. We have to put the exogenous variables, i.e., X_t, in the first `k` rows of `x(m,k+lag)` and the lagged dependent variables, i.e., y_{t-1}, y_{t-2}, $\cdots$, y_{t-p}, in the last `lag` rows of `x(m,k+lag)`.

The input variables are given by `z(m)`, `x(m,k)`, `l`, `n`, `k`, `lag` and `idist`. The output variables are `olse(k+lag)`, `beta(k+lag)` and `se`. In `olse(k+lag)`, the OLSE based on the dependent variable `z(m)` and the explanatory variables `x(m,k+lag)`, i.e., $\hat{\theta}$, is stored. Note that the explanatory variables consist of the exogenous variables and the lagged dependent variables. The unbiased estimator suggested in Chapter 5, i.e., $\overline{\theta}$, is given by `beta(k+lag)` and the standard error of the regression, i.e., $\overline{\sigma}$ in equation 5.6, is `se`.

unbiased(z,x,l,n,k,lag,olse,beta,se,idist)

```
1: c =======================================
2:       subroutine unbiased(z,x,l,n,k,lag,olse,beta,se,idist)
3:       parameter (number=10000,converge=0.001,const=0.9)
4:       dimension y(1000),x(1000,15),olse(15),b(15),z(1000)
5:       dimension bmean(15),beta(15),beta1(15),resid(1000)
6: c
7: c   Input:
8: c     z(m):    Dependent Variable
9: c     x(m,k):  Exogenous Variables
10: c     l - n:   Sample Period
11: c     k:       # of Exogenous Variables
12: c     lag:     # of Lagged Dependent Variables
13: c     idist:   Assumption on Error Term
14: c                1: Normal, 2: Chi^2, 3: Uniform, 4: Residual
15: c   Output:
16: c     olse(k+lag): OLSE
17: c     beta(k+lag): Unbiased Estimate
18: c     se:          Standard Error of Regression
19: c
20:        do 1 i=1,lag
21:         do 1 m=l,n
22:     1 if(m-i.ge.1) x(m,k+i)=z(m-i)
23:       call ols(z,x,l,n,k+lag,olse)
24:       call ser(z,x,l,n,k+lag,olse,se,resid)
25:        do 2 i=1,k+lag
26:       y(i)=z(i)
27:       beta(i)=olse(i)
28:     2 beta1(i)=olse(i)
29:       ix=1
30:       iy=3
31:         do 3 i=1,1000
32:     3 call urnd(ix,iy,rn)
33:       ix1=ix
34:       iy1=iy
35:       pin=1.0
36:        do 4 iter=1,1000
```

```
        do 5 i=1,k+lag
    5 bmean(i)=0.0
      ix=ix1
      iy=iy1
       do 6 ir=1,number
        do 7 m=l,n
          if(idist.eq.1) then
c Normal
      call snrnd(ix,iy,rn)
      y(m)=se*rn
          else if(idist.eq.2) then
c Chi^2
      call snrnd(ix,iy,rn)
      rn=(rn*rn-1.)/sqrt(2.)
      y(m)=se*rn
          else if(idist.eq.3) then
c Uniform
      call urnd(ix,iy,rn)
      rn=2.*sqrt(3.)*(rn-.5)
      y(m)=se*rn
          else if(idist.eq.4) then
c Residual
      call urnd(ix,iy,rn)
      i1=int( rn*float(n-l+1) )+l
      y(m)=resid(i1)*sqrt(float(n-l+1)/float(n-l+1-k-lag))
          endif
         do 8 i=1,lag
    8 x(m,k+i)=y(m-i)
          do 9 i=1,k+lag
    9 y(m)=y(m)+beta(i)*x(m,i)
    7 continue
      call ols(y,x,l,n,k+lag,b)
        do 10 i=1,k+lag
   10 bmean(i)=bmean(i)+b(i)/float(number)
    6 continue
        do 11 i=1,k+lag
   11 beta(i)=beta(i)+pin*( olse(i)-bmean(i) )
        do 12 i=1,k+lag
   12 if( abs(beta1(i)-beta(i)).gt.converge ) go to 13
      if( iter.ne.1 ) go to 14
   13   do 15 i=1,k+lag
   15 beta1(i)=beta(i)
    4 pin=const*pin
   14 call ser(z,x,l,n,k+lag,olse,se,resid)
      return
      end
c =======================================
      subroutine ser(y,x,l,n,k,b,se,resid)
      dimension y(1000),x(1000,15),b(15),resid(1000)
      ave=0.
      var=0.
        do 1 m=l,n
      h=0.
         do 2 i=1,k
    2 h=h+x(m,i)*b(i)
      resid(m)=y(m)-h
      ave=ave+resid(m)/float(n-l+1-k)
    1 var=var+resid(m)*resid(m)/float(n-l+1-k)
      se=sqrt(var)
        do 3 m=l,n
    3 resid(m)=resid(m)-ave
      return
      end
```

```
100: C ========================================
101:       subroutine ols(y,x,l,n,k,b)
102:       dimension y(1000),x(1000,15),b(15),xy(15),xx(15,15)
103:         do 1 i=1,k
104:          do 1 j=1,k
105:       xx(i,j)=0.
106:             do 1 m=l,n
107:     1 xx(i,j)=xx(i,j)+x(m,i)*x(m,j)
108:         do 2 i=1,k
109:       xy(i)=0.
110:             do 2 m=l,n
111:     2 xy(i)=xy(i)+x(m,i)*y(m)
112:       call inverse(xx,k)
113:         do 3 i=1,k
114:       b(i)=0.
115:            do 3 j=1,k
116:     3 b(i)=b(i)+xx(i,j)*xy(j)
117:       return
118:       end
119: C ========================================
120:       subroutine inverse(x,k)
121:       dimension x(15,15)
122:        do 1 i=1,k
123:       a1=x(i,i)
124:       x(i,i)=1.0
125:         do 2 j=1,k
126:     2 x(i,j)=x(i,j)/a1
127:            do 3 m=1,k
128:       if(m.eq.i) go to 3
129:       a2=x(m,i)
130:       x(m,i)=0.0
131:             do 4 j=1,k
132:     4 x(m,j)=x(m,j)-a2*x(i,j)
133:     3   continue
134:     1  continue
135:       return
136:       end
```

snrnd in Lines 45 and 49 and urnd in Lines 32, 54 and 59 (urnd is used in snrnd, too) have to be used together with the above source code.

In inverse(x,k) of Lines 119 – 136, the input variables are x(i,j) and k, and the output variable is given by x(i,j). That is, the original matrix x(i,j) is replaced by the inverse matrix of x(i,j). inverse(x,k) is utilized by ols(y,x,l,n,k,b) in Line 112. In ols(y,x,l,n,k,b) of Lines 100 – 118, given data y and x, the estimation period from l to n and the number of explanatory variables (i.e., k), OLSE is stored in b. In ser(y,x,l,n,k,b,se,resid) of Lines 83 – 99, given y, x, l, n, k and b, the standard error of the regression, se, and a vector of the residuals, resid, are obtained. Note that b denotes an estimate, which is not necessarily OLSE. In Lines 93, 96 and 97, resid(m) is recomputed in order to satisfy the condition that the sum of resid(m) with respect to m=l,l+1,...,n is equal to zero.

In Lines 20 – 24, OLSE of the regression coefficients and the standard error of the regression are obtained using the observed data. In Lines 29 – 32, the first 1000 random draws are discarded because of stability of the random draws. iter in the loop from Line 36 to Line 79 corresponds to the superscript (j). number in the loop from

Line 41 to 71 is equivalent to N in Section 5.3. In Lines 42 – 67, the simulated data `y(m)` are constructed given the parameter value `beta(i)` and the exogenous variables `x(m,i)` for `i=1,2,...,k`, where the distribution of the error terms is assumed to be (N), (X), (U) or (R), which is determined by `idist`. In Line 68, using the simulated data `y` and `x`, OLSE is obtained as `b`. In Lines 69 and 70, the arithmetic average from the N OLSEs is obtained as `bmean(i)`. In Lines 72 and 73, `beta(i)` is updated, which corresponds to $\overline{\theta}^{(j+1)}$. In Lines 74 and 75, convergence of `beta(i)` is checked for each element of `i`. If all the elements of `beta(i)` are less than `converge`, we take `beta(i)` as the unbiased estimate discussed in Chapter 5. Finally, in Line 80, the standard error of regression is computed based on the original data and the unbiased estimate, which is denoted by `se`.

Thus, `unbiased(z,x,l,n,k,lag,olse,beta,se,idist)` yields both OLSE `olse(k+lag)` and the unbiased estimator `beta(k+lag)`, given explanatory and unexplanatory data `z(m)` and `x(m,k+lag)`. In order to obtain the confidence interval discussed in Section 5.3.2, `beta(k+lag)` has to be computed for all `number` OLSEs. In the first step, store `number` OLSEs into a text file, which are obtained in the final iteration. In the next step, for all the stored OLSEs, `beta(k+lag)` is computed. Thus, we have `number` unbiased estimates.

References

Abadir, K.M., 1993, "OLS Bias in A Nonstationary Autoregression," *Econometric Theory*, Vol.9, No.1, pp.81 – 93.

Ali, M.M., 1996, "Distribution of the Least Squares Estimator in a First-Order Autoregressive Model," Unpublished Manuscript (Economics Working Paper Archive, ftp:// econwpa.wustl.edu/ econ-wp/ em/ papers/ 9610/ 9610004.ps).

Andrews, D.W.K., 1993, "Exactly Median-Unbiased Estimation of First Order Autoregressive / Unit Root Models," *Econometrica*, Vol.61, No.1, pp.139 – 165.

Andrews, D.W.K. and Chen, H.Y., 1994, "Approximately Median-Unbiased Estimation of Autoregressive Models," *Journal of Business and Economic Statistics*, Vol.12, No.2, pp.187 – 204.

Diebold, F.X. and Rudebusch, G.D., 1991, "Unbiased Estimation of Autoregressive Models," Unpublished Manuscript, University of Pennsylvania.

Efron, B. and Tibshirani, R.J., 1993, *An Introduction to the Bootstrap* (Monographs on Statistics and Applied Probability 57), Chapman & Hall.

Grubb, D. and Symons, J., 1987, "Bias in Regressions with a Lagged Dependent Variable," *Econometric Theory*, Vol.3, pp.371 – 386.

Hurwicz, L., 1950, "Least-Squares Bias in Time Series," in *Statistical Inference in Dynamic Economic Models*, edited by T.C. Koopmans, pp.365 – 383, John Wiley & Sons.

Imhof, J.P., 1961, "Computing the Distribution of Quadratic Forms in Normal Variates," *Biometrika*, Vol.48, pp.419 – 426.

Kendall, M.G., 1954, "Note on Bias in the Estimation of Autocorrelations," *Biometrika*, Vol.41, pp.403 – 404.

Maekawa, K., 1983, "An Approximation to the Distribution of the Least Squares Estimator in an Autoregressive Model with Exogenous Variables," *Econometrica*, Vol.51, No.1, pp.229 – 238.

Marriott, F.H.C. and Pope, J.A., 1954, "Bias in the Estimation of Autocorrelations," *Biometrika*, Vol.41, pp.390 – 402.

Orcutt, G.H. and Winokur, H.S., 1969, "First Order Autoregression: Inference, Estimation, and Prediction," *Econometrica*, Vol.37, No.1, pp.1 – 14.

Quenouille, M.H., 1956, "Notes on Bias in Estimation," *Biometrika*, Vol.43, pp.353 – 360.

Sawa, T., 1978, "The Exact Moments of the Least Squares Estimator for the Autoregressive Model," *Journal of Econometrics*, Vol.8, pp.159 – 172.

Shaman, P. and Stine, R.A., 1988, "The Bias of Autoregressive Coefficient Estimators," *Journal of the American Statistical Association*, Vol.83, No.403, pp.842 – 848.

Tanaka, K., 1983, "Asymptotic Expansions Associated with AR(1) Model with Unknown Mean," *Econometrica*, Vol.51, No.4, pp.1221 – 1231.

Tanizaki, H., 2000, "Bias Correction of OLSE in the Regression Model with Lagged Dependent Variables," *Computational Statistics and Data Analysis*, Vol.34, No.4, pp.495 – 511.

Tanizaki, H., 2001, "On Least-Squares Bias in the AR(p) Models: Bias Correction Using the Bootstrap Methods," Unpublished Manuscript.

Tse, Y.K., 1982, "Edgeworth Approximations in First-Order Stochastic Difference Equations with Exogenous Variables," *Journal of Econometrics*, Vol.20, pp.175 – 195.

Tsui, A.K. and Ali, M.M., 1994, "Exact Distributions, Density Functions and Moments of the Least Squares Estimator in a First-Order Autoregressive Model," *Computational Statistics & Data Analysis*, Vol.17, No.4, pp.433 – 454.

White, J.S., 1961, "Asymptotic Expansions for the Mean and Variance of the Serial Correlation Coefficient," *Biometrika*, Vol.48, pp.85 – 94.

Wu, C.F.J., 1986, "Jackknife, Bootstrap and Other Resampling Methods in Regression Analysis," *Annals of Statistics*, Vol.14, pp.1261 – 1350 (with discussion).

Chapter 6

State Space Modeling

This chapter is based on Geweke and Tanizaki (1999, 2001) and Tanizaki (1996, 2001a, 2003). In this chapter, we discuss nonlinear non-Gaussian state space modeling, where the sampling techniques such as rejection sampling (RS), importance resampling (IR) and the Metropolis-Hastings sampling (MH) are utilized to obtain the estimate of state mean. We deal with the density-based filtering and smoothing procedures, where we have two algorithms, i.e., one is recursive and another is non-recursive. Generating random draws, both density-based recursive and non-recursive algorithms on filtering and smoothing can be derived.

6.1 Introduction

Various nonlinear non-Gaussian filters and smoothers have been proposed for the last decade in order to improve precision of the state estimates and reduce a computational burden. The state mean and variance are evaluated by generating random draws directly from the filtering density or the smoothing density. Clearly, precision of the state estimates is improved as number of random draws increases. Therefore, the recent filters and smoothers have been developed by applying some sampling techniques such as Gibbs sampling, rejection sampling (RS), importance resampling (IR), the Metropolis-Hastings sampling (MH) and etc.

Carlin, Polson and Stoffer (1992) and Carter and Kohn (1994, 1996) applied the Gibbs sampler to some specific state space models, which are extended to more general state space models by Geweke and Tanizaki (1999, 2001). The Gibbs sampler sometimes gives us the imprecise estimates of the state variables, depending on the underlying state space model (see Carter and Kohn (1994, 1996) and Tanizaki (2003)). Especially when the random variables are highly correlated with each other, it is well known that convergence of the Gibbs sampler is sometimes unacceptably slow. In the case of state space models, the transition equation represents the relationship between the state variable α_t and the lagged one α_{t-1}. Therefore, it is clear that the state variable at present time (i.e., α_t) has high correlation with that at past time (i.e., α_{t-1}). As

for the state space models, thus, the Gibbs sampler is one of the sources of imprecise state estimates. In this chapter, as well as the density-based non-recursive algorithm suggested by Carlin, Polson and Stoffer (1992) and Carter and Kohn (1994, 1996), the density-based recursive algorithms on filtering and smoothing are also discussed, where their algorithms are compared with respect to the three sampling methods, i.e., RS, IR and MH, although any sampling technique can be applied.

Gordon, Salmond and Smith (1993), Kitagawa (1996, 1998) and Kitagawa and Gersch (1996) proposed the nonlinear non-Gaussian state space modeling by IR, which is related to the density-based recursive algorithms and can be applied to almost all the state space models. In the density-based recursive algorithms suggested by Gordon, Salmond and Smith (1993), Kitagawa (1996, 1998) and Kitagawa and Gersch (1996), both filtering and smoothing random draws are based on the one-step ahead prediction random draws. In the case where the past information is too different from the present sample, however, the obtained filtering and smoothing random draws become unrealistic. To avoid this situation, Tanizaki (2001a) suggested taking a more plausible density as the sampling density for random number generation. Note that Kong, Liu and Chen (1994), Liu and Chen (1995, 1998) and Doucet, Godsill and Andrieu (2000) utilized the sampling density other than the one-step ahead prediction density. In addition, the fixed-interval smoother proposed by Kitagawa (1996) and Kitagawa and Gersch (1996) does not give us the exact solution of the state estimate even when the number of random draws is large enough, because the fixed-interval smoother suggested by Kitagawa (1996) is approximated by the fixed-lag smoother. To improve this disadvantage, Tanizaki (2001a) proposed the fixed-interval smoother which yields the exact solution of the state estimate. As an alternative smoother, furthermore, Kitagawa (1996) introduces the fixed-interval smoother based on the two-filter formula, where forward and backward filtering are performed and combined to obtain the smoothing density. The smoother based on the two-filter formula is concisely discussed in Appendix 6.4.

The RS filter and smoother which are the density-based recursive algorithms have been developed by Tanizaki (1996, 1999), Hürzeler and Künsch (1998) and Tanizaki and Mariano (1998). To implement RS for random number generation, we need to compute the supremum included in the acceptance probability, which depends on the underlying functional form of the measurement and transition equations. RS cannot be applied in the case where the acceptance probability is equal to zero, i.e., when the supremum is infinity. Even if the supremum is finite, it takes a lot of computational time when the acceptance probability is close to zero. To improve the problems in RS, Liu, Chen and Wong (1998) suggested combining rejection sampling and importance sampling.

Tanizaki (2001a) have suggested generating random draws from the joint densities of the state variables, i.e., $f(\alpha_t, \alpha_{t-1}|Y_t)$ for filtering and $f(\alpha_{t+1}, \alpha_t|Y_n)$ or $f(\alpha_{t+1}, \alpha_t, \alpha_{t-1}|Y_n)$ for smoothing, where the notations are defined later in Section 6.2.1. Kong, Liu and Chen (1994) and Liu and Chen (1995) also suggested drawing from the joint density of the state variable and the auxiliary variable, where they discuss filtering but

not smoothing. In a lot of literature, smoothing is not investigated because smoothing is much more computer-intensive than filtering. Dealing with the joint densities of the state variables yields less computational smoothers. Along Tanizaki (2001a), we introduce the density-based recursive nonlinear non-Gaussian filters and smoothers using the joint densities of the state variables.

The outline of this chapter is as follows. In Section 6.2, the definition of the state space models is given and some applications are introduced. In Section 6.3, the density-based recursive algorithms on filtering and smoothing are discussed, while in Section 6.4 the density-based non-recursive procedure on smoothing is introduced. In Section 6.5, we perform some Monte Carlo studies to compare the algorithms discussed in Sections 6.3 and 6.4. An empirical example, in Section 6.6 the percent change of Nikkei stock average is estimated as a sum of the time-dependent parameter and the stochastic variance, where the daily data from January 4, 1991 to January 31, 2003 are utilized.

6.2 State Space Models

6.2.1 Definition

Kitagawa (1987), Harvey (1989), Kitagawa and Gersch (1996) and Tanizaki (1996, 2001a, 2003) discuss the nonlinear non-Gaussian state space models, which are described by the following two equations:

$$\text{(Measurement equation)} \qquad y_t = h_t(\alpha_t, \epsilon_t), \tag{6.1}$$

$$\text{(Transition equation)} \qquad \alpha_t = p_t(\alpha_{t-1}, \eta_t), \tag{6.2}$$

for $t = 1, 2, \cdots, n$, where y_t represents the observed data at time t while α_t denotes a vector of **state variable** at time t which is unobservable. The error terms ϵ_t and η_t are assumed to be mutually independently distributed. $h_t(\cdot,\cdot)$ and $p_t(\cdot,\cdot)$ are known, i.e., $h_t(\cdot,\cdot)$ and $p_t(\cdot,\cdot)$ have to be specified by a researcher. Consider three mathematical expectations on the state variable α_t. Let Y_s be the information set up to time s, i.e., $Y_s = \{y_1, y_2, \cdots, y_s\}$. Then, $\alpha_{t|s} \equiv \mathrm{E}(\alpha_t|Y_s)$ is called **prediction** if $t > s$, **filtering** if $t = s$ and **smoothing** if $t < s$. Moreover, there are three kinds of smoothing, i.e., the **fixed-point smoothing** $\alpha_{L|t}$, the **fixed-lag smoothing** $\alpha_{t|t+L}$ and the **fixed-interval smoothing** $\alpha_{t|n}$ for fixed L and fixed n. In this chapter, we focus on the filter and the fixed-interval smoother, i.e., $\alpha_{t|s}$ for $s = t, n$. Thus, the purpose of state space modeling is to estimate the unobservable variable α_t using the observed data y_t.

In the state space model above, we have several applications, which are shown in the next section.

6.2.2 Applications

Some applications of the state space model, especially in a field economics, are discussed in this section, where we consider the following examples: Time Varying Parameter Model, Autoregressive-Moving Average Process, Seasonal Adjustment Models, Prediction of Final Data Using Preliminary Data, Estimation of Permanent Consumption, Markov Switching Model and Stochastic Variance Models.

Time Varying Parameter Model

We consider the case where we deal with time series data. Suppose that the nonlinear regression model can be written as follows:

$$y_t = h_t(x_t, \alpha, \epsilon_t),$$

for $t = 1, 2, \cdots, n$, where y_t is a dependent variable, x_t denotes a $1 \times k$ vector of the explanatory variables, a $k \times 1$ vector of unknown parameters to be estimated is given by α, and ϵ_t is the error term. $h_t(\cdot, \cdot, \cdot)$ is assumed to be a known vector function, which is given by $h_t(x_t, \alpha, \epsilon_t) = x_t\alpha + \epsilon_t$ in a classical linear regression model. Given data (y_t, x_t), $t = 1, 2, \cdots, n$, we want to estimate the unknown parameter α. There are some methods to estimate the equation above, for example, the least squares method, the method of moments, the maximum likelihood estimation method and so on. When the unknown parameters are assumed to be constant over time, the regression model is known as the **fixed-parameter model**. However, structural changes (for example, the first and second oil crises), specification errors, proxy variables and aggregation are all the sources of parameter variation; see Sarris (1973), Belsley (1973), Belsley and Kuh (1973) and Cooley and Prescott (1976). Therefore, we need to consider the regression model such that the parameter is a function of time, which is called the **time varying parameter model**. Using the state space form, the model can be rewritten as the following two equations:

$$\text{(Measurement equation)} \qquad y_t = h_t(x_t, \alpha_t, \epsilon_t), \tag{6.3}$$

$$\text{(Transition equation)} \qquad \alpha_t = \Psi\alpha_{t-1} + \eta_t, \tag{6.4}$$

where the movement of the parameter α_t is assumed to be the first order autoregressive (AR(1)) process, which can be extended to the pth order autoregressive process (AR(p)) process. The error term η_t, independent of ϵ_t, is a white noise. Here, equations (6.3) and (6.4) are referred to as the measurement equation and the transition equation, respectively. The time varying parameter α_t is unobservable, which is estimated using the observed data y_t and x_t. There are numerous other papers which deal with the time varying parameter model, for example, Cooley (1977), Cooley, Rosenberg and Wall (1977), Cooper (1973), Nicholls and Pagan (1985), Tanizaki (1989, 1993a, 2000) and Sant (1977).

Autoregressive Moving Average Process

It is well known that any autoregressive-moving average (ARMA) process can be written in the state space form. See, for example, Aoki (1987, 1990), Brockwell and Davis (1987), Burridge and Wallis (1988), Gardner, Harvey and Phillips (1980), Hannan and Deistler (1988), Harvey (1981, 1989) and Kirchen (1988).

First, consider the following ARMA(p, q) process.

$$y_t = a_1 y_{t-1} + a_2 y_{t-2} + \cdots + a_p y_{t-p} + \eta_t + b_1 \eta_{t-1} + \cdots + b_q \eta_{t-q},$$

where η_t is a white noise. The ARMA(p, q) model above is rewritten as:

$$y_t = a_1 y_{t-1} + a_2 y_{t-2} + \cdots + a_m y_{t-m} + \eta_t + b_1 \eta_{t-1} + \cdots + b_{m-1} \eta_{t-m+1},$$

where $m = \max(p, q+1)$ and some of the coefficients $a_1, a_2, \cdots, a_m, b_1, b_2, \cdots, b_{m-1}$ can be zeros. As it is well known, the ARMA process above is represented as:

$$\begin{aligned} &\text{(Measurement equation)} && y_t = z\alpha_t, \\ &\text{(Transition equation)} && \alpha_t = A\alpha_{t-1} + B\eta_t, \end{aligned}$$

where z, A and B are defined as:

$$z = (1, 0, \cdots, 0), \quad A = \left(\begin{array}{c|c} a_1 & \\ \vdots & I_{m-1} \\ a_{m-1} & \\ \hline a_m & 0 \end{array}\right), \quad B = \begin{pmatrix} 1 \\ b_1 \\ \vdots \\ b_{m-1} \end{pmatrix}.$$

$$1 \times m \qquad m \times m \qquad m \times 1$$

Thus, the state space model is constructed from the ARMA model, where the first element of α_t represents the time series data to be estimated.

Seasonal Adjustment Models

A time series consists of seasonal, cyclical, and irregular components. Each component is unobservable and therefore the state space model is applied to estimate each component separately. Here, two seasonal adjustment models are introduced; one is developed by Pagan (1975) and another is Kitagawa (1996) and Kitagawa and Gersch (1996).

The suggestion by Pagan (1975) is essentially a combination of an econometric model for the cyclical components with the filtering and estimation of the seasonal components formulated in the state space form (see Chow (1983)). Assume, first, that an endogenous variable y_t is the sum of cyclical, seasonal, and irregular components, as given by:

$$y_t = y_t^c + y_t^s + \epsilon_t, \tag{6.5}$$

where y_t^c, y_t^s and ϵ_t denote the cyclical, seasonal, and irregular components, respectively. Second, the cyclical component y_t^c is represented as the following model:

$$y_t^c = Ay_{t-1}^c + Cx_t + u_t, \tag{6.6}$$

where x_t is a $k \times 1$ vector of exogenous variables and u_t denotes a random disturbance. In equation (6.6), the AR(1) model is assumed for simplicity but the AR(p) model is also possible. Finally, an autoregressive seasonal model is assumed for the seasonal component, i.e.,

$$y_t^s = By_{t-m}^s + w_t, \tag{6.7}$$

where w_t represents a random disturbance and m can be 4 for quarterly data and 12 for monthly data. Combining the equations (6.5) – (6.7), we can construct the following state space form:

$$\begin{array}{ll} \text{(Measurement equation)} & y_t = z\alpha_t + \epsilon_t, \\ \text{(Transition equation)} & \alpha_t = M\alpha_{t-1} + Nx_t + \eta_t, \end{array} \tag{6.8}$$

where z, α_t, M, N and η_t are given by:

$$z = (1, 1, 0, \cdots, 0), \quad M = \left(\begin{array}{c|cc} A & 0 & \\ \hline & 0 & I_{m-1} \\ 0 & B & 0 \end{array}\right), \quad N = \begin{pmatrix} C \\ 0 \end{pmatrix}, \quad \eta_t = \begin{pmatrix} u_t \\ w_t \\ 0 \end{pmatrix}.$$

$$1 \times (m+1) \qquad (m+1) \times (m+1) \qquad (m+1) \times k \qquad (m+1) \times 1$$

The first and second elements of α_t represent y_t^c and y_t^s, respectively.

Kitagawa (1996) and Kitagawa and Gersch (1996) suggested an alternative seasonal component model, which is represented by equation (6.5) and the following two equations:

$$y_t^c = a_1 y_{t-1}^c + a_2 y_{t-2}^c + \cdots + a_p y_{t-p}^c + u_t, \tag{6.9}$$

$$y_t^s = -y_{t-1}^s - y_{t-2}^s - \cdots - y_{t-m+1}^s + w_t, \tag{6.10}$$

where equation (6.9) may depend on the other exogenous variables x_t as in equation (6.6). In equation (6.10), B in equation (6.7) is assumed to be $B = 1$ and the nonstationary component is removed from y_t^s. That is, assuming $B = 1$ in equation (6.7), we have $y_t^s - y_{t-m}^s = (1 - L^m)y_t = (1 - L)(1 + L + L^2 + \cdots + L^{m-1})y_t^s = w_t$. Equation (6.10) corresponds to $(1 + L + L^2 + \cdots + L^{m-1})y_t^s = w_t$ and the nonstationary component is included in equation (6.9). Thus, equations (6.5), (6.9) and (6.10) yield the following state space model:

$$\begin{array}{ll} \text{(Measurement equation)} & y_t = z\alpha_t + \epsilon_t, \\ \text{(Transition equation)} & \alpha_t = M\alpha_{t-1} + \eta_t, \end{array} \tag{6.11}$$

where z, α_t, M and η_t are given by:

$$z = (1, 0, \cdots, 0, 1, 0, \cdots, 0), \quad A = \left(\begin{array}{c|c} a_1 & \\ \vdots & I_{p-1} \\ a_{p-1} & \\ \hline a_p & 0 \end{array}\right), \quad B = \left(\begin{array}{c|c} -1 & \\ \vdots & I_{m-2} \\ -1 & \\ \hline -1 & 0 \end{array}\right),$$

$$1 \times (p+m-1) \qquad p \times p \qquad (m-1) \times (m-1)$$

$$M = \begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix}, \qquad \eta_t = (u_t, 0, \cdots, 0, w_t, 0, \cdots, 0)'.$$

$$(p+m-1) \times (p+m-1) \qquad (p+m-1) \times 1$$

All the elements of z and η_t are zeros except for the first and $(p+1)$th elements. The cyclical component y_t^c and the seasonal component y_t^s are given by the first element of α_t and the $(p+1)$th element of α_t, respectively. Note that difference between the two systems (6.8) and (6.11) is a formulation in the seasonal component, which is described in equations (6.7) and (6.10).

Prediction of Final Data Using Preliminary Data

It is well known that economic indicators such as Gross Domestic Product (GDP) are usually reported according to the following two steps: (i) the **preliminary data** are reported and (ii) thereafter we can obtain the final or revised data (see Table 6.1). The problem is how to estimate the **final data** (or the **revised data**) when only the preliminary data are available.

In the case of annual data on the U.S. national accounts, the preliminary data at the present time are reported at the beginning of the next year. The revision process is performed over a few years and every decade, as shown in Table 6.1, where an example of the nominal gross domestic product data (GDP, billions of dollars) is taken.

In Table 6.1, the preliminary data of 1988, 1992, 1995, 1996, 1999, 2001 and 2002 are taken from *Survey of Current Business* (January in 1989, January in 1993, July in 1996, February in 1997, February in 1999, February in 2001 and February in 2002), while the rest of the preliminary data and all the revised data are from *Economic Report of the President* (ERP), published from 1982 to 2002. Each column indicates the year when ERP is published, while each row represents the GDP data of the corresponding year. The superscripts p and r denote the preliminary data and the data revised in the year corresponding to each column. NA indicates that the data are not available, which implies that the data have not been published yet. For instance, take the GDP data of 1984 (see the row corresponding to 1984). The preliminary GDP data of 1984 was reported in 1985 (i.e., 3616.3), and it was revised in 1986 for the first time (i,e., 3726.7). In 1987 and 1988, the second and third revised data were published, respectively (i.e., 3717.5 and 3724.8). Since it was not revised in 1989, the

Table 6.1: Revision Process of U.S. National Accounts (Nominal GDP)

	1982	1983	1984	1985	1986	1987	1988
1979	2370.1^r	2375.2^r	2375.2	2375.2	2464.4^r	2464.4	2464.4
1980	2576.5^r	2587.0^r	2586.4^r	2586.4	2684.4^r	2684.4	2684.4
1981	2868.2^p	2888.5^r	2904.5^r	2907.5^r	3000.5^r	3000.5	3000.5
1982	NA	3011.9^p	3025.7^r	3021.3^r	3114.8^r	3114.8	3114.8
1983	NA	NA	3263.4^p	3256.5^r	3350.9^r	3355.9^r	3355.9
1984	NA	NA	NA	3616.3^p	3726.7^r	3717.5^r	3724.8^r
1985	NA	NA	NA	NA	3951.8^p	3957.0^r	3970.5^r
1986	NA	NA	NA	NA	NA	4171.2^p	4201.3^r
1987	NA	NA	NA	NA	NA	NA	4460.2^p
	1989	1990	1991	1992	1993	1994	1995
1979	2464.4	2464.4	2464.4	2488.6^r	2488.6	2488.6	2488.6
1980	2684.4	2684.4	2684.4	2708.0^r	2708.0	2708.0	2708.0
1981	3000.5	3000.5	3000.5	3030.6^r	3030.6	3030.6	3030.6
1982	3114.8	3114.8	3114.8	3149.6^r	3149.6	3149.6	3149.6
1983	3355.9	3355.9	3355.9	3405.0^r	3405.0	3405.0	3405.0
1984	3724.8	3724.8	3724.8	3777.2^r	3777.2	3777.2	3777.2
1985	3974.1^r	3974.1	3974.1	4038.7^r	4038.7	4038.7	4038.7
1986	4205.4^r	4197.2^r	4197.2	4268.6^r	4268.6	4268.6	4268.6
1987	4497.2^r	4493.8^r	4486.7^r	4539.9^r	4539.9	4539.9	4539.9
1988	4837.8^p	4847.3^r	4840.2^r	4900.4^r	4900.4	4900.4	4900.4
1989	NA	5199.6^p	5163.2^r	5244.0^r	5250.8^r	5250.8	5250.8
1990	NA	NA	5424.4^p	5513.8^r	5522.2^r	5546.1^r	5546.1
1991	NA	NA	NA	5671.8^p	5677.5^r	5722.8^r	5724.8^r
1992	NA	NA	NA	NA	5945.7^p	6038.5^r	6020.2^r
1993	NA	NA	NA	NA	NA	6374.0^p	6343.3^r
1994	NA	NA	NA	NA	NA	NA	6736.9^p
	1996	1997	1998	1999	2000	2001	2002
1979	2557.5^r	2557.5	2557.5	2557.5	2566.4^r	2566.4	2566.4
1980	2784.2^r	2784.2	2784.2	2784.2	2795.6^r	2795.6	2795.6
1981	3115.9^r	3115.9	3115.9	3115.9	3131.3^r	3131.3	3131.3
1982	3242.1^r	3242.1	3242.1	3242.1	3259.2^r	3259.2	3259.2
1983	3514.5^r	3514.5	3514.5	3514.5	3534.9^r	3534.9	3534.9
1984	3902.4^r	3902.4	3902.4	3902.4	3932.7^r	3932.7	3932.7
1985	4180.7^r	4180.7	4180.7	4180.7	4213.0^r	4213.0	4213.0
1986	4422.2^r	4422.2	4422.2	4422.2	4452.9^r	4452.9	4452.9
1987	4692.3^r	4692.3	4692.3	4692.3	4742.5^r	4742.5	4742.5
1988	5049.6^r	5049.6	5049.6	5049.6	5108.3^r	5108.3	5108.3
1989	5438.7^r	5438.7	5438.7	5438.7	5489.1^r	5489.1	5489.1
1990	5743.8^r	5743.8	5743.8	5743.8	5803.2^r	5803.2	5803.2
1991	5916.7^r	5916.7	5916.7	5916.7	5986.2^r	5986.2	5986.2
1992	6244.4^r	6244.4	6244.4	6244.4	6318.9^r	6318.9	6318.9
1993	6550.2^r	6553.0^r	6558.1^r	6558.1	6642.3^r	6642.3	6642.3
1994	6931.4^r	6935.7^r	6947.0^r	6947.0	7054.3^r	7054.3	7054.3
1995	7245.8^p	7253.8^r	7265.4^r	7269.6^r	7400.5^r	7400.5	7400.5
1996	NA	7580.0^p	7636.0^r	7661.6^r	7813.2^r	7813.2	7813.2
1997	NA	NA	8083.4^p	8110.9^r	8300.8^r	8318.4^r	8318.4
1998	NA	NA	NA	8508.9^p	8759.9^r	8790.2^r	8781.5^r
1999	NA	NA	NA	NA	9248.4^p	9299.2^r	9268.6^r
2000	NA	NA	NA	NA	NA	9965.7^p	9872.9^r
2001	NA	NA	NA	NA	NA	NA	10197.7^p

GDP data of 1984 published in 1989 is given by 3724.8. Moreover, the GDP data of 1984 was revised as 3777.2 in 1992, 3902.4 in 1996 and 3932.7 in 2000.

Thus, the data series is revised every year for the first few years and thereafter less frequently. This implies that we cannot really know the true final data, because the data are revised forever while the preliminary data are reported only once. Therefore, it might be possible to consider that the final data are unobservable, which leads to estimation of the final data given the preliminary data.

There is a wide literature dealing with the data revision process. Conrad and Corrado (1979) applied the Kalman filter to improve upon published preliminary estimates of monthly retail sales, using an ARIMA model. Howrey (1978, 1984) used the preliminary data in econometric forecasting and obtained the substantial improvements in forecast accuracy if the preliminary and revised data are used optimally.

In the context of the revision process, the filtering and smoothing techniques are used as follows. There is some relationship between the final and preliminary data, because they are originally same data (see, for example, Conrad and Corrado (1979)). This relationship is referred to as the measurement equation, where the final data is unobservable but the preliminary data is observed. The equation obtained by the underlying economic theory is related to the final data, rather than the preliminary data. This equation is taken as the transition equation. Therefore, we can represent the revision problem with the following state space form:

$$\begin{aligned} &\text{(Measurement equation)} && y_t^p = h_t(y_t^f, \epsilon_t), \\ &\text{(Transition equation)} && y_t^f = p_t(y_{t-1}^f, x_t, \eta_t), \end{aligned}$$

where y_t^p and y_t^f denote the preliminary data and the final data, respectively. The unobserved state variable is given by y_t^f, while y_t^p is observable. Thus, the state space model is utilized to estimate y_t^f (see Mariano and Tanizaki (1995) and Tanizaki and Mariano (1994)).

Estimation of Permanent Consumption

The next economic application is concerned with estimation of permanent consumption. Total consumption consists of permanent and transitory consumption. This relationship is represented by an identity equation, which corresponds to the measurement equation. Permanent consumption depends on life-time income expected in the future, i.e., permanent income. The following expected utility function of the representative agent is maximized with respect to permanent consumption (see Hall (1978, 1990)):

$$\max_{\{c_t^p\}} \quad \mathrm{E}_0\left(\sum_t \beta^t u(c_t^p)\right), \text{ subject to } A_{t+1} = R_t(A_t + y_t - c_t),$$

where $0 < \beta < 1$ and $c_t = c_t^p + c_t^T$. c_t, c_t^p, c_t^T, R_t, A_t, y_t, β, $u(\cdot)$ and $\mathrm{E}_t(\cdot)$ denote per capita total consumption, per capita **permanent consumption**, per capita **transitory**

consumption, the real gross rate of return on savings between periods t and $t+1$, the stock of assets at the beginning of period t, per capita labor income, the discount rate, the representative utility function and the mathematical expectation given information up to t, respectively.

Solving the above maximization problem, we can obtain the transition equation which represents the relationship between c_t^p and c_{t-1}^p. Transitory consumption is assumed to be a random shock with mean zero and variance σ_ϵ.

Under the above setup, the model to this problem is given by:

$$\text{(Measurement equation)} \qquad c_t = c_t^p + \epsilon_t,$$

$$\text{(Transition equation)} \qquad \frac{\beta R_{t-1} u'(c_t^p)}{u'(c_{t-1}^p)} = 1 + \eta_t,$$

Note that $c_t^T = \epsilon_t$, which is assumed to be independent of η_t. c_t is observable while both c_t^p and c_t^T are unobservable, where c_t^p is regarded as the state variable to be estimated by the nonlinear filtering and smoothing technique. Thus, we can estimate permanent and transitory consumption separately. Tanizaki (1993b, 1996) and Mariano and Tanizaki (2000) consider the above example, where the utility function of the representative agent is assumed to be a constant relative risk aversion type of utility function. Also see Diebold and Nerlove (1989) for a concise survey of testing the permanent income hypothesis.

Markov Switching Model

The Markov switching model was developed by Hamilton (1989, 1990, 1991, 1993, 1994), where the discrete random variable is taken for the state variable. Consider the k-dimensional discrete state variable, i.e., $\alpha_t = (\alpha_{1t}, \alpha_{2t}, \cdots, \alpha_{kt})'$, where we assume that one of the k elements of α_t is one and the others are zeros.

First, consider the following model:

$$y_t = h_t(\mu_t^*, \epsilon_t), \tag{6.12}$$

where μ_t^* is a discrete random variable and $h_t(\cdot, \cdot)$ may depend on the other exogenous variable x_t. Assume that μ_t^* depends on the unobserved random variable s_t^*, which is called the state or regime. Suppose that we have k states (or regimes). If $s_t^* = j$, then the process is in regime j and $\mu_t^* = \mu_j$ is taken. We assume that one of the k states at time t occurs depending on time $t-1$.

Define $p = (p_1', p_2', \cdots, p_k')'$ as the transition probability matrix, where $p_i = (p_{i1}, p_{i2}, \cdots, p_{ik})$ for $i = 1, 2, \cdots, k$. Note that $\sum_{i=1}^k p_{ij} = 1$ should be satisfied for all $j = 1, 2, \cdots, k$. p_{ij} implies the conditional probability of $s_t^* = j$ given $s_{t-1}^* = i$, i.e., $p_{ij} \equiv P(s_t^* = j | s_{t-1}^* = i)$. Such a process is described as an k-state Markov chain with transition probabilities $\{p_{ij}\}_{i,j=1,2,\cdots,k}$. The transition probability p_{ij} gives the probability that state i is followed by state j. Under the above setup, each element

of the $k \times 1$ multivariate discrete state variable α_t takes a binary number, i.e.,

$$\alpha_t = \begin{cases} (1,0,0,\cdots,0)', & \text{when } s_t^* = 1, \\ (0,1,0,\cdots,0)', & \text{when } s_t^* = 2, \\ \vdots & \vdots \\ (0,0,0,\cdots,1)', & \text{when } s_t^* = k. \end{cases}$$

Let us define $\mu = (\mu_1, \mu_2, \cdots, \mu_k)$, where each element depends on the regime. Then, μ_t^* is rewritten as: $\mu_t^* = \mu\alpha_t$. Accordingly, the model described in equation (6.12) is represented by the following state space model:

$$\begin{aligned} &\text{(Measurement equation)} \quad && y_t = h_t(\mu\alpha_t, \epsilon_t), \\ &\text{(Transition equation)} && \alpha_t = p\alpha_{t-1} + \eta_t, \end{aligned}$$

for $t = 1, 2, \cdots, n$. μ is a $1 \times k$ vector of unknown parameter to be estimated. η_t is distributed as a k-dimensional discrete random variable. The conditional density of α_t given α_{t-1} is represented by $f_\alpha(\alpha_t|\alpha_{t-1}) = \prod_{i=1}^{k}(p_i\alpha_{t-1})^{\alpha_{it}}$, which implies that the probability which event i occurs at time t is $p_i\alpha_{t-1}$.

Thus, it is assumed in the Markov switching model that the economic situation is stochastically switched from one to another for each time. The Markov switching model is similar to the time varying parameter model introduced above in the sense that the parameter changes over time. From specification of the transition equation, however, the time varying parameter model takes into account a gradual shift in the economic structural change but the Markov switching model deals with a sudden shift because μ_t^* is a discrete random variable which depends on state or regime.

Stochastic Variance Models

In this section, we introduce two stochastic variance models (see Taylor (1994) for the stochastic variance models). One is called the **autoregressive conditional heteroscedasticity (ARCH) model** proposed by Engle (1982) and another is the **stochastic volatility model** (see Ghysels, Harvey, and Renault (1996)). Both models are often applied to financial data.

Let β be a $k \times 1$ vector of unknown parameters to be estimated. y_t and x_t are assumed to be observable variables. The first order **ARCH model** is given by the following two equations:

$$\begin{aligned} &\text{(Measurement equation)} \quad && y_t = x_t\beta + \alpha_t, \\ &\text{(Transition equation)} && \alpha_t = (\delta_0 + \delta_1\alpha_{t-1}^2)^{1/2}\eta_t, \end{aligned}$$

for $t = 1, 2, \cdots, n$, where $\eta_t \sim N(0,1)$, $0 < \delta_0$ and $0 \leq \delta_1 < 1$ have to be satisfied. β, δ_0 and δ_1 are unknown parameters to be estimated. The conditional variance of α_t is represented by $\delta_0 + \delta_1\alpha_{t-1}^2$ while the unconditional variance is given by $\delta_0/(1-\delta_1)$. It might be possible to put the extra error term (say, ϵ_t) in the measurement equation,

i,e., $y_t = x_t\beta + \alpha_t + \epsilon_t$. Using the generalized ARCH (so-called GARCH) model, Watanabe (2000b) examined what distribution is fit for the conditional distribution of daily Japanese stock returns.

As an alternative stochastic variance model, we consider the **stochastic volatility model**, which is defined as follows:

$$\begin{aligned} &\text{(Measurement equation)} \qquad y_t = x_t\beta + \exp\Big(\frac{1}{2}\alpha_t\Big)\epsilon_t, \\ &\text{(Transition equation)} \qquad \alpha_t = \delta\alpha_{t-1} + \eta_t, \end{aligned}$$

for $t = 1, 2, \cdots, n$, where $0 \leq \delta < 1$ has to be satisfied. The error terms ϵ_t and η_t are mutually independently distributed. Watanabe (1999, 2000a) presented a new technique based on a nonlinear filter for the analysis of stochastic volatility models

For the other applications of the state space model in economics, we can find estimation of the rational expectation models (for example, see Burmeister and Wall (1982), Engle and Watson (1987) and McNelis and Neftci (1983)). See Harvey (1987) for a survey of applications of the Kalman filter model.

6.3 Recursive Algorithm

As a solution to estimation of the state variable, we have two kind of estimation methods; one is based on the recursive algorithm and another is the non-recursive algorithm. In the former the random draws of the state variable at time t (i.e., α_t) are recursively generated and in the latter the random draws of all the state variables (i.e., α_1, α_2, $\cdots$, α_n) are simultaneously generated. The former utilizes the sampling techniques such as rejection sampling (Section 3.2), importance resampling (Section 3.3) and the MH algorithm (Section 3.4), while the latter utilizes the Markov chain Monte Carlo (MCMC) methods such as the Gibbs sampler (Section 3.6) and the MH algorithm (they are often used together with rejection sampling and importance resampling). In this section the recursive algorithm is discussed and in the next section the non-recursive algorithm is examined.

Define $f_y(y_t|\alpha_t)$ and $f_\alpha(\alpha_t|\alpha_{t-1})$ by the density functions derived from equations (6.1) and (6.2). The density-based recursive algorithm on filtering is given by:

(Prediction equation)

$$f(\alpha_t|Y_{t-1}) = \int f_\alpha(\alpha_t|\alpha_{t-1}) f(\alpha_{t-1}|Y_{t-1})\, \mathrm{d}\alpha_{t-1}, \tag{6.13}$$

(Updating equation)

$$f(\alpha_t|Y_t) = \frac{f_y(y_t|\alpha_t) f(\alpha_t|Y_{t-1})}{\int f_y(y_t|\alpha_t) f(\alpha_t|Y_{t-1})\, \mathrm{d}\alpha_t}, \tag{6.14}$$

for $t = 1, 2, \cdots, n$. The derivation of equations (6.13) and (6.14) is discussed in Appendix 6.1.

The initial condition is given by: $f(\alpha_1|Y_0) = \int f_\alpha(\alpha_1|\alpha_0) f_\alpha(\alpha_0)\, \mathrm{d}\alpha_0$ if α_0 is stochastic and $f(\alpha_1|Y_0) = f_\alpha(\alpha_1|\alpha_0)$ otherwise, where $f_\alpha(\alpha_0)$ denotes the unconditional density of α_0. The filtering algorithm takes the following two steps: (i) from equation (6.13), $f(\alpha_t|Y_{t-1})$ is obtained given $f(\alpha_{t-1}|Y_{t-1})$, which equation is called **prediction equation**, and (ii) from equation (6.14), $f(\alpha_t|Y_t)$ is derived given $f(\alpha_t|Y_{t-1})$, which equation is called **updating equation**, In equation (6.13), the density of α_t is obtained given Y_{t-1}. In equation (6.14), the past information Y_{t-1} is combined with the present sample y_t and the density of α_t is updated. Thus, $f(\alpha_t|Y_t)$ is recursively obtained for $t = 1, 2, \cdots, n$. $f(\alpha_t|Y_t)$ is called the filtering density and $f(\alpha_t|Y_{t-1})$ is known as the one-step ahead prediction density or simply the prediction density.

The density-based recursive algorithm on smoothing utilizes both the one-step ahead prediction density $f(\alpha_{t+1}|Y_t)$ and the filtering density $f(\alpha_t|Y_t)$, which is represented by:

$$f(\alpha_t|Y_n) = f(\alpha_t|Y_t) \int \frac{f(\alpha_{t+1}|Y_n) f_\alpha(\alpha_{t+1}|\alpha_t)}{f(\alpha_{t+1}|Y_t)}\, \mathrm{d}\alpha_{t+1}, \tag{6.15}$$

for $t = n-1, n-2, \cdots, 1$. See Appendix 6.1 for derivation of the smoothing algorithm shown above. Given filtering density $f(\alpha_t|Y_t)$ and one-step ahead prediction density $f(\alpha_{t+1}|Y_t)$, the smoothing algorithm represented by equation (6.15) is a backward recursion from $f(\alpha_{t+1}|Y_n)$ to $f(\alpha_t|Y_n)$. $f(\alpha_t|Y_n)$ is called the fixed-lag smoothing density or simply the smoothing density.

Let $g(\cdot)$ be a function, e.g., $g(\alpha_t) = \alpha_t$ for mean or $g(\alpha_t) = (\alpha_t - \alpha_{t|s})(\alpha_t - \alpha_{t|s})'$ for variance, where $\alpha_{t|s} \equiv \mathrm{E}(\alpha_t|Y_s)$ is defined in Section 6.2.1. Given $f(\alpha_t|Y_s)$, the conditional expectation of $g(\alpha_t)$ given Y_s is represented by:

$$\mathrm{E}(g(\alpha_t)|Y_s) = \int g(\alpha_t) f(\alpha_t|Y_s)\, \mathrm{d}\alpha_t. \tag{6.16}$$

When we have the unknown parameters in equations (6.1) and (6.2), the following likelihood function is maximized with respect to the parameters:

$$f(Y_n) = \prod_{t=1}^{n} f(y_t|Y_{t-1}) = \prod_{t=1}^{n} \left(\int f_y(y_t|\alpha_t) f(\alpha_t|Y_{t-1})\, \mathrm{d}\alpha_t \right). \tag{6.17}$$

Since $f(y_t|Y_{t-1})$ in equation (6.17) corresponds to the denominator in equation (6.14), we do not need extra computation for evaluation of equation (6.17). Thus, the unknown parameter is obtained by maximum likelihood estimation (MLE).

Our goal is to estimate the expectation in equation (6.16), which is evaluated generating random draws of α_t. The sampling techniques have been discussed in Chapter 3. Again, we overview some sampling techniques as follows. We want to generate random draws from $f(x)$, called the target density, but it is hard to sample from $f(x)$. Suppose that it is easy to generate a random draw from another density $f_*(x)$, called the sampling density. In this case, random draws of x from $f(x)$ are generated by

utilizing the random draws sampled from $f_*(x)$. Let x_i be the ith random draw of x generated from $f(x)$. Suppose that $q(x)$ is proportional to the ratio of the target density and the sampling density, i.e., $q(x) \propto f(x)/f_*(x)$. Then, the target density is rewritten as: $f(x) \propto q(x)f_*(x)$. Based on $q(x)$, the acceptance probability is obtained. Depending on the structure of the acceptance probability, we have three kinds of sampling techniques, i.e., RS, IR and MH. See Liu (1996) and Section 3.7.4 for comparison of the three sampling methods. Thus, to generate random draws of x from $f(x)$, the functional form of $q(x)$ should be known and random draws have to be easily generated from $f_*(x)$.

We apply the sampling techniques to estimation of the state variable. Let $\alpha_{i,t|s}$ be the ith random draw of α_t from $f(\alpha_t|Y_s)$. Using the sampling techniques such as RS, IR and MH, we consider generating $\alpha_{i,t|s}$ (we discuss this later in this chapter). If the random draws $(\alpha_{1,t|s}, \alpha_{2,t|s}, \cdots, \alpha_{N,t|s})$ for $s = t, n$ and $t = 1, 2, \cdots, n$ are available, equation (6.16) is evaluated by $\mathrm{E}(g(\alpha_t)|Y_s) \approx (1/N)\sum_{i=1}^{N} g(\alpha_{i,t|s})$. Similarly, the likelihood function (6.17) is given by:

$$f(Y_n) \approx \prod_{t=1}^{n}\left(\frac{1}{N}\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1})\right), \tag{6.18}$$

where $\alpha_{i,t|t-1} = p_t(\alpha_{i,t-1|t-1}, \eta_{i,t})$ and $\eta_{i,t}$ denotes the ith random draw of η_t.

6.3.1 Filtering

Assuming that $\alpha_{i,t-1|t-1}$ for $i = 1, 2, \cdots, N$ are available, an attempt is made to generate $\alpha_{i,t|t}$ for $i = 1, 2, \cdots, N$. Depending on whether the initial value α_0 is stochastic or not, $\alpha_{i,0|0}$ for $i = 1, 2, \cdots, N$ are assumed to be generated from $f_\alpha(\alpha_0)$ or to be fixed for all i.

We have two representations on the filtering density (6.14). First, as shown from equation (6.14), $f(\alpha_t|Y_t)$ is immediately rewritten as follows:

$$f(\alpha_t|Y_t) \propto q_1(\alpha_t)f(\alpha_t|Y_{t-1}), \tag{6.19}$$

where $q_1(\alpha_t)$ is given by: $q_1(\alpha_t) \propto f_y(y_t|\alpha_t)$. In this case, $f_*(x)$ and $q(x)$ in Sections 3.2 – 3.4 correspond to $f(\alpha_t|Y_{t-1})$ and $q_1(\alpha_t)$ in equation (6.19), respectively. $q_1(\alpha_t)$ is a known function because $f_y(y_t|\alpha_t)$ is derived from equation (6.1), and given $\alpha_{i,t-1|t-1}$ for $i = 1, 2, \cdots, N$ a random draw of α_t from $f(\alpha_t|Y_{t-1})$ is easily generated through the transition equation (6.2). Summarizing above, the random number generation method of $\alpha_{i,t|t}$ given $\alpha_{i,t-1|t-1}$ is as follows.

(i) Generate a random draw of α_0 from $f_\alpha(\alpha_0)$ if α_0 is stochastic and take a fixed value for α_0 otherwise.

(ii) Generate a random draw of α_t from $f(\alpha_t|Y_{t-1})$, denoted by α_t^*, which is a candidate of the $i$the random draw of α_t from $f(\alpha_t|Y_t)$, i.e., $\alpha_{i,t|t}$.

(iii) Based on α_t^*, obtain the ith random draw of α_t given Y_t, denoted by $\alpha_{i,t|t}$, where the sampling techniques introduced in Sections 3.2 – 3.4 are utilized.

(iv) Repeat (ii) and (iii) for $i = 1, 2, \cdots, N$.

(v) Repeat (ii) – (iv) for $t = 1, 2, \cdots, n$.

In Step (ii), α_t^* is generated from the sampling density $f(\alpha_t|Y_{t-1})$, which is performed as follows. When we want N random draws as in importance resampling, $\alpha_{i,t}^*$ is regarded as $\alpha_{i,t|t-1}$, which is obtained from the transition equation $\alpha_{i,t|t-1} = p_t(\alpha_{i,t-1|t-1}, \eta_{i,t})$ by generating a random draw of η_t, i.e., $\eta_{i,t}$. In the case where we need more than N random draws as in rejection sampling and the Metropolis-Hastings algorithm, the ith random draw, i.e., $\alpha_{i,t}^* = \alpha_{i,t|t-1}$, is obtained from the transition equation $\alpha_{i,t|t-1} = f_t(\alpha_{j,t-1|t-1}, \eta_{i,t})$ by choosing j with probability $1/N$ and generating a random draw $\eta_{i,t}$. Thus, $\alpha_{i,t|t}$ can be generated based on $\alpha_{i,t|t-1}$. Gordon, Salmond and Smith (1993), Kitagawa (1996, 1998) and Kitagawa and Gersch (1996) proposed the IR filter based on equation (6.19).

When we have a structural change or an outlier at time t, the present sample $f(y_t|\alpha_t)$ is far from $f(\alpha_t|Y_{t-1})$. In this case, for IR and MH the random draws of α_t from $f(\alpha_t|Y_t)$ become unrealistic because the reasonable random draws of α_t cannot be obtained from $f(\alpha_t|Y_{t-1})$, and for RS it takes a lot of time computationally because the acceptance probability becomes very small. In addition, when a random draw of η_t is not easily obtained, it might be difficult to generate a random draw of α_t from $f(\alpha_t|Y_{t-1})$. As for the second representation of the filtering density, therefore, we explicitly introduce the sampling density of α_t, i.e., $f_*(\alpha_t|\alpha_{t-1})$, to obtain more plausible random draws. Substituting the prediction equation (6.13) into the updating equation (6.14), we have the following equation:

$$\begin{aligned} f(\alpha_t|Y_t) &\propto f_y(y_t|\alpha_t) f(\alpha_t|Y_{t-1}) \\ &= \int f_y(y_t|\alpha_t) f_\alpha(\alpha_t|\alpha_{t-1}) f(\alpha_{t-1}|Y_{t-1})\, \mathrm{d}\alpha_{t-1} \\ &\propto \int f(\alpha_t, \alpha_{t-1}|Y_t)\, \mathrm{d}\alpha_{t-1} \end{aligned}$$

Moreover, eliminating the integration with respect to α_{t-1}, the joint density of α_t and α_{t-1} given Y_t, i.e., $f(\alpha_t, \alpha_{t-1}|Y_t)$, is written as:

$$f(\alpha_t, \alpha_{t-1}|Y_t) \propto q_2(\alpha_t, \alpha_{t-1}) f_*(\alpha_t|\alpha_{t-1}) f(\alpha_{t-1}|Y_{t-1}), \tag{6.20}$$

where $q_2(\alpha_t, \alpha_{t-1})$ is represented by: $q_2(\alpha_t, \alpha_{t-1}) \propto f_y(y_t|\alpha_t) f_\alpha(\alpha_t|\alpha_{t-1}) / f_*(\alpha_t|\alpha_{t-1})$. In equation (6.20), $f_*(\alpha_t|\alpha_{t-1}) f(\alpha_{t-1}|Y_{t-1})$ is taken as the sampling density. When N random draws of α_{t-1} given Y_{t-1}, i.e., $\alpha_{i,t-1|t-1}$ for $i = 1, 2, \cdots, N$, are available, generating a random draw of α_{t-1} from $f(\alpha_{t-1}|Y_{t-1})$ is equivalent to choosing one out of the N random draws $(\alpha_{1,t-1|t-1}, \alpha_{2,t-1|t-1}, \cdots, \alpha_{N,t-1|t-1})$ with equal probability weight. Given $\alpha_{i,t-1|t-1}$, a random draw of α_t (say $\alpha_{i,t}^*$) is generated from $f_*(\alpha_t|\alpha_{i,t-1|t-1})$. Thus, since

the functional form of $q_2(\alpha_t, \alpha_{t-1})$ is known and the random draws of (α_t, α_{t-1}) are generated from $f_*(\alpha_t|\alpha_{t-1})f(\alpha_{t-1}|Y_{t-1})$, the random draws of (α_t, α_{t-1}) from $f(\alpha_t, \alpha_{t-1}|Y_t)$ can be obtained through RS, IR or MH. The ith random draw of (α_t, α_{t-1}) from $f(\alpha_t, \alpha_{t-1}|Y_t)$ is denoted by $(\alpha_{i,t|t}, \alpha_{i,t-1|t})$. The random draw which we want is $\alpha_{i,t|t}$, not $\alpha_{i,t-1|t}$. Note that a random draw of α_t from $f(\alpha_t, \alpha_{t-1}|Y_t)$ is equivalent to that of α_t from $f(\alpha_t|Y_t)$. Furthermore, we point out that the appropriately chosen sampling density might be taken as $f_*(\alpha_t|\alpha_{t-1}) = f_*(\alpha_t)$, That is, the sampling density $f_*(\alpha_t|\alpha_{t-1})$ does not necessarily depend on α_{t-1}. As discussed above, for RS we need to compute the supremum of q_2 with respect to α_t and α_{t-1}. Sometimes, RS is not feasible when equation (6.20) is adopted. Therefore, for equation (6.20), IR or MH is recommended, rather than RS.

The random number generation method of $\alpha_{i,t|t}$ given $\alpha_{i,t-1|t-1}$ is as follows.

(i) Generate a random draw of α_0 from $f(\alpha_0)$ if α_0 is stochastic and take a fixed value for α_0 otherwise.

(ii) Generate a random draw of (α_t, α_{t-1}), denoted by $(\alpha_t^*, \alpha_{t-1}^*)$, from the sampling density $f_*(\alpha_t)f(\alpha_{t-1}|Y_{t-1})$.

(iii) Based on $(\alpha_t^*, \alpha_{t-1}^*)$, obtain the ith random draw of α_t given Y_t, denoted by $\alpha_{i,t|t}$, where the sampling techniques introduced in Sections 3.2 – 3.4 are utilized.

(iv) Repeat (ii) and (iii) for $i = 1, 2, \cdots, N$.

(v) Repeat (ii) – (iv) for $t = 1, 2, \cdots, n$.

For random number generation from $f(\alpha_{t-1}|Y_{t-1})$ in Step (ii), a filtering random draw of α_{t-1} is randomly chosen out of $\alpha_{1,t-1|t-1}$, $\alpha_{2,t-1|t-1}$, $\cdots$, $\alpha_{N,t-1|t-1}$ with probability $1/N$. $f(\alpha_{t-1}|Y_{t-1})$ can be utilized as the sampling density. Repeating this procedure, more than N random draws are also possible for random number generation from the filtering density $f(\alpha_{t-1}|Y_{t-1})$. This random number generation procedure is utilized in rejection sampling and the Metropolis-Hastings algorithm.

6.3.2 Smoothing

Given $\alpha_{i,t+1|n}$, we consider generating $\alpha_{i,t|n}$. Note that the smoothing random draws at time n, i.e., endpoint, are equivalent to the filtering random draws at time n, where both are denoted by $\alpha_{i,n|n}$.

Based on equation (6.15), we have three representations on the smoothing density. By eliminating the integration with respect to α_{t+1} from equation (6.15), the first representation of $f(\alpha_{t+1}, \alpha_t|Y_n)$ is as follows:

$$f(\alpha_{t+1}, \alpha_t|Y_n) \propto q_3(\alpha_{t+1}, \alpha_t)f(\alpha_t|Y_t)f(\alpha_{t+1}|Y_n), \tag{6.21}$$

where q_3 is represented by: $q_3(\alpha_{t+1}, \alpha_t) \propto f_\alpha(\alpha_{t+1}|\alpha_t)/f(\alpha_{t+1}|Y_t)$. For evaluation of $f(\alpha_{t+1}|Y_t)$ in $q_3(\alpha_{t+1}, \alpha_t)$ and $q_4(\alpha_{t+1}, \alpha_t)$ in equations (6.21) and (6.23), from equation

(6.13) we can use the following Monte Carlo integration:

$$f(\alpha_{t+1}|Y_t) = \int f_\alpha(\alpha_{t+1}|\alpha_t) f(\alpha_t|Y_t)\, \mathrm{d}\alpha_t \approx \frac{1}{N'} \sum_{j=1}^{N'} f_\alpha(\alpha_{t+1}|\alpha_{j,t|t}), \tag{6.22}$$

where N' is not necessarily equal to N. To reduce the computational disadvantage, N' may be less than N. For instance, the first N' random draws are chosen for evaluation of the integration above, because $\alpha_{1,t|t}$, $\alpha_{2,t|t}$, $\cdots$, $\alpha_{N,t|t}$ are random in order. Equation (6.22) implies that smoothing is N' times as computer-intensive as filtering. Thus, at each time period t, the order of computation is given by $N \times N'$ for smoothing. Remember that the order of computation is N for filtering.

In equation (6.21), the sampling density is given by $f(\alpha_{t+1}|Y_n) f(\alpha_t|Y_t)$. That is, a random draw of α_t is sampled from $f(\alpha_t|Y_t)$, while that of α_{t+1} is from $f(\alpha_{t+1}|Y_n)$. Sampling a random draw of α_t from $f(\alpha_t|Y_t)$ is equivalent to choosing one of $\alpha_{i,t|t}$, $i = 1, 2, \cdots, N$, with equal probability. A random draw of α_{t+1} from $f(\alpha_{t+1}|Y_n)$ is also generated by choosing one of $\alpha_{i,t+1|n}$, $i = 1, 2, \cdots, N$, with equal probability. Thus, random draws of α_t and α_{t+1} are generated separately.

Furthermore, replacing $f(\alpha_t|Y_t)$ in equation (6.21) by equation (6.19), the second representation of $f(\alpha_{t+1}, \alpha_t|Y_n)$ is written as:

$$f(\alpha_{t+1}, \alpha_t|Y_n) \propto q_4(\alpha_{t+1}, \alpha_t) f(\alpha_t|Y_{t-1}) f(\alpha_{t+1}|Y_n), \tag{6.23}$$

where q_4 is given by: $q_4(\alpha_{t+1}, \alpha_t) \propto f_y(y_t|\alpha_t) f_\alpha(\alpha_{t+1}|\alpha_t) / f(\alpha_{t+1}|Y_t) \propto q_1(\alpha_t) q_3(\alpha_{t+1}, \alpha_t)$. Similarly, in equation (6.23), $f(\alpha_{t+1}|Y_n) f(\alpha_t|Y_{t-1})$ is taken as the sampling density. A random draw of α_t is generated from $\alpha_{i,t|t-1} = p_t(\alpha_{i,t-1|t-1}, \eta_{i,t})$ in the case where exactly N random draws are required and $\alpha_{i,t|t-1} = p_t(\alpha_{j,t-1|t-1}, \eta_{i,t})$ when more than N random draws are needed, where $\eta_{i,t}$ denotes the ith random draw of η_t and j is chosen randomly out of 1, 2, $\cdots$, N. Sampling a random draw of α_{t+1} from $f(\alpha_{t+1}|Y_n)$ leads to choosing one of $\alpha_{i,t+1|n}$, $i = 1, 2, \cdots, N$, randomly.

Equation (6.21) is different from equation (6.23) with respect to the sampling density of α_t, i.e., the former is based on $f(\alpha_t|Y_t)$ while the latter is $f(\alpha_t|Y_{t-1})$. From equation (6.21) or (6.23), we can generate the random draw of (α_{t+1}, α_t) from $f(\alpha_{t+1}, \alpha_t|Y_n)$, which is denoted by $(\alpha_{i,t+1|n}, \alpha_{i,t|n})$. The random draw which we need is $\alpha_{i,t|n}$ because we already have $\alpha_{i,t+1|n}$ at this stage. Thus, given $\alpha_{i,t+1|n}$, $\alpha_{i,t|n}$ is generated. Repeating the procedure for $i = 1, 2, \cdots, N$, we can obtain $\alpha_{i,t|n}$ for $i = 1, 2, \cdots, N$ by the backward recursion.

In general, filtering is approximately close to smoothing when t approaches n, because Y_t approaches Y_n as t goes to n. Therefore, in order to obtain the smoothing random draws around n, it might be plausible to take $f(\alpha_t|Y_t)$ for equation (6.21) and $f(\alpha_t|Y_{t-1})$ for equation (6.23) as the sampling density of α_t. However, when t goes to the starting point, possibly $f(\alpha_t|Y_t)$ or $f(\alpha_t|Y_{t-1})$ is quite different from $f(\alpha_t|Y_n)$. In the third representation, therefore, another sampling density $f_*(\alpha_t|\alpha_{t-1}, \alpha_{t+1})$ is introduced to improve the problem above. Substituting equation (6.13) into equation (6.15) and

Table 6.2: $f(\cdot)$ and $q(\cdot)$ for Densities (6.19) – (6.21), (6.23) and (6.24)

	x	$f(x)$	$q(x)$
(6.19)	α_t	$f(\alpha_t\|Y_t)$	$f_y(y_t\|\alpha_t)$
(6.20)	(α_t, α_{t-1})	$f(\alpha_t, \alpha_{t-1}\|Y_t)$	$\frac{f_y(y_t\|\alpha_t) f_\alpha(\alpha_t\|\alpha_{t-1})}{f_*(\alpha_t\|\alpha_{t-1})}$
(6.21)	(α_{t+1}, α_t)	$f(\alpha_{t+1}, \alpha_t\|Y_n)$	$\frac{f_\alpha(\alpha_{t+1}\|\alpha_t)}{f(\alpha_{t+1}\|Y_t)}$
(6.23)	(α_{t+1}, α_t)	$f(\alpha_{t+1}, \alpha_t\|Y_n)$	$\frac{f_y(y_t\|\alpha_t) f_\alpha(\alpha_{t+1}\|\alpha_t)}{f(\alpha_{t+1}\|Y_t)}$
(6.24)	$(\alpha_{t+1}, \alpha_t, \alpha_{t-1})$	$f(\alpha_{t+1}, \alpha_t, \alpha_{t-1}\|Y_n)$	$\frac{f_y(y_t\|\alpha_t) f_\alpha(\alpha_t\|\alpha_{t-1}) f_\alpha(\alpha_{t+1}\|\alpha_t)}{f_*(\alpha_t\|\alpha_{t-1}, \alpha_{t+1}) f(\alpha_{t+1}\|Y_t)}$

eliminating the two integrations with respect to α_{t+1} and α_{t-1}, the joint density of α_{t+1}, α_t and α_{t-1} given Y_n, i.e., $f(\alpha_{t+1}, \alpha_t, \alpha_{t-1}|Y_n)$, is obtained as:

$$f(\alpha_{t+1}, \alpha_t, \alpha_{t-1}|Y_n) \propto q_5(\alpha_{t+1}, \alpha_t, \alpha_{t-1}) f(\alpha_{t-1}|Y_{t-1}) f_*(\alpha_t|\alpha_{t-1}, \alpha_{t+1}) f(\alpha_{t+1}|Y_n), \tag{6.24}$$

where q_5 is given by: $q_5(\alpha_{t+1}, \alpha_t, \alpha_{t-1}) \propto q_4(\alpha_{t+1}, \alpha_t) f_\alpha(\alpha_t|\alpha_{t-1}) / f_*(\alpha_t|\alpha_{t-1}, \alpha_{t+1})$. In equation (6.24), $f(\alpha_{t+1}|Y_n) f_*(\alpha_t|\alpha_{t-1}, \alpha_{t+1}) f(\alpha_{t-1}|Y_{t-1})$ is taken as the sampling density. After random draws of α_{t+1} and α_{t-1} are mutually independently generated from $f(\alpha_{t+1}|Y_n)$ and $f(\alpha_{t-1}|Y_{t-1})$, respectively, a random draw of α_t is sampled from another sampling density $f_*(\alpha_t|\alpha_{t-1}, \alpha_{t+1})$. Thus, $(\alpha_{i,t+1|n}, \alpha_{i,t|n}, \alpha_{i,t-1|n})$ is generated from equation (6.24), but the random draw which we want is $\alpha_{i,t|n}$ because $\alpha_{i,t+1|n}$ is already available at this stage and $\alpha_{i,t-1|n}$ can be obtained at the next one. $f_*(\alpha_t|\alpha_{t-1}, \alpha_{t+1}) = f_*(\alpha_t)$ is also a possible candidate of the appropriately chosen sampling density.

6.3.3 Discussion

Both equations (6.19) and (6.20) are related to filtering while equations (6.21), (6.23) and (6.24) correspond to smoothing. As shown in Sections 6.3.1 and 6.3.2, the random draws of α_t are generated from equations (6.19) – (6.21), (6.23) and (6.24). The correspondence between $f(\cdot)$ and $q(\cdot)$ in equation (3.2) of Section 3.2 is summarized in Table 6.2, where x denotes the random variable, $f(x)$ is the target density and $q(x)$ represents the ratio of the kernel and the sampling density.

The IR filter which uses equation (6.19) is proposed by Gordon, Salmond and Smith (1993), Kitagawa (1996, 1998) and Kitagawa and Gersch (1996). The RS filter based on (6.19) is introduced by Tanizaki (1996, 1999, 2001a), Tanizaki and Mariano (1998) and Hürzeler and Künsch (1998). In addition to the MH filter with (6.19), the

Table 6.3: Number of Generated Random Draws at Time t

Sampling Method	Filtering (6.19) and (6.20)	Smoothing (6.21), (6.23) and (6.24)
RS	$N(1+N_R)$	$N(1+N_R)\times N'$
IR	N	$N\times N'$
MH	$N+M$	$(N+M)\times N'$

IR, RS and MH filters based on equation (6.20) and the IR, RS and MH smoothers with equations (6.21), (6.23) and (6.24) are proposed by Tanizaki (2001a), where the RS filters and smoothers proposed by Tanizaki (1996, 1999), Tanizaki and Mariano (1998) are substantially extended to much less computational estimators.

Under the same number of random draws, it is easily expected that RS gives us the best estimates of the three sampling techniques while MH yields the worst estimates. The features of RS are that (i) we can generate random numbers from any density function when the supremum in the acceptance probability exists and (ii) precision of the random draws does not depend on choice of the sampling density (computational time depends on choice of the sampling density). For RS, however, the supremum has to be computed. We sometimes have the case where the supremum is not finite or the case where it is not easy to compute the supremum. Practically, it is difficult to obtain the supremums of q_2, q_3, q_4 and q_5 except for special cases. We cannot implement RS in this case. However, we can expect that there exists the supremum of q_1 in many cases, because it is highly possible to have the case the maximum value of $f_y(y_t|\alpha_t)$ with respect to α_t exists. Therefore, it might be recommended to apply RS to equation (6.19), rather than equations (6.20) – (6.21), (6.23) and (6.24). The average number of random draws generated at each time period is shown in Table 6.3. As discussed in Sections 3.2 and 3.4, both N_R and M depend on the functional form of the target and sampling densities. Remember that N_R denotes the number of rejections in rejection sampling and M represents the burn-in period in the MH algorithm. For IR, RS and MH, the order of computation time is shown in to Table 6.3. However, for IR, the order of computation time should be given by N^2 for filtering and $N^2\times N'$ for smoothing, i.e., the values in Table 6.3 multiplied by N, because we need to choose one out of N probability weights in order to obtain one random draw.

It might be possible to use different sampling techniques for filtering and smoothing. In other words, possibly we may obtain the RS filter based on equation (6.19) and the IR smoother based on equation (6.21). Moreover, for different time period, we may combine equations (6.19) and (6.20) for filtering and equations (6.21), (6.23) and (6.24) for smoothing. To show an example, suppose that we have a structural change or an outlier at time period t', which implies that $f(\alpha_{t'}|Y_{t'-1})$ is far from $f(\alpha_{t'}|Y_{t'})$. In this case, if equation (6.19) is implemented, for IR and MH we cannot obtain the plausible random draws of $\alpha_{t'}$ from $f(\alpha_{t'}|Y_{t'})$ and for RS it extremely takes a lot of

computational time to have the random draws of $\alpha_{t'}$ from $f(\alpha_{t'}|Y_{t'})$. Therefore, as shown in equation (6.20), we can introduce another sampling density $f_*(\alpha_{t'}|\alpha_{t'-1})$ at time t' to avoid this problem. Depending on the situation which we have, we can switch from equation (6.19) to equation (6.20) at time t'. By combining different sampling techniques between filtering and smoothing or utilizing different sampling densities at different time periods, it might be expected that the obtained filtering and smoothing solutions give us more precise state estimates.

In addition, it is also useful for filtering to take another sampling density $f_*(\alpha_t|\alpha_{t-1})$ when it is not easy to generate a random draw of α_t from $f(\alpha_t|Y_{t-1})$. That is, even though the density function of η_t is known, we sometimes have the case where it is difficult to obtain random draws of η_t. In this case, we can easily deal with this problem by utilizing $f_*(\alpha_t|\alpha_{t-1})$. Thus, the filtering and smoothing procedures introduced in this chapter is very flexible and easy to use in practice.

As an alternative smoother, we have the fixed-interval smoother based on the **two-filter formula**, proposed by Kitagawa (1996), where forward and backward filtering are performed and combined to obtain the smoothing density. The smoother based on the two-filter formula is discussed in Appendix 6.4, p.382.

6.3.4 Estimation of Parameter

Now we discuss about estimation of the unknown parameter. The maximum likelihood estimation method is usually used for parameter estimation, which is shown as follows. Let θ be the unknown parameter to be estimated. Suppose that $f(Y_n)$ in equation (6.17) depends on θ. That is, we have $f(Y_n) \equiv f(Y_n;\theta)$. Denote $L(\theta) \equiv \log f(Y_n;\theta)$. For maximization, by the first order Taylor series expansion we need to have the following equation:

$$\begin{aligned}\frac{\partial L(\theta)}{\partial \theta} &= 0 \\ &\approx \frac{\partial L(\theta^*)}{\partial \theta} + \frac{\partial^2 L(\theta^*)}{\partial \theta \partial \theta'}(\theta - \theta^*).\end{aligned}$$

Therefore, for optimization, replacing θ^* and θ by $\theta^{(j)}$ and $\theta^{(j+1)}$, we can rewrite the above equation as follows:

$$\theta^{(j+1)} = \theta^{(j)} - \Big(\frac{\partial^2 L(\theta^{(j)})}{\partial \theta \partial \theta'}\Big)^{-1} \frac{\partial L(\theta^{(j)})}{\partial \theta},$$

where $\theta^{(j)}$ denotes the jth iteration of θ.

Moreover, according to the method of scoring, $-\dfrac{\partial^2 L(\theta^{(j)})}{\partial \theta \partial \theta'}$ is replaced by:

$$-\mathrm{E}\Big(\frac{\partial^2 L(\theta^{(j)})}{\partial \theta \partial \theta'}\Big) = \mathrm{E}\bigg(\Big(\frac{\partial L(\theta^{(j)})}{\partial \theta}\Big)\Big(\frac{\partial L(\theta^{(j)})}{\partial \theta}\Big)'\bigg),$$

which is, again, replaced by its estimate:

$$\sum_{t=1}^{n}\Big(\frac{\partial \log\Big(\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1}^{(j)})\Big)}{\partial\theta}\Big)\Big(\frac{\partial \log\Big(\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1}^{(j)})\Big)}{\partial\theta}\Big)' \approx \sum_{t=1}^{n}\Big(\frac{\Delta \log\Big(\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1}^{(j)})\Big)}{\Delta\theta^{(j)}}\Big)\Big(\frac{\Delta \log\Big(\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1}^{(j)})\Big)}{\Delta\theta^{(j)}}\Big)',$$

where $\alpha_{i,t|t-1}^{(j)}$ denotes the ith random draw of α_t from $f(\alpha_t|Y_{t-1})$ given $\theta^{(j)}$, and

$$\Delta \log\Big(\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1}^{(j)})\Big) = \log\Big(\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1}^{(j)})\Big) - \log\Big(\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1}^{(j-1)})\Big),$$
$$\Delta\theta^{(j)} = \theta^{(j)} - \theta^{(j-1)}.$$

Therefore, θ is numerically updated as follows:

$$\theta^{(j+1)} = \theta^{(j)} + \Bigg(\sum_{t=1}^{n}\Big(\frac{\Delta \log\Big(\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1}^{(j)})\Big)}{\Delta\theta^{(j)}}\Big)\Big(\frac{\Delta \log\Big(\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1}^{(j)})\Big)}{\Delta\theta^{(j)}}\Big)'\Bigg)^{-1} \times \Bigg(\sum_{t=1}^{n}\frac{\Delta \log\Big(\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1}^{(j)})\Big)}{\Delta\theta^{(j)}}\Bigg). \tag{6.25}$$

Thus, given $\theta^{(j)}$, we can obtain $f_y(y_t|\alpha_{i,t|t-1}^{(j)})$ in the recursive algorithm on filtering. Therefore, the parameter θ is estimated using equation (6.25).

We have shown the above optimization method. However, conventionally it is not easy to implement the maximization procedure discussed in this section, because $\alpha_{i,t|t-1}$, $i = 1, 2, \cdots, N$, depend on $\theta^{(j)}$ and accordingly the likelihood function evaluated at $\alpha_{i,t|t-1}$ for $i = 1, 2, \cdots, N$ and $t = 1, 2, \cdots, n$ is randomly distributed around the true likelihood function. In the likelihood function (6.18), $(1/N)\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1})$ is generally different from $\lim_{N\to\infty}(1/N)\sum_{i=1}^{N} f_y(y_t|\alpha_{i,t|t-1})$. Some other estimation procedures are concisely described as follows.

As shown above, the traditional maximization procedures such as the simple grid search and the Newton-Raphson optimization are conventionally used for maximization of the likelihood function. However, if the number of the parameters increases, these procedures are not helpful. Moreover, as Kitagawa (1998) pointed out, we encounter the following two problems: (i) the non-Gaussian smoother is computationally intensive and the repeated application of a numerical optimization procedure for evaluating the likelihood may make it almost impractical and (ii) the log-likelihood computed by the Monte Carlo filter is subject to a sampling error and accordingly precise maximum likelihood parameter estimates can be obtained only by using a very large number of particles or by parallel application of many Monte Carlo filters. To improve the two problems above, Kitagawa (1998) proposed estimating the state

vectors and the unknown parameters simultaneously, which estimation procedure is called the self-organizing filter and smoother in his paper. There, a vector of the unknown parameters is regarded as the state vector which movement is appropriately assumed by a researcher through the transition equation. One of the disadvantages in Kitagawa (1998) is that the estimated parameters are time-dependent even though the unknown parameters are assumed to be fixed over time in the underlying state space models.

As for another remark, the derivative-based methods such as the Newton-Raphson optimization procedure generally have a problem that there is no method of distinguishing between a local optimum and a global one. It is well known that the method of the simulated annealing can find the best among multiple solutions (see, for example, Kirkpatrick, Gelatt, Jr. and Vecchi (1983), Bohachevsky, Johnson and Stein (1986), Goffe, Ferrier and Rogers (1994) and Brooks and Morgan (1995)). In the case where the function to be maximized is known, the simulated annealing is a powerful tool. For the nonlinear and non-Gaussian state space models, however, we cannot obtain the explicit functional form of the likelihood function, because the integration is included in equation (6.17). Therefore, the simulated annealing suggested by Kirkpatrick, Gelatt, Jr. and Vecchi (1983) cannot be directly applied to this problem. In Tanizaki (2001b), the likelihood function (6.17) is regarded as a kernel of the density function with respect to the unknown parameters. Based on the likelihood function, random draws of the state vectors and those of the unknown parameters are generated, where Gibbs sampling and the Metropolis-Hastings algorithm may be taken for random number generation. Note that for random number generation it might be possible to use the other sampling techniques such as rejection sampling instead of the Metropolis-Hastings algorithm. See Geman and Geman (1984), Tanner and Wong (1987), Gelfand and Smith (1990), Arnold (1993), Boswell, Gore, Patil and Taillie (1993), Smith and Roberts (1993), O'Hagan (1994), Tierney (1994), Chib and Greenberg (1995), Geweke (1996, 1997) for the sampling techniques. Through the random draws of the unknown parameters, we numerically obtain the functional form of the likelihood function with respect to the unknown parameters, where the nonparametric density estimation technique is applied. Based on the nonparametric density estimation, for each unknown parameter the mode is obtained, which corresponds to the global maximum, i.e., the maximum likelihood estimate of the parameter. See Prakasa Rao (1983), Silverman (1986), Ullah (1988), Härdle (1990) and Izenman (1991) for the nonparametric density estimation. Thus, Tanizaki (2001b) proposed estimating the state vectors and the unknown parameters simultaneously, using the Monte Carlo optimization procedure. However, the problem in Tanizaki (2001b) is that convergence is very slow because the nonparametric density estimation is utilized.

The Bayesian approach is also a good tool to solve the estimation problem, which was proposed by Carlin, Polson and Stoffer (1992), Carter and Kohn (1994, 1996), Chib and Greenberg (1996) and Geweke and Tanizaki (2001). Assuming the appropriate prior densities for the parameters, random draws of the state vectors and the parameters are generated, and the arithmetic averages of the random draws of the

parameters are obtained. Moreover, the maximization procedure based on the EM algorithm is also one solution, which will be discussed later in Section 6.4.

6.4 Non-Recursive Algorithm

In Section 6.3, the prediction, filtering and smoothing formulas are represented as the recursive algorithms based on the density functions. In this section, we introduce an alternative solution to a nonlinear and nonnormal smoother, which is not represented by the conventional recursive algorithm. The non-recursive formulas on prediction, filtering and smoothing are described in Tanizaki (1996, 1997) and Tanizaki and Mariano (1998). However, we can easily show equivalence between both the algorithms (see Appendix 6.2 for the equivalence).

Let us define $A_t = \{\alpha_0, \alpha_1, \cdots, \alpha_t\}$, which is a set consisting of the state variables up to time t. Suppose that $f_\alpha(A_t)$ and $f_y(Y_t|A_t)$ are represented as:

$$f_\alpha(A_t) = \begin{cases} f_\alpha(\alpha_0)\prod_{s=1}^{t} f_\alpha(\alpha_s|\alpha_{s-1}), & \text{if } \alpha_0 \text{ is stochastic,} \\ \prod_{s=1}^{t} f_\alpha(\alpha_s|\alpha_{s-1}), & \text{otherwise,} \end{cases} \tag{6.26}$$

$$f_y(Y_t|A_t) = \prod_{s=1}^{t} f_y(y_s|\alpha_s). \tag{6.27}$$

Based on the two densities $f_\alpha(A_t)$ and $f_y(Y_t|A_t)$, the filtering and smoothing formulas can be derived. Tanizaki (1996, 1997) and Tanizaki and Mariano (1998) made an attempt to evaluate the prediction, filtering and smoothing estimates, generating random draws of A_n from $f_\alpha(A_n)$.

One-Step Ahead Prediction: The conditional density of A_t given Y_{t-1}, $f(A_t|Y_{t-1})$, is given by:

$$f(A_t|Y_{t-1}) = \frac{f(A_t, Y_{t-1})}{\int f(A_t, Y_{t-1})\,\mathrm{d}A_t} = \frac{f_\alpha(A_t)f_y(Y_{t-1}|A_{t-1})}{\int f_\alpha(A_t)f_y(Y_{t-1}|A_{t-1})\,\mathrm{d}A_t}, \tag{6.28}$$

where $f(A_t, Y_{t-1}) = f_\alpha(A_t)f_y(Y_{t-1}|A_{t-1})$ is utilized in the second equality of equation (6.28).

Therefore, integrating $f(A_t|Y_{t-1})$ with respect to A_{t-1}, the one-step ahead prediction density is written as:

$$f(\alpha_t|Y_{t-1}) = \int f(A_t|Y_{t-1})\,\mathrm{d}A_{t-1}. \tag{6.29}$$

Equation (6.29) can be derived from equations (6.13) and (6.14). This fact is discussed in Appendix 6.2. Usually, we want to evaluate $\mathrm{E}(\alpha_t|Y_{t-1})$, rather than $f(\alpha_t|Y_{t-1})$, which expectation is shown as $\int \alpha_t f(A_t|Y_{t-})\ \mathrm{d}A_t$.

Filtering: The conditional density of A_t given Y_t, $f(A_t|Y_t)$, is represented as:

$$f(A_t|Y_t) = \frac{f(A_t, Y_t)}{\int f(A_t, Y_t)\, \mathrm{d}A_t} = \frac{f_\alpha(A_t) f_y(Y_t|A_t)}{\int f_\alpha(A_t) f_y(Y_t|A_t)\, \mathrm{d}A_t}, \tag{6.30}$$

where $f(A_t, Y_t) = f_\alpha(A_t) f_y(Y_t|A_t)$ is utilized in the second equality of equation (6.30).

Therefore, integrating $f(A_t|Y_t)$ with respect to A_{t-1}, the filtering density is written as follows:

$$f(\alpha_t|Y_t) = \int f(A_t|Y_t)\, \mathrm{d}A_{t-1}. \tag{6.31}$$

Equation (6.31) can be derived from equations (6.13) and (6.14). See Appendix 6.2 for equivalence between (6.31) and (6.14).

Smoothing: The conditional density of A_n given Y_n, $f(A_n|Y_n)$, is obtained as follows:

$$f(A_n|Y_n) = \frac{f(A_n, Y_n)}{\int f(A_n, Y_n)\, \mathrm{d}A_n} = \frac{f_\alpha(A_n) f_y(y_n|A_n)}{\int f_\alpha(A_n) f_y(Y_n|A_n)\, \mathrm{d}A_n}. \tag{6.32}$$

Let us define $A_t^+ = \{\alpha_t, \alpha_{t+1}, \cdots, \alpha_n\}$, where A_t^+ satisfies the following properties: (i) $A_n = A_0^+$ and (ii) $A_n = \{A_t, A_{t+1}^+\}$ for $t = 0, 1, \cdots, n-1$. From equation (6.32), using A_{t+1}^+ the smoothing density at time t, i.e., $f(\alpha_t|Y_n)$, is given by:

$$f(\alpha_t|Y_n) = \frac{\iint f_\alpha(A_n) f_y(Y_n|A_n)\, \mathrm{d}A_{t-1}\, \mathrm{d}A_{t+1}^+}{\int f_\alpha(A_n) f_y(Y_n|A_n)\, \mathrm{d}A_n}, \tag{6.33}$$

for $t = 1, 2, \cdots, n$. Again, note that it is easy to derive the standard density-based smoothing algorithm (6.15) from equation (6.33). See Appendix 6.2 for equivalence between (6.15) from (6.33).

Mean, Variance and Likelihood Function: Using equation (6.28) for prediction, equation (6.30) for filtering and equation (6.32) for smoothing, evaluation of the conditional expectation of a function $g(\alpha_t)$ is given by:

$$\begin{aligned} \mathrm{E}(g(\alpha_t)|Y_s) &= \int g(\alpha_t) f(A_r|Y_s)\, \mathrm{d}A_r = \frac{\int g(\alpha_t) f(A_r, Y_s)\, \mathrm{d}A_r}{\int f(A_r, Y_s)\, \mathrm{d}A_r} \\ &= \frac{\int g(\alpha_t) f_\alpha(A_r) f_y(Y_s|A_s)\, \mathrm{d}A_r}{\int f_\alpha(A_r) f_y(Y_s|A_s)\, \mathrm{d}A_r}, \end{aligned}$$

for $(r, s) = (t, t-1), (t, t), (n, n)$.

In the case where equations (6.1) and (6.2) depends on an unknown parameter, the likelihood function to be maximized is written as:

$$f(Y_n) \equiv f(Y_n; \theta) = \int f(A_n, Y_n)\, \mathrm{d}A_n = \int f_\alpha(A_n) f_y(Y_n|A_n)\, \mathrm{d}A_n, \tag{6.34}$$

which corresponds to the denominator of equation (6.32) in the smoothing formula and moreover it is equivalent to the innovation form of the likelihood function given by equation (6.17). However, evaluation of (6.34) is not easy and it is not practical.

An alternative estimation method of an unknown parameter is known as the EM algorithm (Expectation-Maximization algorithm), which is very easy to use in this case. In the EM algorithm, the expected log-likelihood function is maximized with respect to the parameter, given all the observed data Y_n (see Dempster, Laird and Rubin (1977), Rund (1991) and Laird (1993) for the EM algorithm). That is, for the EM algorithm, the following expected log-likelihood function is maximized:

$$\begin{aligned} \mathrm{E}\Big(\log(f(A_n, Y_n))|Y_n\Big) &= \mathrm{E}\Big(\log(f_y(Y_n|A_n)f_\alpha(A_n))|Y_n\Big) \\ &= \int \log(f_y(Y_n|A_n)f_\alpha(A_n))f(A_n|Y_n)\,\mathrm{d}A_n. \end{aligned} \tag{6.35}$$

As for the features of the EM algorithm, it is known that the convergence speed is very slow but it quickly searches the neighborhood of the true parameter value. Shumway and Stoffer (1982) and Tanizaki (1989, 2000) applied the EM algorithm to the state space model in linear and normal case.

6.4.1 Smoothing

We have shown the smoothing density (6.32) in the previous section. In this section, we show how to generate random draws of A_n directly from $f(A_n|Y_n)$. According to the Gibbs sampling theory, random draws of A_n from $f(A_n|Y_n)$ are based on those of α_t from $f(\alpha_t|A_{t-1}, A_{t+1}^+, Y_n)$ for $t = 1, 2, \cdots, n$, which is derived from equations (6.26) and (6.27) and represented as the following density function:

$$\begin{aligned} f(\alpha_t|A_{t-1}, A_{t+1}^+, Y_n) &= \frac{f(A_n|Y_n)}{f(A_{t-1}, A_{t+1}^+|Y_n)} = \frac{f_y(Y_n|A_n)f_\alpha(A_n)}{\int f_y(Y_n|A_n)f_\alpha(A_n)\,\mathrm{d}\alpha_t} \\ &\propto \begin{cases} f_y(y_t|\alpha_t)f_\alpha(\alpha_t|\alpha_{t-1})f_\alpha(\alpha_{t+1}|\alpha_t), & \text{if } t = 1, 2, \cdots, n-1, \\ f_y(y_t|\alpha_t)f_\alpha(\alpha_t|\alpha_{t-1}), & \text{if } t = n \text{ (i.e., endpoint),} \end{cases} \end{aligned} \tag{6.36}$$

where the second equality of equation (6.36) utilizes equations (6.26) and (6.27). Thus, equation (6.36) implies that a kernel of $f(\alpha_t|A_{t-1}, A_{t+1}^+, Y_n)$ is given by $f_y(y_t|\alpha_t)$ $f_\alpha(\alpha_t|\alpha_{t-1})f_\alpha(\alpha_{t+1}|\alpha_t)$ when $t = 1, 2, \cdots, n-1$ and $f_y(y_t|\alpha_t)f_\alpha(\alpha_t|\alpha_{t-1})$ when $t = n$ (i.e., endpoint).

Using a kernel of $f(\alpha_t|A_{t-1}, A_{t+1}^+, Y_n)$, we consider generating random draws of A_n directly from $f(A_n|Y_n)$. Here, the Gibbs sampler is applied to random number generation. Let $\alpha_{i,t}$ be the ith random draw of the state vector at time t. Define $A_{i,t}$ and $A_{i,t}^+$ as $A_{i,t} = \{\alpha_{i,0}, \alpha_{i,1}, \cdots, \alpha_{i,t}\}$ and $A_{i,t}^+ = \{\alpha_{i,t}, \alpha_{i,t+1}, \cdots, \alpha_{i,n}\}$, respectively, which are the ith random draws of A_t and A_t^+. When it is not easy to generate random draws from equation (6.36), the sampling techniques such as importance resampling, rejection sampling and the Metropolis-Hastings are utilized. Here, using the Metropolis-Hastings algorithm we show the random number generation method from equation (6.36).

Let $f_*(z|x)$ be the sampling density, which is the conditional distribution of z given x. We should choose the sampling density $f_*(z|x)$ such that random draws can be easily and quickly generated. Define the acceptance probability $\omega(x, z)$ as follows:

$$\omega(x,z) = \begin{cases} \min\left(\dfrac{f(z|A_{i,t-1}, A^+_{i-1,t+1}, Y_n)/f_*(z|x)}{f(x|A_{i,t-1}, A^+_{i-1,t+1}, Y_n)/f_*(x|z)}, 1 \right), \\ \qquad\qquad \text{if } f(x|A_{i,t-1}, A^+_{i-1,t+1}, Y_n) f_*(z|x) > 0, \\ 1, \qquad\qquad \text{otherwise.} \end{cases}$$

To generate random draws from $f(A_n|Y_n)$, the following procedure is taken:

(i) Pick up appropriate values for $\alpha_{-M+1,0}$ and $\alpha_{-M,t}$, $t = 1, 2, \cdots, n$.

(ii) Generate a random draw z from $f_*(\cdot|\alpha_{i-1,t})$ and a uniform random draw u from the uniform distribution between zero and one.

(iii) Set $\alpha_{i,t} = z$ if $u \leq \omega(\alpha_{i-1,t}, z)$ and $\alpha_{i,t} = \alpha_{i-1,t}$ otherwise.

(iv) Repeat (ii) and (iii) for $t = 1, 2, \cdots, n$.

(v) Repeat (ii) – (iv) for $i = -M + 1, -M + 2, \cdots, N$, where M is known as the burn-in period.

Note that the Metropolis-Hastings algorithm is used in procedures (ii) and (iii). In procedure (i), typically, the smoothing estimates based on the extended Kalman filter are taken for $\alpha_{0,t}$, $t = 1, 2, \cdots, n$. $\alpha_{i,0}$ for $i = -M + 1, -M + 2, \cdots, N$ depend on the underlying assumption of α_0. That is, $\alpha_{i,0}$ for $i = -M+1, -M+2, \cdots, N$ are generated from $f_\alpha(\alpha_0)$ if α_0 is stochastic and they are fixed as α_0 for all i if α_0 is nonstochastic.

The last N random draws are utilized for a further analysis, because of the convergence property of the Markov chain Monte Carlo methods. Based on the random draws $\alpha_{i,t}$ for $i = 1, 2, \cdots, N$, evaluation of $\mathrm{E}(g(\alpha_t)|Y_n)$ is simply obtained as the arithmetic average of $g(\alpha_{i,t})$, $i = 1, 2, \cdots, N$, which is represented by:

$$\frac{1}{N}\sum_{i=1}^{N} g(\alpha_{i,t}) \longrightarrow \mathrm{E}(g(\alpha_t)|Y_n) = \int g(\alpha_t) f(\alpha_t|Y_n)\, \mathrm{d}\alpha_t,$$

where $g(\cdot)$ is a function, which is typically $g(\alpha_t) = \alpha_t$ for mean or $g(\alpha_t) = (\alpha_t - \alpha_{t|n})(\alpha_t - \alpha_{t|n})'$ for variance. Note that $\alpha_{t|n} \equiv \mathrm{E}(\alpha_t|Y_n) \approx (1/N)\sum_{i=1}^{N} \alpha_{i,t}$.

Usually, 10 – 20% of N is taken for M, which implies that the first M random draws are discarded. However, there is no general rule for choice of M and N.

6.4.2 Estimation of Parameter

As discussed above, the log-likelihood function is given by equation (6.34). For estimation of unknown parameters, however, it is easy that the conditional expectation

of the log-likelihood function given by equation (6.35) is maximized (i.e., EM algorithm). Using the random draws generated from $f(A_n|Y_n)$, equation (6.35) is evaluated as follows:

$$\int \log(f(A_n, Y_n)) f(A_n|Y_n)\, \mathrm{d}A_n = \mathrm{E}\Big(\log(f(A_n, Y_n))|Y_n\Big)$$
$$\approx \frac{1}{N}\sum_{i=1}^{N} \log(f_y(Y_n|A_{i,n}) f_\alpha(A_{i,n})). \qquad (6.37)$$

From computational point of view, it is sometimes very easy to maximize equation (6.37), rather than equation (6.34), with respect to the unknown parameter vector θ. Note that θ is included in $f(A_n, Y_n)$, which is explicitly written as $f(A_n, Y_n) \equiv f(A_n, Y_n; \theta)$.

Shumway and Stoffer (1982) applied the EM algorithm to the state-space model in linear and normal case. For the procedure suggested in this section, it is much easier to utilize the EM algorithm, rather than maximization of equation (6.34). As for the features of the **EM algorithm**, it is known that the convergence speed is very slow but it quickly searches the neighborhood of the true parameter value. However, as mentioned in Section 6.3.4, note that we still have the problem in which the log-likelihood (6.37) is subject to the sampling error and accordingly precise maximum likelihood parameter estimates cannot be obtained.

6.4.3 Discussion

For the Markov chain Monte Carlo method, numerous number of random draws have to be generated to obtain the same precision of the smoothing estimates as both the resampling procedure and the rejection sampling procedure. Generally, it is intractable to generate $\alpha_{i,t}$ from $f(\alpha_t|A_{i,t-1}, A^+_{i-1,t+1}, Y_n)$. In this case, there are some methods to generate random draws, i.e., importance resampling, rejection sampling and the Metropolis-Hastings algorithm. It is known that rejection sampling sometimes takes a long time computationally or it is not feasible in the case where the acceptance probability does not exist. Importance resampling is generally less computational than rejection sampling, but it is also quite computer-intensive. Therefore, in order to generate numerous random draws very quickly, we apply the Metropolis-Hastings algorithm in procedures (ii) and (iii).

The Metropolis-Hastings algorithm has the problem of specifying the sampling density, which is the crucial criticism. Several generic choices of the sampling density are discussed by Tierney (1994) and Chib and Greenberg (1995). We may take the following several candidates for the sampling density function $f_*(z|x)$. First, It might be natural to take the density function obtained from the transition equation (6.2), i.e., $f_*(z|x) = f_\alpha(z|\alpha_{i,t-1})$. In this case, $f_*(z|x)$ does not depend on x, i.e., $f_*(z|x) = f_*(z)$, which is called the independence chain. Second, it is also possible to utilize the extended Kalman smoothed estimates, i.e., $f_*(z|x) = N(a^*_{t|n}, c^2\Sigma^*_{t|n})$, which is also

the independence chain, where $a^*_{t|n}$ and $\Sigma^*_{t|n}$ denote the first and second moments (i.e., mean and variance) based on the extended Kalman smoothed estimates at time t and c is an appropriate constant value, which is called the tuning parameter. Third, we may take the sampling density called the random walk chain, i.e., $f_*(z|x) = f_*(z - x)$, which is written as $f_*(z|x) = N(x, c^2\Sigma^*_{t|n})$. Fourth, in the case where the state variable α_t lies on an interval, a uniform distribution between the interval might be taken as the sampling density. In any case, it is clear that choice of the sampling density influences convergence of the random draws generated from $f(\alpha_t|A_{t-1}, A^+_{t+1}, Y_n)$.

In this section, the filtering problem has not been discussed until now. The filtering procedure might be implemented as follows. Simply, replacing n by t in the procedures (i) – (v) of Section 6.4.1, the random draws from the filtering density $f(\alpha_t|Y_t)$ are given by $\alpha_{i,t}$, $i = -M + 1, -M + 2, \cdots, N$, where t corresponds to the endpoint in the procedures (i) – (v). Recall that the random draws obtained at the endpoint represent the filtering random draws. Therefore, in addition to the procedures (i) – (v), we should put the following procedure:

(vi) Repeat (i) – (v) for $t = 1, 2, \cdots, n$.

Accordingly, filtering is more computer-intensive than smoothing. In the standard density-based smoothing algorithm shown in Section 6.3, $f(\alpha_t|Y_{t-1})$ and $f(\alpha_t|Y_t)$ are required. After the one-step ahead prediction density $f(\alpha_t|Y_{t-1})$ and the filtering density $f(\alpha_t|Y_t)$ are computed for $t = 1, 2, \cdots, n$, the smoothing density $f(\alpha_t|Y_n)$ is obtained by the backward recursive algorithm for $t = n - 1, n - 2, \cdots, 1$. Thus, clearly smoothing is more computer-intensive than filtering in the conventional density-based recursive algorithm. However, according to the Markov chain Monte Carlo procedure shown above, it is much easier to compute smoothing, rather than filtering. For filtering and smoothing, computational burden is as follows. The number of iteration is given by $n \times (M + N)$ for smoothing and $\sum_{t=1}^{n}(M + N)t = (M + N)n(n - 1)/2$ for filtering. It seems that for smoothing the Markov chain Monte Carlo procedure is less computational than the density-based recursive algorithm. However, the Markov chain Monte-Carlo methods need a lot of random draws in general, compared with the independence Monte-Carlo methods such as importance sampling and rejection sampling, because in the Markov chain Monte-Carlo methods we usually discard the first M random draws and a random draw is positively correlated with the next random draw (remember that the ith random draw depends on the $(i - 1)$th random draw for both the Gibbs sampler and the Metropolis-Hastings algorithm). Moreover, in the case of the state space model, it is known that convergence of the Gibbs sampler is very slow (see Carter and Kohn (1994, 1996)), because α_t is highly correlated with α_{t-1}, α_{t-2}, $\cdots$, α_0 from the structure of the transition equation (6.2). Therefore, an extremely large number of random draws have to be generated for convergence, when we apply the Markov chain Monte Carlo methods to state space models.

As a final remark, note as follows. As mentioned above, the maximization problem based on the simulation techniques has the sampling error. Therefore, in this case it is quite difficult to obtain the true parameter estimate which gives us the maximum

likelihood. To improve this problem, recently the Bayesian approach is broadly used. In the Bayesian procedure, the Gibbs sampler is performed using $f(\alpha_t|A_{t-1}, A_{t+1}^+, Y_n) \equiv f(\alpha_t|A_{t-1}, A_{t+1}^+, Y_n, \theta)$ for $t = 1, 2, \cdots, n$ and $f(\theta|A_n, Y_n) \propto f(A_n|Y_n, \theta) f_\theta(\theta)$. Note that $f(A_n|Y_n) \equiv f(A_n|Y_n, \theta)$ as shown in equation (6.34) and $f_\theta(\theta)$ denotes the prior distribution of θ. Thus, the Bayesian approach is a helpful tool to estimate the unknown parameter. However, as discussed above, remember that the Gibbs sampler has the convergence problem when it is applied to the state space model.

6.5 Monte Carlo Studies

6.5.1 Simulation Procedure

The filters and smoothers suggested in this chapter are examined in this section. we take $N = 1000, 2000, 5000$ for the recursive algorithms and $N = 10000, 20000, 50000$ for the non-recursive algorithm. The simulation procedure is shown as follows.

(i) Generating random numbers of ϵ_t and η_t for $t = 1, 2, \cdots, n$, compute a set of data (y_t, α_t) from equations (6.1) and (6.2), where $n = 100$ is taken. We will discuss later about the functional form of (6.1) and (6.2).

(ii) Given the data set, obtain the filtering and smoothing estimates for both recursive and non-recursive algorithms.

(iii) Repeat (i) and (ii) G times and compare the root mean square error (RMSE), defined as: $\text{RMSE} = (1/n) \sum_{t=1}^{n} \text{MSE}_{t|s}^{1/2}$ for $s = t, n$, where $\text{MSE}_{t|s} = (1/G) \sum_{g=1}^{G} (\overline{\alpha}_{t|s}^{(g)} - \alpha_t^{(g)})^2$ and $\overline{\alpha}_{t|s}$ takes the estimated state mean while α_t denotes the artificially simulated state value which is obtained in (i). The superscript (g) denotes the gth simulation run and $G = 1000$ is taken.

We consider six functional forms for (6.1) and (6.2), i.e., Simulations I – VI. Simulations I – V are univariate cases while Simulation VI represents a multivariate case. In Simulations I – III and V, ϵ_t, η_t and α_0 are assumed to be mutually independently distributed as: $\epsilon_t \sim N(0, 1)$, $\eta_t \sim N(0, 1)$ and $\alpha_0 \sim N(0, 1)$. The true parameter value is set to be $\delta = 0.5, 0.9, 1.0$ in Simulations I and V, and $\delta = 0.5, 0.9$ in Simulations II and III.

Simulation I (Linear and Normal Model): The univariate system is: $y_t = \alpha_t + \epsilon_t$ and $\alpha_t = \delta\alpha_{t-1} + \eta_t$, where ϵ_t, η_t and α_0 are assumed to be mutually independently distributed as: $\epsilon_t \sim N(0, 1)$, $\eta_t \sim N(0, 1)$ and $\alpha_0 \sim N(0, 1)$.

Simulation II (ARCH Model): The model is given by: $y_t = \alpha_t + \epsilon_t$ and $\alpha_t = (\delta_0 + \delta\alpha_{t-1}^2)^{1/2}\eta_t$ for $\delta_0 > 0$, $0 \leq \delta < 1$ and $\delta_0 = 1 - \delta$ are taken, where ϵ_t, η_t and α_0 are assumed to be mutually independently distributed as: $\epsilon_t \sim N(0, 1)$, $\eta_t \sim N(0, 1)$ and $\alpha_0 \sim N(0, 1)$. y_t consists of the ARCH(1) process α_t and the error term ϵ_t. See Engle (1982) and Bollerslev, Engle and Nelson (1994) for the ARCH model.

Simulation III (Stochastic Volatility Model): We take the state space model as: $y_t = \exp(0.5\alpha_t)\epsilon_t$ and $\alpha_t = \delta\alpha_{t-1} + \eta_t$ for $0 \leq \delta < 1$, where ϵ_t, η_t and α_0 are assumed to be mutually independently distributed as: $\epsilon_t \sim N(0,1)$, $\eta_t \sim N(0,1)$ and $\alpha_0 \sim N(0,1)$. See Ghysels, Harvey and Renault (1996) for the stochastic volatility model.

Simulation IV (Nonstationary Growth Model): The system is: $y_t = \alpha_t^2/20 + \epsilon_t$ and $\alpha_t = \alpha_{t-1}/2 + 25\alpha_{t-1}/(1+\alpha_{t-1}^2) + 8\cos(1.2(t-1)) + \eta_t$, where ϵ_t, η_t and α_0 are mutually independently distributed as: $\epsilon_t \sim N(0,1)$, $\eta_t \sim N(0,10)$ and $\alpha_0 \sim N(0,10)$. This model is examined in Kitagawa (1987, 1996, 1998) and Carlin, Polson and Stoffer (1992), where the Gibbs sampler suggested by Carlin, Polson and Stoffer (1992) does not work at all (see, for example, Tanizaki (2003)).

Simulation V (Structural Change): The data generating process is given by: $y_t = \alpha_t + \epsilon_t$ and $\alpha_t = d_t + \delta\alpha_{t-1} + \eta_t$, but the estimated system is: $y_t = \alpha_t + \epsilon_t$ and $\alpha_t = \delta\alpha_{t-1} + \eta_t$, where $d_t = 1$ for $t = 21, 22, \cdots, 40$, $d_t = -1$ for $t = 61, 62, \cdots, 80$ and $d_t = 0$ otherwise. ϵ_t, η_t and α_0 are assumed to be mutually independently distributed as: $\epsilon_t \sim N(0,1)$, $\eta_t \sim N(0,1)$ and $\alpha_0 \sim N(0,1)$. This model corresponds to the case where the sudden shifts occur at time periods 21, 41, 61 and 81.

Simulation VI (Bivariate Non-Gaussian Model): We consider the following bivariate state space model: $y_t = \alpha_{1t}x_t + \alpha_{2t} + \epsilon_t$ and $\alpha_t = \alpha_{t-1} + \eta_t$, where $\alpha_t = (\alpha_{1t}, \alpha_{2t})'$ and $\eta_t = (\eta_{1t}, \eta_{2t})'$. Each density is assumed to be: $\epsilon_t \sim$ Logistic, which cumulative distribution is given by: $F(x) = (\exp(-x)+1)^{-1}$, $\eta_{1t} \sim N(0,1)$, $\eta_{2t} \sim t(3)$, and $x_t \sim U(0,1)$. For the initial value $\alpha_0 = (\alpha_{10}, \alpha_{20})'$, $\alpha_{10} \sim N(0,1)$ and $\alpha_{20} \sim t(3)$ are taken. ϵ_t, η_{1t}, η_{2t}, x_t, α_{10} and α_{20} are assumed to be mutually independent.

6.5.2 Results and Discussion

The results are in Tables 6.4 – 6.6, where δ in Simulations I – III and V is assumed to be known. Except for Table 6.6, the values in each table represent the RMSEs defined in Step (iii) of Section 6.5.1. The small RMSE indicates a good estimator, because RMSE represents a measure of precision of the state estimates. RMSE decreases as the number of random draws (i.e., N) increases, because the simulation errors disappear as N goes to infinity. For all the tables, F and S denote filtering and smoothing, respectively.

In Table 6.4, F(6.19) indicates the filter based on equation (6.19). EK denotes the **extended Kalman filter** in nonlinear cases (i.e., Simulations II – IV) and the standard **Kalman filter** in linear cases (i.e., Simulations I, V and VI). EK indicates that the nonlinear measurement and transition equations are linearized and applied to the standard Kalman filter algorithm. See Appendix 6.3 for the Kalman filter algorithm. As it is expected, EK should be the best estimator in the case of Simulation I, but in the

Table 6.4: Filtering (6.19)

Simulation	δ	$N \setminus M$	EK	F(6.19) RS	IR	MH 1000	MH 5000
I	0.5	1000	0.7295	0.7300	0.7303	0.7314	0.7312
		2000	—	0.7297	0.7300	0.7305	0.7306
		5000	—	0.7295	0.7297	0.7300	0.7301
	0.9	1000	0.7738	0.7742	0.7748	0.7760	0.7764
		2000	—	0.7739	0.7744	0.7750	0.7751
		5000	—	0.7739	0.7741	0.7744	0.7743
	1.0	1000	0.7870	0.7874	0.7881	0.7893	0.7899
		2000	—	0.7872	0.7877	0.7886	0.7886
		5000	—	0.7870	0.7873	0.7878	0.7876
II	0.5	1000	0.7040	0.6890	0.6902	0.6910	0.6913
		2000	—	0.6887	0.6894	0.6899	0.6901
		5000	—	0.6886	0.6888	0.6892	0.6892
	0.9	1000	0.6415	0.5334	0.5346	0.5368	0.5365
		2000	—	0.5331	0.5342	0.5348	0.5344
		5000	—	0.5331	0.5336	0.5340	0.5338
III	0.5	1000	1.1519	0.9336	0.9340	0.9343	0.9344
		2000	—	0.9334	0.9335	0.9338	0.9336
		5000	—	0.9333	0.9333	0.9335	0.9335
	0.9	1000	2.2612	1.1096	1.1104	1.1115	1.1120
		2000	—	1.1096	1.1096	1.1103	1.1101
		5000	—	1.1093	1.1094	1.1099	1.1100
IV		1000	22.249	4.6194	4.6666	4.7089	4.7393
		2000	—	4.6157	4.6358	4.6783	4.6623
		5000	—	4.6128	4.6281	4.6394	4.6341
V	0.5	1000	0.8246	0.8251	0.8263	0.8274	0.8272
		2000	—	0.8248	0.8256	0.8264	0.8259
		5000	—	0.8247	0.8250	0.8252	0.8254
	0.9	1000	0.8680	0.8688	0.8724	0.8735	0.8738
		2000	—	0.8683	0.8704	0.8718	0.8714
		5000	—	0.8681	0.8689	0.8696	0.8697
	1.0	1000	0.8760	0.8767	0.8813	0.8834	0.8831
		2000	—	0.8764	0.8785	0.8800	0.8799
		5000	—	0.8761	0.8774	0.8784	0.8781
VI	α_{1t}	1000	2.8216	2.7800	2.8293	2.8824	2.8950
		2000	—	2.7719	2.7929	2.8420	2.8348
		5000	—	2.7642	2.7842	2.7916	2.7967
	α_{2t}	1000	1.9841	1.9322	1.9964	2.0121	2.0039
		2000	—	1.9263	1.9625	1.9909	1.9724
		5000	—	1.9211	1.9410	1.9532	1.9507

1) F(6.19) denotes the filter with (6.19).
2) EK denotes the **extended Kalman filter**, where the nonlinear measurement and transition equations are linearized and applied to the standard Kalman filter algorithm. Simulations I, V and VI are linear in the state variable for the measurement and transition equations, and accordingly the Kalman filter algorithm is directly applied to Simulations I, V and VI. Note that EK does not depend on N. See Appendix 6.3 for the linear and normal case.

Table 6.5: Smoothing (6.21) with Filtering (6.19)

Simulation	δ	$N \setminus M$	EK	S(6.21)+F(6.19) RS	IR	MH 1000	MH 5000	GM 1000	GM 5000
I	0.5	1000	0.7067	0.7074	0.7076	0.7095	0.7093	0.7069	0.7070
		2000	—	0.7069	0.7072	0.7079	0.7082	0.7068	0.7069
		5000	—	0.7067	0.7069	0.7072	0.7073	0.7068	0.7068
	0.9	1000	0.6834	0.6841	0.6848	0.6874	0.6877	0.6842	0.6843
		2000	—	0.6839	0.6841	0.6853	0.6858	0.6841	0.6841
		5000	—	0.6836	0.6837	0.6844	0.6843	0.6840	0.6840
	1.0	1000	0.6714	0.6725	0.6731	0.6755	0.6765	0.6723	0.6724
		2000	—	0.6719	0.6724	0.6740	0.6741	0.6721	0.6722
		5000	—	0.6715	0.6717	0.6725	0.6724	0.6721	0.6722
II	0.5	1000	0.7040	0.6798	0.6813	0.6829	0.6832	0.6800	0.6800
		2000	—	0.6796	0.6803	0.6808	0.6813	0.6798	0.6799
		5000	—	0.6793	0.6797	0.6801	0.6800	0.6798	0.6798
	0.9	1000	0.6415	0.5154	0.5176	0.5204	0.5201	0.5215	0.5211
		2000	—	0.5150	0.5165	0.5172	0.5175	0.5208	0.5208
		5000	—	0.5150	0.5154	0.5160	0.5158	0.5207	0.5206
III	0.5	1000	1.1519	0.9049	0.9058	0.9070	0.9066	0.9042	0.9041
		2000	—	0.9044	0.9048	0.9052	0.9053	0.9040	0.9041
		5000	—	0.9040	0.9043	0.9045	0.9046	0.9040	0.9039
	0.9	1000	2.2612	0.9296	0.9345	0.9356	0.9359	0.9287	0.9288
		2000	—	0.9297	0.9316	0.9316	0.9321	0.9284	0.9286
		5000	—	0.9284	0.9293	0.9299	0.9302	0.9285	0.9284
IV		1000	19.063	1.7141	1.8663	1.9846	2.0548	10.937	10.847
		2000	—	1.7153	1.7813	1.8858	1.8400	10.904	10.966
		5000	—	1.7113	1.7533	1.7828	1.8296	10.909	10.925
V	0.5	1000	0.7528	0.7535	0.7546	0.7581	0.7576	0.7533	0.7533
		2000	—	0.7532	0.7536	0.7558	0.7554	0.7531	0.7530
		5000	—	0.7529	0.7530	0.7539	0.7541	0.7530	0.7530
	0.9	1000	0.6932	0.6944	0.6999	0.7041	0.7047	0.6940	0.6941
		2000	—	0.6938	0.6965	0.7000	0.7000	0.6939	0.6939
		5000	—	0.6936	0.6945	0.6962	0.6965	0.6938	0.6938
	1.0	1000	0.6790	0.6806	0.6870	0.6915	0.6916	0.6800	0.6800
		2000	—	0.6800	0.6833	0.6862	0.6860	0.6798	0.6798
		5000	—	0.6793	0.6815	0.6825	0.6827	0.6796	0.6797
VI	α_{1t}	1000	2.0540	2.0697	2.3780	2.2644	2.2728	2.3428	2.2919
		2000	—	2.0428	2.3433	2.1714	2.1523	2.2354	2.2419
		5000	—	2.0201	2.3031	2.0823	2.0941	2.1348	2.1089
	α_{2t}	1000	1.5460	1.4918	1.6774	1.6261	1.6158	1.6127	1.5957
		2000	—	1.4820	1.6424	1.5730	1.5526	1.5672	1.5645
		5000	—	1.4710	1.6027	1.5178	1.5241	1.5188	1.5097

1) S(6.21) denotes the smoother with (6.21), where $N' = N$ is taken.
2) For S(6.21)+F(6.19) of RS, F(6.19) utilizes RS but S(6.21) does IR.
3) GM denotes the Gibbs sampler with the MH algorithm, where $10 \times N$ random draws are generated in addition to M random draws. For example, $N = 500$ indicates that 10000 random draws are generated for GM. $M = 1000, 5000$ are chosen. The third column in the table (i.e., N) is applied to S(6.21)+F(6.19).

Table 6.6: Computation Time (minutes)

Simulation	δ	$N \setminus M$	F(6.19) RS	F(6.19) IR	F(6.19) MH 1000	S(6.21)+F(6.19) RS	S(6.21)+F(6.19) IR	S(6.21)+F(6.19) MH 1000	GM 5000
I	0.5	500	.007	.001	.003	.111	.113	.335	.031
		1000	.015	.001	.004	.429	.442	.884	.053
	0.9	500	.007	.001	.003	.118	.117	.351	.032
		1000	.015	.002	.005	.455	.465	.928	.053
	1.0	500	.008	.001	.004	.119	.118	.356	.032
		1000	.016	.002	.004	.463	.471	.940	.053
II	0.5	500	.008	.001	.003	.213	.213	.612	.046
		1000	.018	.002	.005	.834	.847	1.624	.077
	0.9	500	.009	.001	.004	.221	.222	.638	.045
		1000	.018	.003	.005	.868	.883	1.694	.076
III	0.5	500	.013	.002	.004	.126	.113	.320	.056
		1000	.028	.002	.006	.477	.451	.844	.093
	0.9	500	.015	.001	.005	.136	.123	.345	.057
		1000	.031	.003	.006	.516	.485	.913	.097
IV		500	.023	.001	.005	.280	.259	.793	.070
		1000	.048	.003	.006	1.074	1.032	2.102	.117
V	0.5	500	.014	.001	.003	.121	.113	.343	.032
		1000	.032	.002	.005	.454	.449	.957	.053
	0.9	500	.017	.001	.003	.131	.120	.362	.032
		1000	.039	.002	.004	.490	.477	.965	.054
	1.0	500	.019	.001	.004	.134	.122	.366	.032
		1000	.041	.002	.005	.498	.482	.974	.053
VI		500	.053	.005	.012	.290	.236	.710	.147
		1000	.125	.008	.016	1.071	.938	1.884	.244

1) All the values in Table 6.6 represent the arithmetic averages from 1000 simulation runs. That is, each value shows how many minutes a simulation run it takes.
2) For S(6.21)+F(6.19) of RS, F(6.19) utilizes RS but S(6.21) does IR.
3) For S(6.21), $N' = N$ is taken.
4) GM denotes the Gibbs sampler with the MH algorithm, where $20 \times N$ random draws are generated in addition to M random draws. For example, $N = 500$ indicates that 10000 random draws are generated for GM. $M = 5000$ is chosen. The third column in the table (i.e., N) is applied to F(6.19) and S(6.21)+F(6.19).
5) Windows 2000 SP2 Operating System, Pentium III 1GHz CPU and G77 (Gnu Fortran for GCC) Version 3.1 are used for all the computations. G77 and GCC are downloaded from `http://www.delorie.com/djgpp/`.

other cases RS should show the smallest RMSE of RS, IR and MH. For all of $\delta = 0.5$, 0.9, 1.0 in Simulation I, RS with $N = 5000$ is very close to EK, compared with IR and MH. Moreover, taking an example of $\delta = 0.5$ in Simulation I, $N = 1000$ of RS is equal to $N = 5000$ of MH, which implies that MH needs 5 times more random draws than RS to keep the same precision, or equivalently, the acceptance rate of RS is about 20% in average (this is a rough interpretation because RMSE is not a linear function of N). Similarly, in $\delta = 0.5$ of Simulation I, $N = 1000$ of RS is equal to $N = 2000$ of IR, which implies that IR needs twice as many random draws as RS. Taking Simulations IV and VI, $N = 1000$ of RS is better than $N = 5000$ of IR and MH. IR and MH need more than 5 times as many random draws as RS. Simulation VI represents a multivariate non-Gaussian case, where the RMSEs are shown for both α_{1t} and α_{2t} in Simulation VI of Table 6.4. As a result, we can find that RMSEs of RS are the smallest for all the simulation studies. Moreover, in the case where $M = 1000$ is compared with $M = 5000$ for MH, both cases are very similar to each other. This implies that $M = 1000$ is large enough for the burn-in period.

In Table 6.5, S(6.21)+F(6.19) indicates the smoother based on equation (6.21) after performing the filter represented by F(6.19). Because it is not easy to compute the supremum of $q_3(\alpha_{t+1}, \alpha_t)$, RS in S(6.21)+F(6.19) represents IR S(6.21) based on RS F(6.19), where RS F(6.19) indicates F(6.19) using RS and similarly IR S(6.21) denotes S(6.21) with IR. As shown in Tables 6.4 and 6.5, the filtering estimates are more volatile than the smoothing estimates, because smoothing uses more information than filtering. Moreover, the same facts found in Table 6.4 are also observed in Table 6.5. Because in Simulation I the system is linear and normal, EK gives us the best estimator. Ii is observed in Simulation I that $N = 5000$ of RS is very close to EK. Therefore, we can conclude that $N = 5000$ is enough large for RS. Comparing RS, IR and MH, in the case $\delta = 0.5$ in Simulation I, $N = 2000$ of RS is similar to $N = 5000$ of IR, and $N = 1000$ of RS is almost same as $N = 5000$ of MH. GM indicates the Gibbs sampler with the MH algorithm, discussed in Section 6.4.1. Since GM with $M = 1000$ is very close to GM with $M = 5000$ except for Simulation VI, we can conclude that $M = 1000$ is enough for Simulations I – V. However, in Simulation VI, GM with $M = 5000$ is clearly smaller than GM with $M = 1000$ for $N = 10000$, 20000, 50000. We can see that the random draws obtained by the Gibbs sampler are not converged. $M = 1000$ is not sufficient in the case of Simulation VI. In $\delta = 0.9$ of Simulation I, Simulation IV, and Simulation VI, GM with $M = 5000$ and $N = 50000$ is quite larger than RS with $N = 5000$. This result shows that GM sometimes does not perform well, depending on nonlinearity and nonnormality of the system.

Table 6.6 is related to Table 6.3. CPU times (minutes) are compared in the case of Simulations I – VI and $N = 500$, 1000, where F(6.19) and S(6.21) are examined for RS, IR and MH. RS S(6.21) is equivalent to IR S(6.21) with RS F(6.19). Table 6.6 shows that the CPU times (minutes) are quite different, depending on the underlying state space model and the parameter δ. Because of the burn-in period M, MH takes more time than IR. As for RS, the CPU times are proportional to the rejection rates, which depend on functional forms of the system, error terms and time periods. The

rejection rates in RS are different for each time period t, because the observed data y_t takes a different value for each t. Remember that in the case of RS F(6.19) the acceptance probability is given by $\omega(\alpha_t) = f_y(y_t|\alpha_t)/\sup_z f_y(y_t|z)$, which depends on the observed data y_t.

In Table 6.7, $f_*(\alpha_t|\alpha_{t-1})$ is explicitly introduced for filtering, but not for smoothing, where $f_*(\alpha_t|\alpha_{t-1})f(\alpha_{t-1}|Y_{t-1})$ is taken as the sampling density for filtering (remember that $f(\alpha_t|Y_{t-1})$ is utilized for the sampling density in Table 6.4) and $f(\alpha_t|Y_t)f(\alpha_{t+1}|Y_n)$ is used for smoothing. For F(6.20) in Simulations I – V, $f_*(\alpha_t|\alpha_{t-1}) = N(\alpha^*_{t|t}, c^2\Sigma^*_{t|t})$ and $c^2 = 4$ are taken, where $(\alpha^*_{t|t}, \Sigma^*_{t|t})$ denotes the mean and variance estimated by the extended Kalman filter, which is obtained by applying the linearized nonlinear measurement and transition equations directly to the standard Kalman filter formula (see, for example, Tanizaki (1996) and Tanizaki and Mariano (1996)). Especially, for F(6.20) in Simulation V, $f_*(\alpha_t|\alpha_{t-1})$ is chosen taking into account the sudden shifts. It might be appropriate to consider that $\alpha^*_{t|t}$ and $\Sigma^*_{t|t}$ have some information on the sudden shifts. Note that F(6.19) of Simulation V in Table 6.4 does not include any information on the sudden shifts in the sampling density, where $f_*(\alpha_t|\alpha_{t-1}) = N(\delta\alpha_{t-1}, 1)$ is taken. Thus, the sampling density is explicitly introduced in Table 6.7. Since it is generally difficult to compute the supremum of q_2, RS is not shown in Table 6.7. We can compare F(6.19) in Table 6.4 with F(6.20) in Table 6.7 for filtering, and S(6.21)+F(6.19) in Table 6.4 with S(6.21)+F(6.20) in Table 6.7 for smoothing. For Simulation I, the RMSEs in Table 6.7 are very close to those in Table 6.4. For almost all the cases of Simulations II and III, however, the RMSEs in Table 6.4 are slightly smaller than those in Table 6.7. However, for Simulation IV, F(6.20) does not work well at all, in which it may seem that the extended Kalman filter estimates $\alpha^*_{t|t}$ and $\Sigma^*_{t|t}$ are completely far from the true values $\alpha_{t|t}$ and $\Sigma_{t|t}$. For Simulation V, we consider the case where the data generating process is different from the estimated state space model, which is common in practice. For the sampling density, we take into account the sudden shifts in F(6.20) of Table 6.7, but not in F(6.19) of Table 6.4. The sampling density in F(6.20) of Table 6.7 corresponds to the case where we can guess the shifts from the extended Kalman filter estimates, while that in F(6.19) of Table 6.4 is the case where we ignore the shifts even though clearly we can observe some shifts from the data y_t. The case in Table 6.4 might be unrealistic while that in Table 6.7 is more plausible. For comparison between the two cases, Simulation V is taken in Tables 6.4 and 6.7. As it is expected, IR and MH in Table 6.7 are smaller than those in Table 6.4. Therefore, we can conclude that much improvement in the state estimates might be expected when the plausible sampling density $f_*(\alpha_t|\alpha_{t-1})$ is chosen. Only one case of the sampling density, $f_*(\alpha_t|\alpha_{t-1}) = N(\alpha^*_{t|1}, c^2\Sigma^*_{t|t})$, is shown in Table 6.7, although we can consider the other kinds of the sampling density. Furthermore, in Table 6.7, S(6.21)+F(6.20) is also examined, which should be compared with S(6.21)+F(6.19) in Table 6.5. In a lot of cases (except for $\delta = 0.9$ in Simulations II and III, and Simulation IV), S(6.21)+F(6.20) is better than S(6.21)+F(6.19). Especially, for small N, S(6.21)+F(6.20) shows a relatively good performance over S(6.21)+F(6.19).

In Table 6.8, we investigate for Simulations I – V how sensitive the approximation

Table 6.7: Filtering and Smoothing Using Sampling Density

Simulation	δ	$N \setminus M$	F(6.20) IR	F(6.20) MH 1000	S(6.21) IR	S(6.21) MH 1000
I	0.5	1000	0.7301	0.7304	0.7075	0.7084
		2000	0.7298	0.7298	0.7071	0.7074
		5000	0.7296	0.7297	0.7068	0.7069
	0.9	1000	0.7744	0.7754	0.6847	0.6864
		2000	0.7743	0.7744	0.6841	0.6850
		5000	0.7740	0.7741	0.6837	0.6840
	1.0	1000	0.7878	0.7887	0.6729	0.6749
		2000	0.7875	0.7874	0.6722	0.6730
		5000	0.7872	0.7874	0.6717	0.6720
II	0.5	1000	0.6891	0.6895	0.6801	0.6813
		2000	0.6889	0.6888	0.6800	0.6798
		5000	0.6887	0.6888	0.6794	0.6797
	0.9	1000	0.5525	0.5543	0.5378	0.5391
		2000	0.5497	0.5508	0.5344	0.5347
		5000	0.5466	0.5481	0.5310	0.5309
III	0.5	1000	0.9341	0.9352	0.9054	0.9072
		2000	0.9336	0.9341	0.9046	0.9058
		5000	0.9334	0.9334	0.9042	0.9044
	0.9	1000	1.1124	1.1176	0.9356	0.9435
		2000	1.1103	1.1138	0.9315	0.9354
		5000	1.1098	1.1110	0.9294	0.9315
IV		1000	14.2083	16.3691	14.2686	16.4000
		2000	14.1476	16.2388	14.2090	16.3000
		5000	14.0404	16.1196	14.1373	16.2115
V	0.5	1000	0.8252	0.8257	0.7533	0.7551
		2000	0.8250	0.8249	0.7531	0.7537
		5000	0.8248	0.8248	0.7528	0.7530
	0.9	1000	0.8690	0.8702	0.6951	0.6982
		2000	0.8688	0.8692	0.6941	0.6962
		5000	0.8683	0.8686	0.6935	0.6942
	1.0	1000	0.8771	0.8790	0.6820	0.6851
		2000	0.8767	0.8771	0.6807	0.6825
		5000	0.8763	0.8766	0.6797	0.6802

1) $f_*(\alpha_t|\alpha_{t-1}) = N(\alpha^*_{t|t}, c^2\Sigma^*_{t|t})$ and $c^2 = 4$ are taken, where $(\alpha^*_{t|t}, \Sigma^*_{t|t})$ denotes the mean and variance estimated by the extended Kalman filter, which is obtained by applying the linearized nonlinear measurement and transition equations directly to the standard Kalman filter formula (see, for example, Tanizaki (1996) and Tanizaki and Mariano (1996)).

2) S(6.21) in the above table indicates S(6.21)+F(6.20), where $N' = N$ is taken.

Table 6.8: S(6.21) with F(6.19): $N' = 1000, 2000, 5000$

Simulation	δ	N'	RS	IR	MH
I	0.5	1000	0.7067	0.7069	0.7072
		2000	0.7067	0.7069	0.7072
		5000	0.7067	0.7069	0.7072
	0.9	1000	0.6836	0.6838	0.6844
		2000	0.6836	0.6838	0.6844
		5000	0.6836	0.6837	0.6844
	1.0	1000	0.6716	0.6718	0.6725
		2000	0.6716	0.6717	0.6725
		5000	0.6715	0.6717	0.6725
II	0.5	1000	0.6793	0.6797	0.6802
		2000	0.6793	0.6796	0.6802
		5000	0.6793	0.6797	0.6801
	0.9	1000	0.5150	0.5156	0.5161
		2000	0.5149	0.5155	0.5160
		5000	0.5150	0.5154	0.5160
III	0.5	1000	0.9041	0.9044	0.9045
		2000	0.9041	0.9043	0.9045
		5000	0.9040	0.9043	0.9045
	0.9	1000	0.9285	0.9311	0.9303
		2000	0.9287	0.9302	0.9300
		5000	0.9284	0.9293	0.9299
IV		1000	1.7179	1.8152	1.8055
		2000	1.7174	1.7853	1.7855
		5000	1.7113	1.7533	1.7828
V	0.5	1000	0.7529	0.7529	0.7539
		2000	0.7529	0.7529	0.7539
		5000	0.7529	0.7530	0.7539
	0.9	1000	0.6936	0.6946	0.6962
		2000	0.6936	0.6946	0.6962
		5000	0.6936	0.6945	0.6962
	1.0	1000	0.6794	0.6816	0.6828
		2000	0.6793	0.6815	0.6827
		5000	0.6793	0.6815	0.6825

1) For RS, F(6.19) utilizes RS but S(6.21) does IR.
2) For MH, $M = 1000$ is taken.

Table 6.9: IR S(6.71) Based on IR F(6.19)

Simulation	δ	N	S(6.71) IR	Simulation	δ	N	S(6.71) IR
I	0.5	1000	0.7071	III	0.5	1000	0.9050
		2000	0.7069			2000	0.9046
		5000	0.7067			5000	0.9041
	0.9	1000	0.6843		0.9	1000	0.9353
		2000	0.6839			2000	0.9328
		5000	0.6836			5000	0.9312
	1.0	1000	0.6722	V	0.5	1000	0.7535
		2000	0.6719			2000	0.7532
		5000	0.6715			5000	0.7529
II	0.5	1000	0.6799		0.9	1000	0.6957
		2000	0.6796			2000	0.6944
		5000	0.6793			5000	0.6937
	0.9	1000	0.5156		1.0	1000	0.6816
		2000	0.5151			2000	0.6805
		5000	0.5150			5000	0.6796

- $f_*(\alpha_t) = N(\alpha_{t|t}, 1)$ is taken, where $\alpha_{t|t}$ is obtained from IR F(6.19).

of $f(\alpha_{t+1}|Y_t)$ in equation (6.22) is, where $N' = 1000, 2000, 5000$ and $N = 5000$ are taken. RS, IR and MH are used for the sampling method. $N' = 5000$ in Table 6.8 is equivalent to $N = 5000$ in Table 6.5. We have the result that $N' = 1000$ is very close to $N' = 2000, 5000$ in the RMSE criterion for all the three sampling methods. Since the order of computation is $N(1 + N_R) \times N'$ for RS smoothing, $N \times N'$ for IR smoothing, and $(N + M) \times N'$ for MH smoothing. we can reduce the computational burden by utilizing N' less than N.

Finally, as for the alternative fixed-interval smoother, Kitagawa (1996) proposed the smoother based on the **two-filter formula**, which can be also discussed in the same context. See Appendix 6.4 for detail discussion on the two-filter formula. In Table 6.9, the smoother based on the two-filter formula is examined for Simulations I – III and V. The sampling distribution $f_*(\alpha_t) = N(\alpha_{t|t}, 1)$ is utilized for the two-filter formula. where $\alpha_{t|t}$ represents the filtering estimate obtained from IR F(6.19). The two-filter formula does not necessarily require the sampling distribution. However, from computational point of view, it is easier to introduce the sampling distribution explicitly. Therefore, in Table 6.9 we utilize the sampling distribution as $f_*(\alpha_t) = N(\alpha_{t|t}, 1)$. After implementing IR F(6.19), we perform IR S(6.71) which is the smoother based on the two-filter formula. See Appendix 6.4 for S(6.71). Each value in Table 6.9 should be compared with that in IR S(6.21) of Table 6.5. As a result, IR S(6.71) performs better than IR S(6.21) in almost all the cases.

Figure 6.1: Nikkei Stock Average (Japanese Yen)

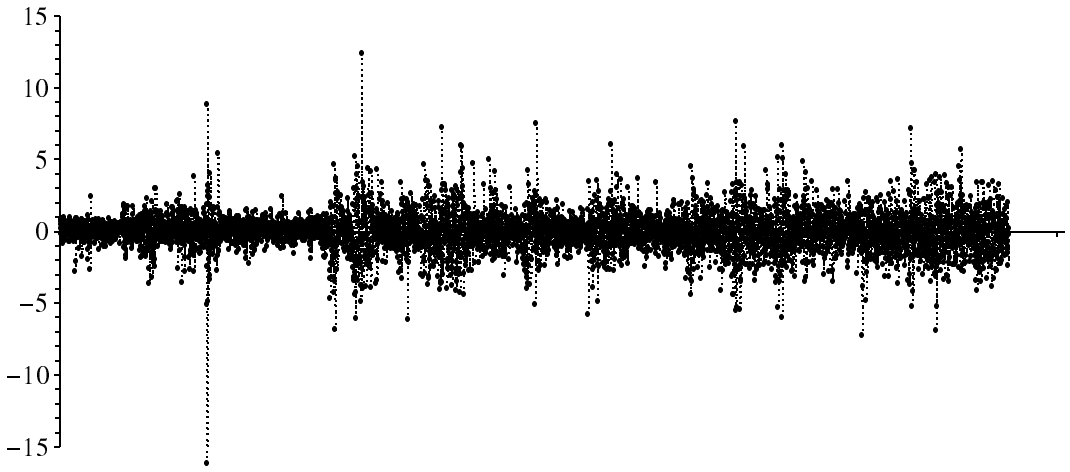
6.6 Empirical Example

In this section, we estimate the **Nikkei stock average** using the state space model. We assume that the growth rate of the Nikkei stock average consists of two components, i.e., the trend component and the noise component. Both components are unobservable. Using the state space model, the two components are estimated separately. It is known that the the Nikkei stock average describes Japanese business cycle. In this section, we show empirically that the trend component in the percent change of the Nikkei stock average moves along with Japanese business cycle.

6.6.1 Introduction

Figure 6.1 represents the daily movement of the Nikkei stock average (closing price, Japanese yen) from January 4, 1985 to January 31, 2003, where the vertical and horizontal lines indicate Japanese yen and time. In Figure 6.1, the Japanese stock price extraordinarily decreases from 25746.56 on October 19, 1987 to 21910.08 on October 20, 1987, which corresponds to Black Monday on October 19, 1987 in the New York stock market. However, the stock price has been increasing until the end of 1989. The largest closing price is 38915.87 on December 29, 1989. Until the end of 1989, we had the rapid economic expansion based on vastly inflated stock and land prices, so-called the **bubble economy**. Since then, the collapse of Japan's overheated stock and real estate markets has started. Especially, a drop in 1990 was really serious. That is, the stock price fell down from 38712.88 on January 4, 1990 to 20221.86 on October 1, 1990, which is almost 50% drop. Recently, since we had 20833.21 on April 12, 2000, the stock price has been decreasing. It is 8339.94 on January 31, 2003. Thus, the Japanese stock price increased up to the end of 1989, decreased until around March or

Figure 6.2: Percent Changes of Nikkei Stock Average (%)



April in 1992, was relatively stable up to April in 2000, and has declined since then.

Figure 6.2 shows the percent changes of the Nikkei stock average (%), where the daily data in Figure 6.1 are converted into the percent change data. Figure 6.2 displays the percent in the vertical line and the time period in the horizontal line. For comparison with Figure 6.1, Figure 6.2 takes the same time period as Figure 6.1, i.e., Figures 6.1 and 6.2 take the same scale in the horizontal line. −16.1% on October 20, 1987 is the largest drop (it is out of range in Figure 6.2), which came from Black Monday in the New York stoch market. 12.4% on October 2, 1990 is the largest rise, which corresponds to the unification of Germany occurred on October 1, 1990. The second largest rise in the stock price was 8.9% on October 21, 1987, which is a counter-reaction against the largest drop on October 20, 1987. Although there are some outliers, the percent changes up to the end of 1989 seem to be distributed within a relatively small range, i.e., variance of the percent change until the end of 1989 is samll. Especially, for recent few years the percent changes are more broadly distributed. Thus, we can see that the percent change data of the Japanese stock price are heteroscedastic. In any case, it may seem from Figure 6.2 that the percent change data are randomly distributed and accordingly there is not any trend in the percent changes. In the next section, using the state space model we aim to pick up the trend term which represents the Japanese business cycle.

6.6.2 Setup of the Model

In this section, the model is briefly described as an application of the state space model. The observed data y_t is denoted by:

$y_t =$ Percent Change of the Nikkei Stock Average from Time $t-1$ to Time t,

where the daily data from January 4, 1985 to January 31, 2003 are utilized and the percent changes from previous day are computed, i.e., the percent change data from January 5, 1985 to January 31, 2003 are available. The sample size is 4580. The estimation method discussed in Section 6.3, i.e., the recursive algorithm, is quite complicated in programming, although the state mean estimated by the recursive algorithm is very efficient in the sense that RMSE of the smoother discussed in 6.3.2 is much smaller than that of the smoother discussed in Section 6.4.1 (compare RS, IR and MH with GM in Table 6.5). Therefore, from simplicity of estimation, we adopt the non-recursive smoothing algorithm shown in Section 6.4.1 for the estimation method, where the Metropolis algorithm within the Gibbs sampler is implemented for the sampling method. In addition, as for the parameter estimation, the Bayesian approach is utilized because of its simplicity.

To see the movement of the percent changes of the Nikkei stock average, we consider three models, called Models 1 – 3, where we assume that the original data y_t consist of the true mean and the other irregular component for each time t. In this empirical example, we aim to estimate the percent change of the Nikkei stock average without any noise, which is denoted by α_t. For all Models 1 – 3, the flat priors are taken for all the unknown parameters. The posterior densities corresponding to the priors are derived as follows.

Model 1 (Time Varying Mean with Normal Error): As the first model, the percent change of the Nikkei stock average is taken as a sum of the time-dependent mean and the Gaussian noise, which is given by the following model:

$$\begin{aligned} &\text{(Measurement equation)} && y_t = \alpha_t + \epsilon_t, \\ &\text{(Transition equation)} && \alpha_t = \alpha_{t-1} + \eta_t, \end{aligned}$$

for $t = 1, 2, \cdots, n$, where $\epsilon_t \sim N(0, \sigma_\epsilon^2)$ and $\eta_t \sim N(0, \sigma_\eta^2)$.

Under the above state space model, the conditional distribution of y_t given α_t, $f_y(y_t|\alpha_t)$, and that of α_t given α_{-1}, $f_\alpha(\alpha_t|\alpha_{t-1})$, are derived from the measurement and transition equations, respectively, which distributions are represented as:

$$f_y(y_t|\alpha_t) = \frac{1}{\sqrt{2\pi\sigma_\epsilon^2}} \exp\Big(-\frac{1}{2\sigma_\epsilon^2}(y_t - \alpha_t)^2\Big), \tag{6.38}$$

$$f_\alpha(\alpha_t|\alpha_{t-1}) = \frac{1}{\sqrt{2\pi\sigma_\eta^2}} \exp\Big(-\frac{1}{2\sigma_\eta^2}(\alpha_t - \alpha_{t-1})^2\Big). \tag{6.39}$$

There are the three unknown parameters in the model, i.e., α_0, σ_ϵ^2 and σ_η^2. We consider estimating the three parameters by the Bayesian approach. As for the prior densities, all the parameters are assumed to be the flat priors, which implies that all the values can be uniformly taken. Especially, as for σ_ϵ^2 and σ_η^2, since variances do not take

non-positive values, the log of variance is assumed to be uniform. Thus, the prior distributions of α_0, σ_ϵ^2 and σ_η^2 are assumed to be as follows:

$$f_{\alpha_0}(\alpha_0) \propto \text{ const.}, \quad f_{\sigma_\epsilon}(\sigma_\epsilon^2) \propto \frac{1}{\sigma_\epsilon^2}, \quad f_{\sigma_\eta}(\sigma_\eta^2) \propto \frac{1}{\sigma_\eta^2}. \tag{6.40}$$

Using densities (6.38), (6.39) and (6.40), we can obtain the following posterior densities:

$$f(\alpha_t|A_{t-1}, A_{t+1}^+, \sigma_\epsilon^2, \sigma_\eta^2) \propto \begin{cases} \frac{1}{\sqrt{2\pi\sigma_\epsilon^2}} \exp\Big(-\frac{1}{2\sigma_\epsilon^2}(y_t - \alpha_t)^2\Big) \\ \quad \times \frac{1}{\sqrt{2\pi\sigma_\eta^2}} \exp\Big(-\frac{1}{2\sigma_\eta^2}(\alpha_{t+1} - \alpha_t)^2\Big) \\ \quad \times \frac{1}{\sqrt{2\pi\sigma_\eta^2}} \exp\Big(-\frac{1}{2\sigma_\eta^2}(\alpha_t - \alpha_{t-1})^2\Big), \\ \qquad\qquad \text{for } t = 1, 2, \cdots, n-1, \\ \frac{1}{\sqrt{2\pi\sigma_\epsilon^2}} \exp\Big(-\frac{1}{2\sigma_\epsilon^2}(y_t - \alpha_t)^2\Big) \\ \quad \times \frac{1}{\sqrt{2\pi\sigma_\eta^2}} \exp\Big(-\frac{1}{2\sigma_\eta^2}(\alpha_t - \alpha_{t-1})^2\Big), \\ \qquad\qquad \text{for } t = n, \end{cases}$$

$$\alpha_0 \mid A_1^+, \sigma_\epsilon^2, \sigma_\eta^2 \sim N(\alpha_1, \sigma_\eta^2),$$

$$\frac{1}{\sigma_\epsilon^2} \mid A_n, \sigma_\eta^2 \sim G\Big(\frac{n}{2}, \frac{2}{\sum_{t=1}^n (y_t - \alpha_t)^2}\Big),$$

$$\frac{1}{\sigma_\eta^2} \mid A_n, \sigma_\epsilon^2 \sim G\Big(\frac{n}{2}, \frac{2}{\sum_{t=1}^n (\alpha_t - \alpha_{t-1})^2}\Big),$$

where $A_t = \{\alpha_0, \alpha_1, \cdots, \alpha_t\}$ and $A_{t+1}^+ = \{\alpha_{t+1}, \alpha_{t+2}, \cdots, \alpha_n\}$. Using the above four posterior densities, the Gibbs sampler can be performed to generate random draws from the above four posterior densities. It is easy to generate the random draws of α_t from $f(\alpha_t|A_{t-1}, A_{t+1}^+, \sigma_\epsilon^2, \sigma_\eta^2)$, because $\alpha_t|A_{t-1}, A_{t+1}^+, \sigma_\epsilon^2, \sigma_\eta^2 \sim N(b_t, \tau_t^2)$, where $b_t = (y_t/\sigma_\epsilon^2 + \alpha_{t+1}/\sigma_\eta^2 + \alpha_{t-1}/\sigma_\eta^2)/\tau_t^2$ and $\tau_t^2 = 1/(1/\sigma_\epsilon^2 + 2/\sigma_\eta^2)$ for $t = 1, 2, \cdots, n$ and $b_t = (y_t/\sigma_\epsilon^2 + \alpha_{t-1}/\sigma_\eta^2)/\tau_t^2$ and $\tau_t^2 = 1/(1/\sigma_\epsilon^2 + 1/\sigma_\eta^2)$ for $t = n$. However, for comparison with Models 2 and 3, we utilize the Metropolis-Hastings algorithm to generate random draws from $f(\alpha_t|A_{t-1}, A_{t+1}^+, \sigma_\epsilon^2, \sigma_\eta^2)$.

Model 2 (Time Varying Mean with Stochastic Volatility Error): In Model 1, we have assumed that the error term in the measurement equation, i.e., ϵ_t, is normal and homoscedastic. From Figure 6.2, however, we can see that there are a lot of outliers. This fact implies that the true distribution of the error term (i.e., irregular component) should be distributed more broadly than the normal distribution. In fact, the average,

standard error, skewness and kurtosis obtained from the percent changes of the Nikkei stock average are computed as 0.0027, 1.4019, 0.0992 and 10.3561, respectively. 5, 10, 25, 50, 75, 90 and 95 percent points from the percent change data are given by −2.2386, −1.5984, −0.6760, 0.0196, 0.6714, 1.5156 and 2.1699. As for the error term included in the measurement equation of Model 1, it is more appropriate to choose a non-Gaussian distribution with fat tails. Thus, the percent change data of the Nikkei stock average are clearly distributed with fat tails, compared with a normal distribution. Furthermore, we can observe in Figure 6.2 that the percent changes up to the end of 1989 seem to be distributed within a relatively small range, but for recent few years the percent changes are more broadly distributed. Thus, the percent change data of the Japanese stock price are heteroscedastic. Therefore, we assume that the error term has a stochastic volatility error for Model 2 and an ARCH(1) error for Model 3. Here, we consider the stochastic variance model for the error, which is specified as follows:

$$\begin{aligned} &\text{(Measurement equation)} \qquad y_t = \alpha_t + \exp\Big(\frac{1}{2}\beta_t\Big)\epsilon_t, \\ &\text{(Transition equation)} \qquad \begin{cases} \alpha_t = \alpha_{t-1} + \eta_t, \\ \beta_t = \delta\beta_{t-1} + \nu_t, \end{cases} \end{aligned}$$

for $t = 1, 2, \cdots, n$, where $\epsilon_t \sim N(0,1)$, $\eta_t \sim N(0, \sigma_\eta^2)$ and $\nu_t \sim N(0, \sigma_\nu^2)$. In Model 2, $\exp(\frac{1}{2}\beta_t)\epsilon_t$ indicates the error term, which is heteroscedastic. $\exp(\frac{1}{2}\beta_t)\epsilon_t$ with $\beta_t = \delta\beta_{t-1} + \nu_t$ is non-Gaussian even if both ϵ_t and ν_t are Gaussian. It is assumed that $\exp(\frac{1}{2}\beta_t)\epsilon_t$ is conditionally Gaussian because ϵ_t is Gaussian. Especially, Watanabe (2000b) has shown that ϵ_t is also non-Gaussian in the case of daily Japanese stock returns. However, for simplicity of discussion we assume in this section that ϵ_t is Gaussian. Thus, $f_y(y_t|\alpha_t, \beta_t)$, $f_\alpha(\alpha_t|\alpha_{t-1})$ and $f_\beta(\beta_t|\beta_{t-1})$ are given by:

$$f_y(y_t|\alpha_t, \beta_t) = \frac{1}{\sqrt{2\pi\exp(\beta_t)}} \exp\Big(-\frac{1}{2\exp(\beta_t)}(y_t - \alpha_t)^2\Big), \tag{6.41}$$

$$f_\alpha(\alpha_t|\alpha_{t-1}) = \frac{1}{\sqrt{2\pi\sigma_\eta^2}} \exp\Big(-\frac{1}{2\sigma_\eta^2}(\alpha_t - \alpha_{t-1})^2\Big), \tag{6.42}$$

$$f_\beta(\beta_t|\beta_{t-1}) = \frac{1}{\sqrt{2\pi\sigma_\nu^2}} \exp\Big(-\frac{1}{2\sigma_\nu^2}(\beta_t - \delta\beta_{t-1})^2\Big), \tag{6.43}$$

In addition, the diffuse prior densities are assumed as follows:

$$\left.\begin{aligned} &f_{\alpha_0}(\alpha_0) \propto \text{ const.}, \quad f_{\beta_0}(\beta_0) \propto \text{ const.}, \quad f_\delta(\delta) \propto \text{ const.}, \\ &f_{\sigma_\eta}(\sigma_\eta^2) \propto \frac{1}{\sigma_\eta^2}, \quad f_{\sigma_\nu}(\sigma_\nu^2) \propto \frac{1}{\sigma_\nu^2}. \end{aligned}\right\} \tag{6.44}$$

Given densities (6.41) – (6.44), the following seven posterior densities are obtained as:

$$f(\alpha_t|A_{t-1}, A_{t+1}^+, B_n, \delta, \sigma_\eta^2, \sigma_v^2) \propto \begin{cases} \frac{1}{\sqrt{2\pi\exp(\beta_t)}}\exp\Big(-\frac{1}{2\exp(\beta_t)}(y_t-\alpha_t)^2\Big) \\ \quad\times\frac{1}{\sqrt{2\pi\sigma_\eta^2}}\exp\Big(-\frac{1}{2\sigma_\eta^2}(\alpha_{t+1}-\alpha_t)^2\Big) \\ \quad\times\frac{1}{\sqrt{2\pi\sigma_\eta^2}}\exp\Big(-\frac{1}{2\sigma_\eta^2}(\alpha_t-\alpha_{t-1})^2\Big), \\ \qquad\qquad \text{for } t=1,2,\cdots,n-1, \\ \frac{1}{\sqrt{2\pi\exp(\beta_t)}}\exp\Big(-\frac{1}{2\exp(\beta_t)}(y_t-\alpha_t)^2\Big) \\ \quad\times\frac{1}{\sqrt{2\pi\sigma_\eta^2}}\exp\Big(-\frac{1}{2\sigma_\eta^2}(\alpha_t-\alpha_{t-1})^2\Big), \\ \qquad\qquad \text{for } t=n, \end{cases}$$

$$f(\beta_t|A_n, B_{t-1}, B_{t+1}^+, \delta, \sigma_\eta^2, \sigma_v^2) \propto \begin{cases} \frac{1}{\sqrt{2\pi\exp(\beta_t)}}\exp\Big(-\frac{1}{2\exp(\beta_t)}(y_t-\alpha_t)^2\Big) \\ \quad\times\frac{1}{\sqrt{2\pi\sigma_v^2}}\exp\Big(-\frac{1}{2\sigma_v^2}(\beta_{t+1}-\delta\beta_t)^2\Big) \\ \quad\times\frac{1}{\sqrt{2\pi\sigma_v^2}}\exp\Big(-\frac{1}{2\sigma_v^2}(\beta_t-\delta\beta_{t-1})^2\Big), \\ \qquad\qquad \text{for } t=1,2,\cdots,n-1, \\ \frac{1}{\sqrt{2\pi\exp(\beta_t)}}\exp\Big(-\frac{1}{2\exp(\beta_t)}(y_t-\alpha_t)^2\Big) \\ \quad\times\frac{1}{\sqrt{2\pi\sigma_v^2}}\exp\Big(-\frac{1}{2\sigma_v^2}(\beta_t-\delta\beta_{t-1})^2\Big), \\ \qquad\qquad \text{for } t=n, \end{cases}$$

$$\alpha_0 \mid A_1^+, B_n, \delta, \sigma_\eta^2, \sigma_v^2 \sim N(\alpha_1, \sigma_\eta^2),$$

$$\beta_0 \mid A_n, B_1^+, \delta, \sigma_\eta^2, \sigma_v^2 \sim N\Big(\frac{\beta_1}{\delta}, \frac{\sigma_v^2}{\delta^2}\Big),$$

$$\delta \mid A_n, B_n, \sigma_\eta^2, \sigma_v^2 \sim N\Big(\frac{\sum_{t=1}^n \beta_t\beta_{t-1}}{\sum_{t=1}^n \beta_{t-1}^2}, \frac{\sigma_v^2}{\sum_{t=1}^n \beta_{t-1}^2}\Big),$$

$$\frac{1}{\sigma_\eta^2} \mid A_n, B_n, \delta, \sigma_v^2 \sim G\Big(\frac{n}{2}, \frac{2}{\sum_{t=1}^n(\alpha_t-\alpha_{t-1})^2}\Big),$$

$$\frac{1}{\sigma_v^2} \mid A_n, B_n, \delta, \sigma_\eta^2 \sim G\Big(\frac{n}{2}, \frac{2}{\sum_{t=1}^n(\beta_t-\delta\beta_{t-1})^2}\Big),$$

where $B_t = \{\beta_0, \beta_1, \cdots, \beta_t\}$ and $B_{t+1}^+ = \{\beta_{t+1}, \beta_{t+2}, \cdots, \beta_n\}$.

Model 3 (Time Varying Mean with ARCH(1) Error): As the third model, the percent change of the Nikkei stock average is taken as the time-dependent mean and the first-order autoregressive heteroscedasticity noise (i.e., ARCH(1) noise), which is given by the following model:

$$\begin{aligned} &\text{(Measurement equation)} && y_t = \alpha_t + \epsilon_t, \\ &\text{(Transition equation)} && \alpha_t = \alpha_{t-1} + \eta_t, \end{aligned}$$

for $t = 1, 2, \cdots, n$, where $\epsilon_t \sim N(0, \sigma_t^2)$ for $\sigma_t^2 = \sigma_\epsilon^2(1 + \delta\epsilon_{t-1}^2) = \sigma_\epsilon^2(1 + \delta(y_{t-1} - \alpha_{t-1})^2)$, $0 \le \sigma_\epsilon^2\delta < 1$ and $\eta_t \sim N(0, \sigma_\eta^2)$. In this model, $\sigma_\epsilon^2\delta$ is known as an ARCH effect.

Under the above state space model, the conditional distribution of y_t given y_{t-1}, α_t and α_{t-1}, $f_y(y_t|y_{t-1}, \alpha_t, \alpha_{t-1})$, and that of α_t given α_{t-1}, $f_\alpha(\alpha_t|\alpha_{t-1})$, are derived from the measurement and transition equations, respectively, which distributions are represented as:

$$\begin{aligned} f_y(y_t|y_{t-1}, \alpha_t, \alpha_{t-1}) = &\frac{1}{\sqrt{2\sigma_\epsilon^2(1 + \delta(y_{t-1} - \alpha_{t-1})^2)}}, \\ &\times \exp\Big(-\frac{(y_t - \alpha_t)^2}{2\pi\sigma_\epsilon^2(1 + \delta(y_{t-1} - \alpha_{t-1})^2)}\Big), \end{aligned} \tag{6.45}$$

$$f_\alpha(\alpha_t|\alpha_{t-1}) = \frac{1}{\sqrt{2\pi\sigma_\eta^2}} \exp\Big(-\frac{1}{2\sigma_\eta^2}(\alpha_t - \alpha_{t-1})^2\Big). \tag{6.46}$$

We take $\sigma_1^2 = \sigma_\epsilon^2(1 + \delta\alpha_0^2)$, i.e., $y_0 = 0$, for the initial variance. There are four unknown parameters in the model, i.e., α_0, σ_ϵ^2, δ and σ_η^2. We consider estimating the four parameters by the Bayesian approach. As for the prior densities, all the parameters are assumed to be the flat priors, which implies that all the real values can be uniformly chosen. Especially, since variance does not take non-positive values, the log of variance is assumed to be uniform. Thus, the prior distributions of α_0, σ_ϵ^2, δ and σ_η^2 are assumed to be as follows:

$$\left.\begin{aligned} &f_{\alpha_0}(\alpha_0) \propto \text{ const.}, \quad f_{\sigma_\epsilon}(\sigma_\epsilon^2) \propto \frac{1}{\sigma_\epsilon^2}, \quad f_\delta(\delta) \propto \text{ const.}, \\ &f_{\sigma_\eta}(\sigma_\eta^2) \propto \frac{1}{\sigma_\eta^2}, \end{aligned}\right\}, \tag{6.47}$$

where $0 \le \delta < 1/\sigma_\epsilon^2$. Using densities (6.45), (6.46) and (6.47), the posterior densities are related to the following conditional densities:

$$f(\alpha_t|A_{t-1}, A_{t+1}^+, \sigma_\epsilon^2, \delta, \sigma_\eta^2)$$

$$
\propto \begin{cases}
\dfrac{1}{\sqrt{2\pi\sigma_\epsilon^2(1+\delta(y_{t-1}-\alpha_{t-1})^2)}}\exp\Big(-\dfrac{(y_t-\alpha_t)^2}{2\sigma_\epsilon^2(1+\delta(y_{t-1}-\alpha_{t-1})^2)}\Big) \\
\quad\times\dfrac{1}{\sqrt{2\pi\sigma_\epsilon^2(1+\delta(y_t-\alpha_t)^2)}}\exp\Big(-\dfrac{(y_{t+1}-\alpha_{t+1})^2}{2\sigma_\epsilon^2(1+\delta(y_t-\alpha_t)^2)}\Big) \\
\quad\times\dfrac{1}{\sqrt{2\pi\sigma_\eta^2}}\exp\Big(-\dfrac{1}{2\sigma_\eta^2}(\alpha_{t+1}-\alpha_t)^2\Big) \\
\quad\times\dfrac{1}{\sqrt{2\pi\sigma_\eta^2}}\exp\Big(-\dfrac{1}{2\sigma_\eta^2}(\alpha_t-\alpha_{t-1})^2\Big), \\
\qquad\qquad \text{for } t=1,2,\cdots,n-1, \\
\dfrac{1}{\sqrt{2\pi\sigma_\epsilon^2(1+\delta(y_{t-1}-\alpha_{t-1})^2)}}\exp\Big(-\dfrac{(y_t-\alpha_t)^2}{2\sigma_\epsilon^2(1+\delta(y_{t-1}-\alpha_{t-1})^2)}\Big) \\
\quad\times\dfrac{1}{\sqrt{2\pi\sigma_\eta^2}}\exp\Big(-\dfrac{1}{2\sigma_\eta^2}(\alpha_t-\alpha_{t-1})^2\Big), \\
\qquad\qquad \text{for } t=n,
\end{cases}
$$

$$
\alpha_0 \mid A_1^+, \sigma_\epsilon^2, \sigma_\eta^2 \sim N(\alpha_1, \sigma_\eta^2),
$$

$$
\frac{1}{\sigma_\epsilon^2} \mid A_n, \delta, \sigma_\eta^2 \sim G\Big(\frac{n}{2}, \frac{2(1+\delta(y_{t-1}-\alpha_{t-1})^2)}{\sum_{t=1}^n (y_t-\alpha_t)^2}\Big),
$$

$$
f(\delta|A_n, \sigma_\epsilon^2, \sigma_\eta^2) \propto \prod_{t=1}^n \frac{1}{\sqrt{1+\delta(y_{t-1}-\alpha_{t-1})^2}}\exp\Big(-\frac{(y_t-\alpha_t)^2}{2\sigma_\epsilon^2(1+\delta(y_{t-1}-\alpha_{t-1})^2)}\Big),
$$

$$
\frac{1}{\sigma_\eta^2} \mid A_n, \sigma_\epsilon^2 \sim G\Big(\frac{n}{2}, \frac{2}{\sum_{t=1}^n (\alpha_t-\alpha_{t-1})^2}\Big).
$$

The Gibbs sampler can be performed to generate random draws from the posterior density, utilizing the above conditional densities.

Note that it is easy to extend the model above to the ARCH(p) error. That is, for $\epsilon_t \sim N(0, \sigma_t^2)$, it is also possible to assume that σ_t^2 is given by: $\sigma_t^2 = \sigma_\epsilon^2(1 + \delta_1\epsilon_{t-1}^2 + \delta_2\epsilon_{t-2}^2 + \cdots + \delta_p\epsilon_{t-p}^2)$, where $\epsilon_{t-i} = y_{t-i} - \alpha_{t-i}$, $i = 1, 2, \cdots, p$, in this case. Therefore, equation (6.45) is given by $f_y(y_t|y_{t-1}, y_{t-2}, \cdots, y_{t-p}, \alpha_t, \alpha_{t-1}, \alpha_{t-p})$, not $f_y(y_t|y_{t-1}, \alpha_t, \alpha_{t-1})$, and $f(\alpha_t|A_{t-1}, A_{t+1}^+, \sigma_\epsilon^2, \delta_1, \delta_2, \cdots, \delta_p, \sigma_\eta^2)$ for $t = 1, 2, \cdots, n-p$ is modified as follows:

$$
\begin{aligned}
&f(\alpha_t|A_{t-1}, A_{t+1}^+, \sigma_\epsilon^2, \delta_1, \delta_2, \cdots, \delta_p, \sigma_\eta^2) \\
&\quad\propto \prod_{i=0}^p \frac{1}{\sqrt{2\pi\sigma_{t+i}^2}}\exp\Big(-\frac{(y_{t+i}-\alpha_{t+i})^2}{2\sigma_{t+i}^2}\Big) \\
&\qquad\times\frac{1}{\sqrt{2\pi\sigma_\eta^2}}\exp\Big(-\frac{1}{2\sigma_\eta^2}(\alpha_{t+1}-\alpha_t)^2\Big)\frac{1}{\sqrt{2\pi\sigma_\eta^2}}\exp\Big(-\frac{1}{2\sigma_\eta^2}(\alpha_t-\alpha_{t-1})^2\Big), \\
&\qquad\qquad\qquad \text{for } t=1,2,\cdots,n-p,
\end{aligned}
$$

which corresponds to $f(\alpha_t|A_{t-1}, A_{t+1}^+, \sigma_\epsilon^2, \delta, \sigma_\eta^2)$, shown above.

Table 6.10: Basic Statistics in Models 1 – 3

	Model 1		Model 2			Model 3		
	σ_ϵ^2	σ_η^2	σ_η^2	δ	σ_v^2	σ_ϵ^2	δ	σ_η^2
AVE	1.964	.00001566	.0000296	.9792	.04546	1.335	.2770	.00000800
SER	0.041	.00001497	.0000205	.0041	.00671	.0391	.0276	.00000768
Skewness	0.080	2.359	2.005	−.2632	.42540	.0837	.2375	2.865
Kurtosis	3.012	9.519	8.184	3.133	3.3026	3.028	3.134	12.64
0.010	1.871	.00000225	.0000083	.9690	.03190	1.246	.2173	.00000169
0.025	1.885	.00000265	.0000094	.9708	.03370	1.260	.2262	.00000188
0.050	1.897	.00000319	.0000105	.9723	.03530	1.272	.2336	.00000210
0.100	1.912	.00000443	.0000119	.9739	.03724	1.285	.2427	.00000242
0.250	1.936	.00000651	.0000157	.9766	.04075	1.308	.2580	.00000355
0.500	1.964	.00001023	.0000230	.9794	.04496	1.335	.2761	.00000585
0.750	1.991	.00001867	.0000362	.9821	.04966	1.361	.2951	.00000888
0.900	2.017	.00003493	.0000567	.9843	.05430	1.385	.3131	.00001499
0.950	2.033	.00004729	.0000726	.9856	.05727	1.400	.3243	.00002579
0.975	2.046	.00006077	.0000864	.9867	.05992	1.413	.3345	.00003414
0.990	2.062	.00007810	.0001033	.9880	.06317	1.429	.3471	.00004013
$\mathrm{E}\big(\log f_y(Y_n\|A_n)\|Y_n\big)$	−8044.04		−6871.67			−7838.93		

As for $t = n-p+1, n-p+2, \cdots, n$, p in $f(\alpha_t|A_{t-1}, A_{t+1}^+, \sigma_\epsilon^2, \delta_1, \delta_2, \cdots, \delta_p, \sigma_\eta^2)$ should be replaced by $p-1$ when $t = n-p+1$, $p-2$ when $t = n-p+2, \cdots, 0$ when $t = n$. Remember that σ_{t+i}^2, $i = 1, 2, \cdots, p$, depend on α_t because σ_t^2 is a function of α_{t-1}, α_{t-2}, $\cdots$, α_{t-p}. In addition to the conditional density of α_t, the conditional densities of σ_ϵ^2 and $\{\delta_i\}_{i=1}^p$ also have to be modified. Thus, we can extend Model 3 to the ARCH(p) cases. In this section, however, for simplicity we take the ARCH(1) model.

6.6.3 Results and Discussion

In this section, $M = 10^5$ and $N = 10^6$ are taken. Remember that $M + N$ random draws are generated and that the first M random draws are discarded from further consideration, taking into account of the convergence property of the Gibbs sampler. We consider that $M = 10^5$ might be large enough for convergence.

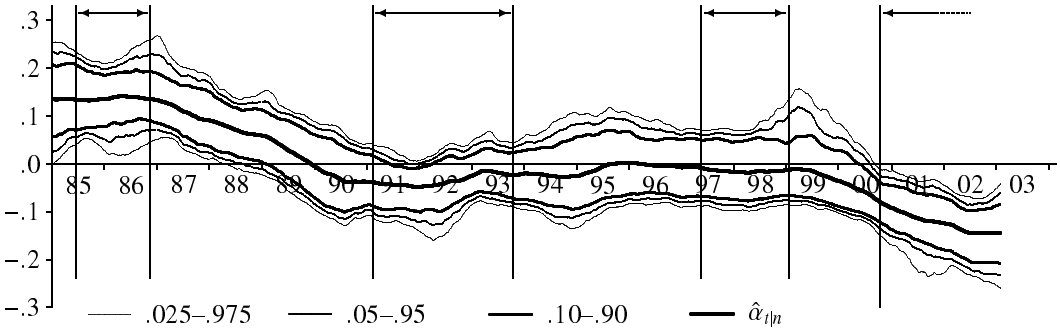
The results are in Table 6.10 for the parameter estimates, Figure 6.4 for the estimated state means and Figure 6.5 for the volatilities estimated in Models 2 and 3. In Table 6.10, AVE, SER, Skewness and Kurtosis represents the arithmetic average, standard error, skewness and kurtosis obtained from $N = 10^6$ random draws, which are generated from each posterior density. 0.010, 0.025, 0.050, 0.100, 0.250, 0.500, 0.750, 0.900, 0.950, 0.975 and 0.990 denote 1%, 2.5%, 5%, 10%, 25%, 50%, 75%, 90%, 95%, 97.5%, 99% points from $N = 10^6$ random draws. E(log $f_y(Y_n|A_n)|Y_n$) indicates the expectated density function of Y_n given A_n, i.e., $\frac{1}{N}\sum_{i=1}^N \log f_y(Y_n|A_{i,n})$, which is not an evaluation of the likelihood function (6.37) but a rough measure of the likelihood function.

In Table 6.10, Model 1 is estimated under normality assumption for the error terms ϵ_t and η_t. However, from Figure 6.2, the percent change data y_t likely has heavier tails than a normal random variable and also seem to be heteroscedastic over time t. Therefore, We consider the stochastic volatility model in Model 2 and the ARCH(1) model in Model 3. Both the stochastic volatility effect and the ARCH effect are significantly grater than zero (see AVE, SER and the percent points of δ in Models 2 and 3). Especially, δ in Model 2 is very close to one and accordingly the volatility effect continues persistently from time $t-1$ to time t. Moreover, the ARCH(1) effect in Model 3 is given by $\sigma_\epsilon^2\delta = 1.335 \times 0.2770 = 0.3698$ and it is statistically different from zero, which implies that the volatility changes over time. Comparing Models 1 – 3, Model 2 might be the best model, because Model 2 gives us the largest estimate of $\mathrm{E}(\log f_y(Y_n|A_n)|Y_n)$, i.e., -8044.04 for Model 1, -6871.67 for Model 2 and -7838.93 for Model 3.

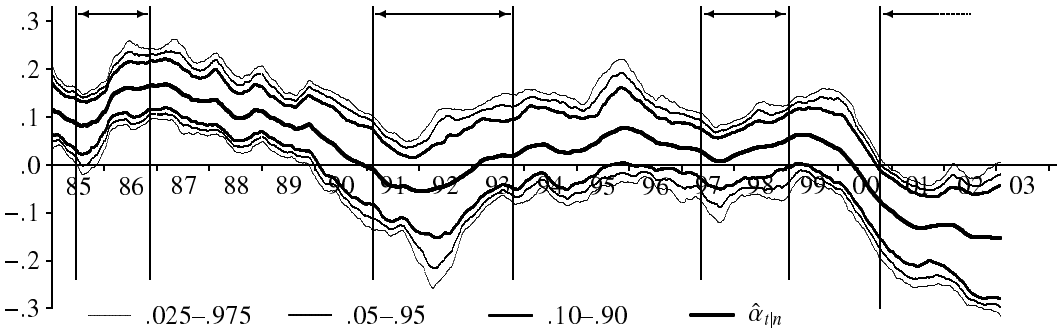
In Figure 6.3, the trend component included in the original percent change data is displayed for Models 1 – 3, where .025–.975, .05–.95 and .10–.90 denote the interval between 2.5 and 97.5 percent points, that between 5 and 95 percent points and that between 10 and 90 percent points, respectively. Thus, .025–.975, .05–.95 and .10–.90 represent 95%, 90% and 80% confidence intervals. The 2.5%, 5%, 10%, 90%, 95% and 97.5% points are obtained from the 10^6 random draws which are sorted in order of size. Moreover, $\hat{\alpha}_{t|n}$ indicates the estimate of $\alpha_{t|n} = \mathrm{E}(\alpha_t|Y_n)$, i.e., the fixed-interval smoothing estimate. Note that the vertical axis in Figure 6.3 is much smaller than that in Figure 6.2 with respect to scale. This implies that the trend component is very small for all the three models and accordingly the noise component plays an important role in the percent changes of Nikkei stock average. In addition to the movement of the trend term, the period from peak to bottom in a business cycle, i.e., the recession period, is represented by $\longleftrightarrow$ in (a) – (c) of Figure 6.3. The Economic and Social Research Institute, Cabinet Office, Government of Japan reports the peak and bottom of the business cycle in Japan. During the period from January 4, 1985 to January 31, 2003, the peaks are June in 1985, February in 1991, May in 1997 and October in 2000, and the bottoms are November in 1986, October in 1993 and January in 1999. See Figure 6.4 for the diffusion and composit indices in Japan, which are useful to see business cycles. Thus, in Figure 6.3, $\longleftrightarrow$ indicates the period from peak to bottom in a business cycle. All the models seem to have a similar movement, but Model 2 has larger fluctuation in the percent change than Models 1 and 3. The confidence intervals of Model 2 are larger than those of Models 1 and 3 with respect to the width of each interval. By the confidence intervals, it is also easy to know the movement of the distribution. For example, in Model 2, the distribution of the trend component around 1992 is skewed to the left, i.e., the negative tail is fatter than the positive one, and the distribution around 1995 is skewed to the right. At the same time, using the confidence intervals, we can test whether the trend component is statistically equal to zero. For example, we can see that the trend component in Model 2 is significantly greater than zero during the period from 1986 to 1989 and it is negative during the period 2001 to 2002. Although the confidence intervals of Model 2 are longer than

Figure 6.3: Movement of Trend Component α_t (%)

(a) Model 1



(b) Model 2



(c) Model 3

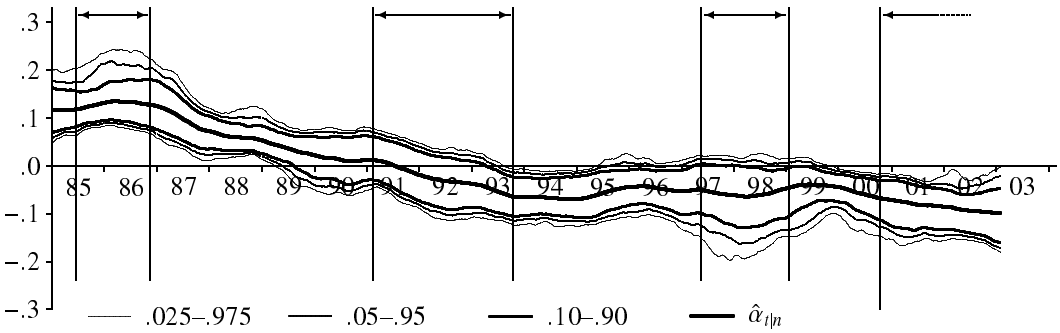


Figure 6.4: Diffusion and Composit Indices

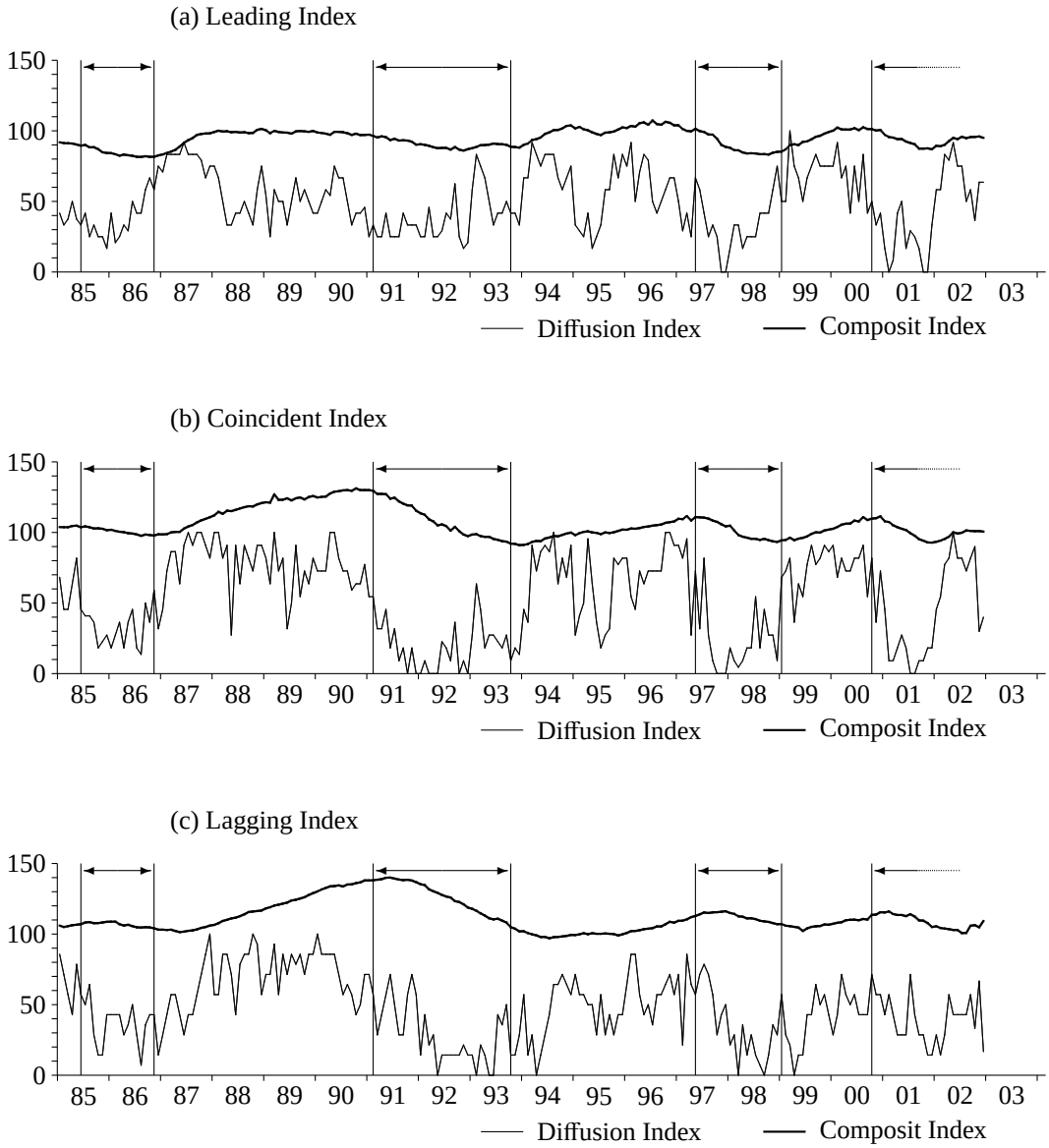
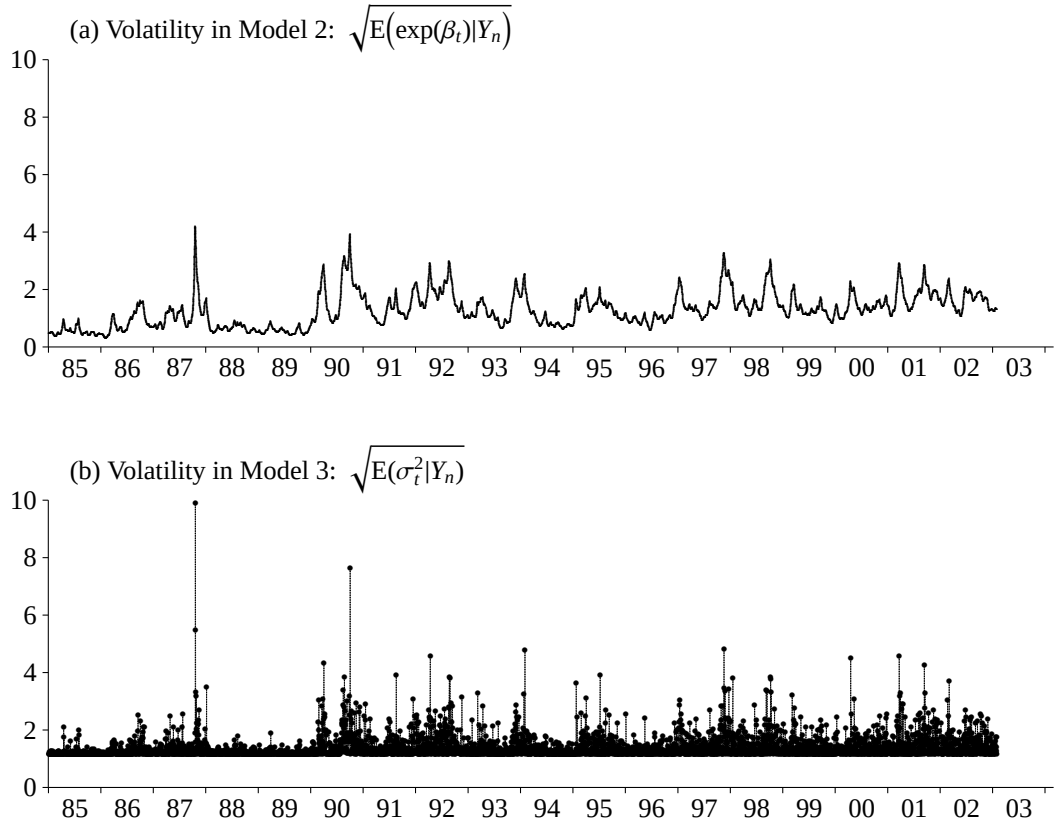


Figure 6.5: Volatility in Models 2 and 3



those of Models 1 and 3, Model 2 might be more plausible, because each recession period has a trough in the trend component of Model 2. In addition, we can observe that the estimated trend component in Model 2 has a similar tendency to the Nikkei stock average in Figure 6.1, but the Nikkei stock average is one or two years behind the estimated trend component in Model 2.

Furthermore, to see the relationship between the trend term and the business cycle, in Figure 6.4 the diffusion and composit indices, which are the monthly data, are displayed, where $\longleftrightarrow$ is the period from peak to bottom in a business cycle, (a), (b) and (c) show the leading index, the coincident index and the lagging index, respectively. In Figures 6.4 (a) – (c), the diffusion indices are represented by % and the composit indices are normalized to be 100 in 1995. The trend component in Model 2 shows a similar movement to the composit index in the leading index (Figure 6.4 (a)), rather than the coincident and lagging indecies. In this sense, we can conclude that the trend component in the percent change of the Nikkei stock average forecasts the business cycle.

Next, we focus on the volatility, rather than the trend component, for Models 2 and 3. The volatility is given by $\exp(\beta_t)$ for Model 2 and σ_t^2 for Model 3, which represents

a fluctuation in the stock price. When there is a sharp fluctuation in the stock price, we sometimes have a big gain but sometimes a big loss. Thus, the volatility implies the risk in a sense. We assume that the error is homoscedastic over time in Model 1 but heteroscedastic in Models 2 and 3. Clearly, the percent changes are heteroscedastic from Figure 6.2, where the percent changes of the Nikkei stock average are displayed. In Figure 6.5, the movements in the volatilities are shown. We had Black Monday on October 19, 1987, when the stock price drastically fell down in New York stock market. On Black Monday, the volatility excremely increases for both models. Since 1990, the Japanese stock price drastically starts to decrease, which represents the bubble phenomenon. However, the volatility increases after the bubble burst. Thus, it is shown in Figure 6.5 that we have larger volatility after the bubble burst in 1990 than before that. Note that the volatility in Model 3 is not smooth, compared with Model 2. The error term in Model 3 is assumed to be the ARCH(1) process in this section. When the ARCH(p) process is assumed for the error term, we can obtain smooth volatility.

6.6.4 Further Models and Discussion

In Section 6.6.3, it is concluded that Model 2 is the best model in the criterion of $\mathrm{E}(\log f_y(Y_n|A_n)|Y_n)$. In this section, Model 2 is modified to obtain more realistic model. There are two modifications; (i) a sign effect of the percent change in the previous day and (ii) a holiday effect. (i) is given by Model 2a, (ii) corresponds to Model 2b, and (i) and (ii) are combined in Model c.

Model 2a (Time Varying Mean with Stochastic Volatility Error and Non-Symmetry Effect): In Model 2, we have assumed that the error term in the measurement equation, i.e., ϵ_t, is symmetric. However, it is known that the negative percent change becomes more volatile in the next day than the positive percent change, i.e., there is a negative relation between expected return and volatility (see Glosten, Jagannathan and Runkle (1993)). Therefore, in Model 2a we examine how the sign of y_{t-1} affects the volatility, i.e., we examine whether there is a **non-symmetry effect** in the volatility. Taking into account difference between negative percent change and positive one in the previous day, Model 2a is specified as follows:

$$\begin{aligned} &\text{(Measurement equation)} \qquad y_t = \alpha_t + \exp\Big(\frac{1}{2}\beta_t\Big)\epsilon_t, \\ &\text{(Transition equation)} \qquad \begin{cases} \alpha_t = \alpha_{t-1} + \eta_t, \\ \beta_t = \delta_1 d_t^- + \delta_2 d_t^+ + \delta_3 \beta_{t-1} + v_t, \end{cases} \end{aligned}$$

for $t = 1, 2, \cdots, n$, where $\epsilon_t \sim N(0,1)$, $\eta_t \sim N(0, \sigma_\eta^2)$ and $v_t \sim N(0, \sigma_v^2)$. d_t^- and d_t^+ represent the dummy variables, where $d_t^- = 1$ if $y_{t-1} < 0$ and $d_t^- = 0$ otherwise, and $d_t^+ = 1$ if $y_{t-1} > 0$ and $d_t^+ = 0$ otherwise.

Model 2b (Time Varying Mean with Stochastic Volatility Error and Holiday Effect): In Models 1 – 3, we have not considered how long the stock market is closed between time t and time $t-1$. When we have much information into the market, the stock price becomes volatile. The volatility depends on the amount of information, which is roughly equivalent to the number of nontrading days between trading $t-1$ and t. See Nelson (1991) for the holiday effect. Therefore, using Model 2b, we examine whether there is a **holiday effect** in the Japanese stock market. The model is specified as follows:

$$\begin{aligned} &\text{(Measurement equation)} \qquad y_t = \alpha_t + \exp\Big(\frac{1}{2}\beta_t\Big)\epsilon_t, \\ &\text{(Transition equation)} \qquad \begin{cases} \alpha_t = \alpha_{t-1} + \eta_t, \\ \beta_t = \delta_1 DT_t + \delta_2 \beta_{t-1} + \nu_t, \end{cases} \end{aligned}$$

for $t = 1, 2, \cdots, n$, where $\epsilon_t \sim N(0,1)$, $\eta_t \sim N(0, \sigma_\eta^2)$ and $\nu_t \sim N(0, \sigma_\nu^2)$. DT_t indicates the number of nontrading days between time t and time $t-1$. That is, when time $t-1$ is Friday we usually have $DT_t = 2$, because the market is closed for two days (Saturday and Sunday). If Monday is a national holiday, we obtain $DT_t = 3$, where time $t-1$ is Friday and time t is Tuesday. Thus, DT_t represents the holiday effect, which is a proxy variable of the trading volume or the amount of information.

Model 2c (Time Varying Mean with Stochastic Volatility Error and Both Non-Symmetry and Holiday Effects): In Model 2c, Models 2a and 2b are combined. The model is given by:

$$\begin{aligned} &\text{(Measurement equation)} \qquad y_t = \alpha_t + \exp\Big(\frac{1}{2}\beta_t\Big)\epsilon_t, \\ &\text{(Transition equation)} \qquad \begin{cases} \alpha_t = \alpha_{t-1} + \eta_t, \\ \beta_t = \delta_1 d_t^- + \delta_2 d_t^+ + \delta_3 DT_t + \delta_4 \beta_{t-1} + \nu_t, \end{cases} \end{aligned}$$

for $t = 1, 2, \cdots, n$, where $\epsilon_t \sim N(0,1)$, $\eta_t \sim N(0, \sigma_\eta^2)$ and $\nu_t \sim N(0, \sigma_\nu^2)$.

Results and Discussion: The results are in Table 6.11 and Figures 6.6 and 6.7. In Model 2a, it is examined whether a positive or negative sign of the previous percent change affects the volatility of the present percent change. It is known that in the stock market a negative shock yesterday yields high volatility today but a positive shock yesterday does not. See Nelson (1991) and Glosten, Jagannathan and Runkle (1993) for non-symmetry of the volatility. That is, d_t^- should have a positive effect to the volatility of y_t and d_t^+ should have a negative effect to the volatility of y_t. From Table 6.11, both δ_1 and δ_2 in Model 2a are significantly equal to non-zero. We obtain the results that δ_1 should be statistically positive and δ_2 should be negative, which are consistent with the past research, e.g., Nelson (1991) and Glosten, Jagannathan and Runkle (1993).

Table 6.11: Basic Statistics in Models 2a – 2c

	Model 2a				
	σ_η^2	δ_1	δ_2	δ_3	σ_v^2
AVE	.0000354	.1133	−.0996	.9706	.04372
SER	.0000364	.0137	.0128	.0043	.00648
Skewness	1.511	.0687	−.0600	−.2445	.4407
Kurtosis	5.225	3.019	3.014	3.121	3.376
0.010	.0000028	.0821	−.1299	.9597	.03071
0.025	.0000031	.0868	−.1250	.9616	.03234
0.050	.0000035	.0910	−.1208	.9632	.03388
0.100	.0000041	.0959	−.1160	.9650	.03577
0.250	.0000069	.1041	−.1081	.9678	.03919
0.500	.0000256	.1132	−.0995	.9708	.04324
0.750	.0000495	.1224	−.0910	.9736	.04777
0.900	.0000902	.1309	−.0833	.9760	.05221
0.950	.0001109	.1361	−.0787	.9774	.05510
0.975	.0001314	.1407	−.0748	.9786	.05772
0.990	.0001553	.1459	−.0705	.9800	.06083
$\mathrm{E}\left(\log f_y(Y_n\|A_n)\|Y_n\right)$	−6857.36				

	Model 2b			
	σ_η^2	δ_1	δ_2	σ_v^2
AVE	.00001740	.02299	.9754	.04879
SER	.00000990	.00762	.0045	.00735
Skewness	1.413	.1451	−.2782	.4276
Kurtosis	5.713	3.103	3.166	3.276
0.010	.00000483	.00595	.9640	.03395
0.025	.00000554	.00853	.9660	.03585
0.050	.00000627	.01077	.9677	.03761
0.100	.00000725	.01339	.9695	.03976
0.250	.00000993	.01780	.9725	.04363
0.500	.00001519	.02281	.9756	.04829
0.750	.00002223	.02799	.9785	.05334
0.900	.00003024	.03283	.9810	.05848
0.950	.00003734	.03583	.9824	.06179
0.975	.00004322	.03847	.9836	.06475
0.990	.00004962	.04161	.9850	.06828
$\mathrm{E}\left(\log f_y(Y_n\|A_n)\|Y_n\right)$	−6865.08			

Table 6.11: Basic Statistics in Models 2a – 2c —< Continued >—

	Model 2c					
	σ_η^2	δ_1	δ_2	δ_3	δ_4	σ_v^2
AVE	.00001814	.05949	−.1432	.1121	.9630	.05081
SER	.00001409	.01694	.0165	.0223	.0050	.00727
Skewness	1.852	.0111	−.0704	.0373	−.2205	.3569
Kurtosis	6.821	3.039	2.970	2.969	3.077	3.219
0.010	.00000414	.02018	−.1823	.0613	.9507	.03573
0.025	.00000510	.02621	−.1761	.0691	.9528	.03772
0.050	.00000578	.03151	−.1707	.0758	.9546	.03958
0.100	.00000669	.03773	−.1645	.0836	.9566	.04182
0.250	.00000852	.04811	−.1542	.0969	.9598	.04576
0.500	.00001275	.05956	−.1430	.1120	.9632	.05040
0.750	.00002282	.07083	−.1319	.1272	.9665	.05542
0.900	.00003770	.08106	−.1221	.1408	.9693	.06032
0.950	.00004882	.08725	−.1163	.1489	.9709	.06345
0.975	.00005819	.09272	−.1114	.1559	.9723	.06626
0.990	.00006672	.09933	−.1057	.1641	.9738	.06966
$\mathrm{E}\left(\log f_y(Y_n\|A_n)\|Y_n\right)$	−6836.01					

In Model 2b, we discuss whether the number of nontrading days affects the volatility. Nelson (1991) and Watanabe (2000a) have shown that there is a positive correlation between return volatility and trading volume. When the number of nontrading days between trading days t and $t-1$ increases, it might be plausible to consider that the trading volume on trading day t also increases. Thus, in Model 2b we consider that the length of the period between trading days t and $t-1$ might be equivalent to the trading volume. Therefore, the number of nontrading days between trading days t and $t-1$ (i.e., the number of holidays between t and $t-1$, denoted by DT_t) should be positively correlated with the volatility of y_t. As a result, the estimate of δ_1 is 0.02299 in Model 2b, which is positive, and accordingly δ_1 is significantly positive, judging from the percent points, i.e., 0.00595 for 0.010 (1% point) and 0.04161 for 0.990 (99% point). Therefore, it is shown from Table 6.11 that the number of nontrading days depends on the volatility.

In Model 2c, Model 2a and Model 2b are combined. That is, both the non-symmetry and holiday effects are simultaneously introduced into Model 2. δ_1 and δ_2 represent the non-symmetry effect in the volatility, while δ_3 indicates the holiday effect. It is shown from Model 2c in Table 6.11 that δ_1, δ_2 and δ_3 are not equal to zero. The estimate of δ_1 is positive, that of δ_2 is negative and that of δ_3 is greater than zero. Therefore, we can conclude that δ_1 is positive, δ_2 is negative and δ_3 is positive, which results are consistent with Models 2a and 2c. That is, in Japanese stock market, negative returns at time $t-1$ yield high volatility at time t, positive returns at time $t-1$ imply low volatility at time t, and the number of nontrading days between trading days t and $t-1$ raises volatility at time t.

Comparing $\mathrm{E}(\log f_y(Y_n|A_n)|Y_n)$ for Models 2 and 2a – 2c, we obtain −6871.67 in

Model 2, −6857.36 in Model 2a, −6865.08 in Model 2b, and −6836.01 in Model 2c. Thus, according to the criterion of $E(\log f_y(Y_n|A_n)|Y_n)$, it might be concluded that Model 2c is the best model, where volatility depend on both non-symmetry and holiday effects.

In Figure 6.6, the movement of trend component α_t is displayed for the three models, i.e., Models 2a – 2c, where the interval between 2.5 and 97.5 percent points, that between 5 and 95 percent points and that between 10 and 90 percent points are shown together with the smoothing estimate $\hat{\alpha}_{t|n}$. We can observe from Figure 6.6 that the movement of trend component is quite different for each model. However, the trend term is statistically positive during the period from 1985 to 1989 and negative after 2000. The period from 1985 to 1989 corresponds to the bubble period, while the period after 2000 implies the recent serious recession in Japan.

In Figure 6.7, the volatility is shown for each model. The volatility in Model 2 (Figure 6.5 (a)), that in Model 2a (Figure 6.7 (a)), that in Model 2b (Figure 6.7 (b)) and that in Model 2c (Figure 6.7 (c)) are very close to each other. We cannot distinguish them at a glance. However, in Model 2a the number of jagged lines increases, compared with Model 2. Similarly, Model 2c has more jagged lines than Models 2, 2a and 2b. Thus, Model 2c can catch a lot of small movements by including extra effects such as non-symmetry and holiday effects into the volatility.

6.6.5 Concluding Remarks

In Section 6.6, we have shown an application of nonlinear non-Gaussian state space modeling, where daily data of the Nikkei stock average are taken. We assume in this section that the percent change of the Nikkei stock average consists of the trend and irregular components.

First, we have discussed three models, i.e., Models 1 – 3. Model 1 is the simplest model, where the irregular component is Gaussian. Models 2 and 3 have assumed that the irregular component has stochastic variance. The stochastic volatility model is adopted for the irregular component in Model 2, while the irregular component in Model 3 is assumed to be the ARCH(1) error. As a result, we have obtained the result that Model 2 is more plausible than Models 1 and 3 from the $E(\log f_y(Y_n|A_n)|Y_n)$ criterion, which is a rough measure of the likelihood function.

Next, using Model 2, we have examined whether the non-symmetry effect on the sign of y_{t-1} (Model 2a), the holiday effect (Model 2b) or both effects (Model 2c) affect volatility. It is known that a negative return at time $t-1$ yields a high volatility at time t but a positive return gives us a low volatility at time t. Moreover, as discussed in Nelson (1991), the volatility at time t increases as the number of nontrading days between trading days t and $t-1$ is large. In this section, we also have obtained these results using the Japanese stock price data (i.e., Nikkei stock average). From the $E(\log f_y(Y_n|A_n)|Y_n)$ criterion, we have concluded that the best model is given by Model 2c, where both effects are simultaneously taken into account. Thus, the holiday effect as well as the non-sysmmetry effect are important for volatility.

Figure 6.6: Movement of Trend Component α_t (%): Models 2a – 2c

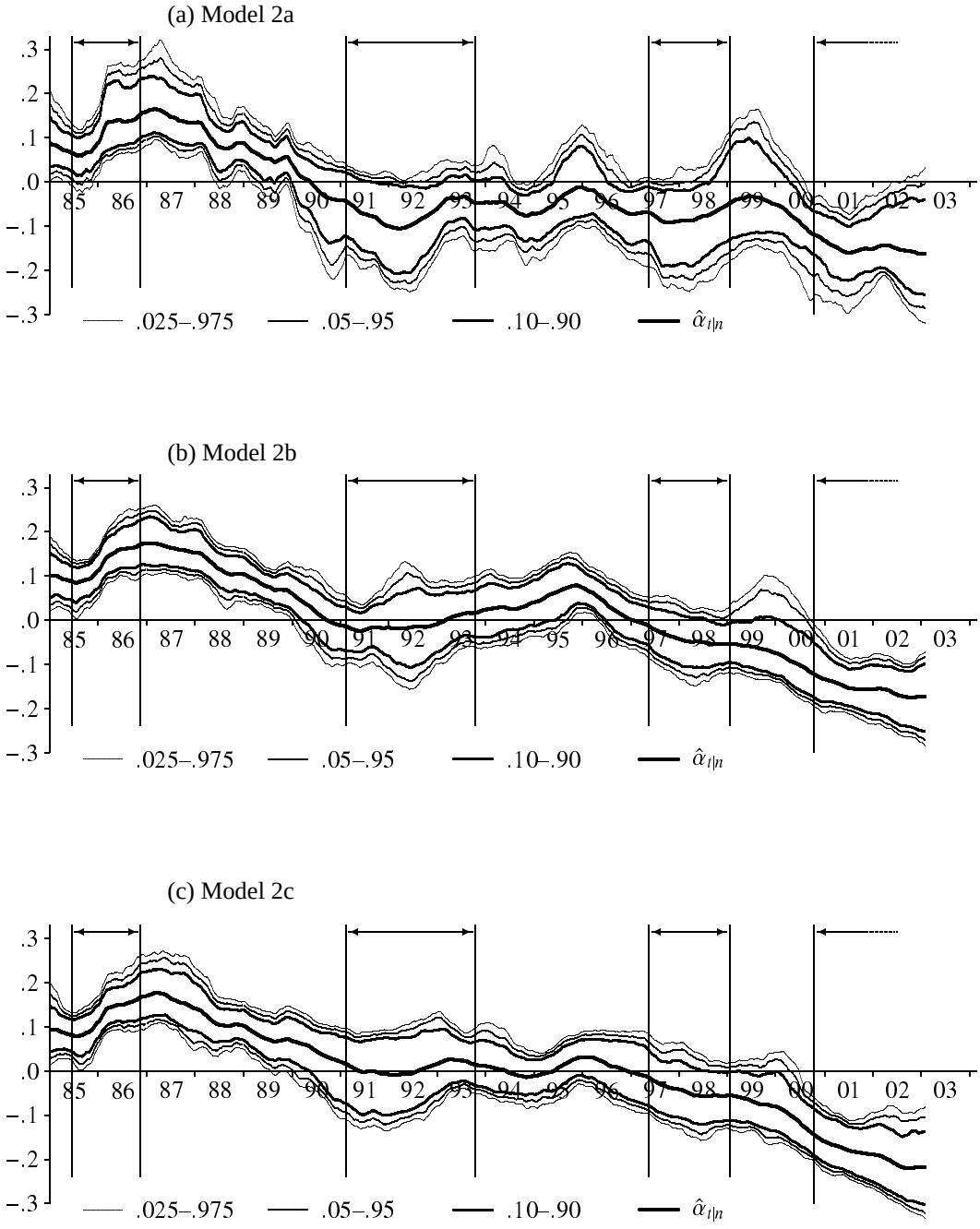
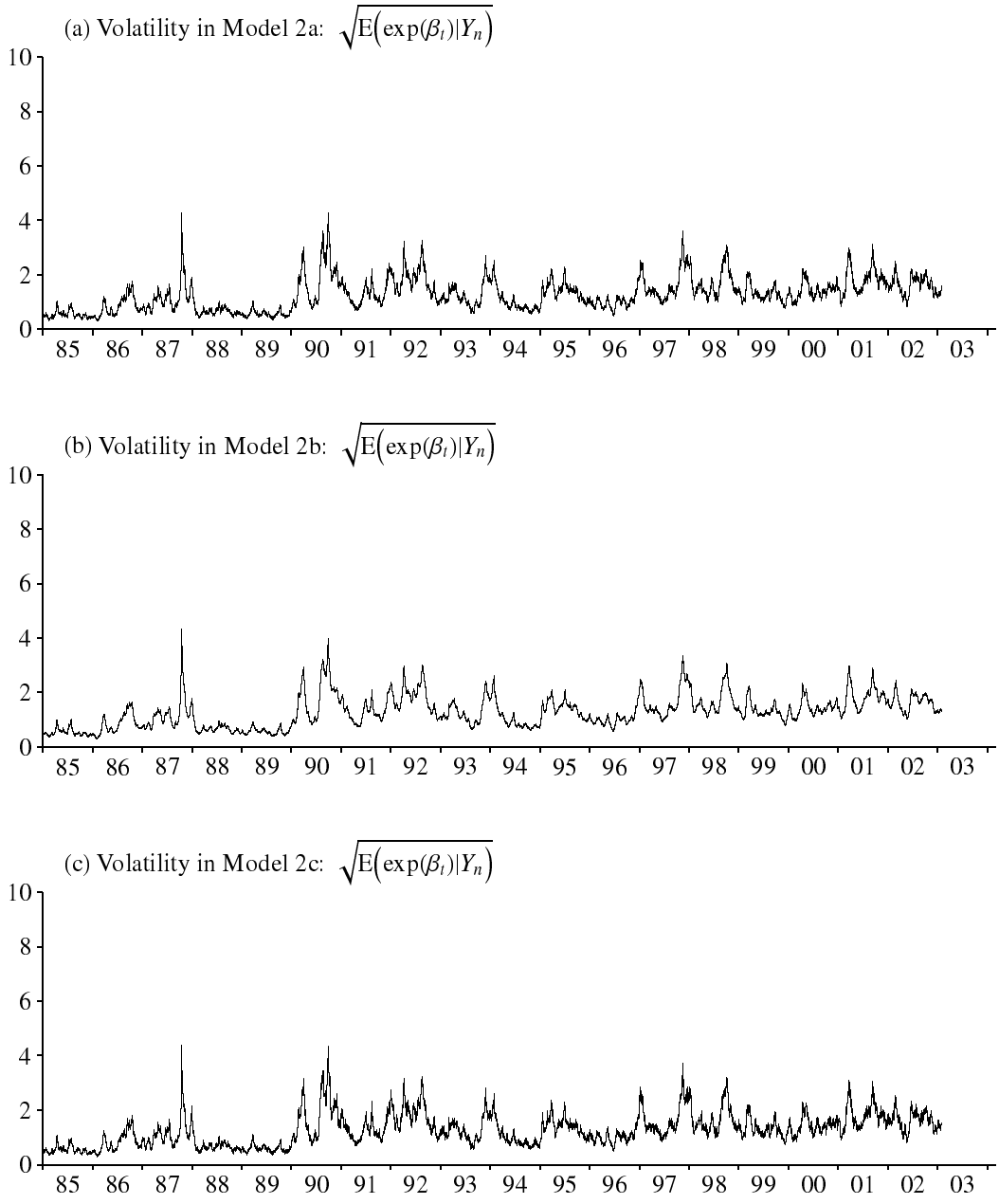


Figure 6.7: Volatility in Models 2a – 2c



The trend component represents Japanese business cycle to some extent. From the confidence intervals, the trend component is significantly positive during the period from 1985 to 1989, which period corresponds to the bubble period, and negative from 2000 to 2003, which period indicates a recent serious recession in Japan. Furthermore, we have observed that volatility is relatively small before 1990 but large after 1990. After the bubble burst in 1990, the volatility clearly increases.

6.7 Summary

In this chapter, we have introduced two kinds of nonlinear and non-Gaussian filters and smoothers, i.e., recursive and non-recursive algorithms. In the recursive algorithms (Section 6.3), given random draws of α_{t-1}, random draws of α_t are recursively generated from $f(\alpha_t|Y_t)$ for filtering and $f(\alpha_t|Y_n)$ for smoothing. In the non-recursive procedure (Section 6.4), random draws of A_n are simultaneously generated from $f(A_n|Y_n)$ utilizing the Gibbs sampler, where $A_n = \{\alpha_0, \alpha_1, \cdots, \alpha_n\}$.

Recursive Algorithm: The recursive algorithm introduced in Section 6.3 is based on the joint densities of the state variables, i.e., $f(\alpha_t, \alpha_{t-1}|Y_t)$ for filtering and $f(\alpha_{t+1}, \alpha_t|Y_n)$ or $f(\alpha_{t+1}, \alpha_t, \alpha_{t-1}|Y_n)$ for smoothing, where the sampling techniques such as RS, IR and MH are applied to generate random draws of α_t given Y_s, $s = t, n$.

It might be expected that RS gives us the most precise state estimates and that MH yields the worst of the three sampling techniques, which results are consistent with the simulation results from the Monte Carlo studies. For RS, however, we need to compute the supremum in the acceptance probability. Especially, as for equations (6.20), (6.21), (6.23) and (6.24), we often have the case where the supremum does not exist or the case where it is difficult to compute the supremum. Therefore, for equations (6.20), (6.21), (6.23) and (6.24), it is better to use IR, rather than RS and MH. Moreover, even though the supremum exists, it takes a lot of time computationally to obtain the random draws of the state variable α_t when the acceptance probability is close to zero. Both MH and IR can be applied to almost all the nonlinear non-Gaussian cases. This feature of IR and MH is one of the advantages over RS. Moreover, computational burden of IR and MH does not depend on the acceptance probability. Accordingly, in the case of IR and MH, equation (6.19) is computationally equivalent to equation (6.20) for filtering and similarly equations (6.21), (6.23) and (6.24) give us the same computational burden for smoothing.

It is possible to take different sampling methods between filtering and smoothing, i.e., for example, RS may be taken for filtering while IR is used for smoothing. Or at different time periods we can adopt different sampling densities. That is, taking an example of filtering, the sampling density is taken as $f_*(\alpha_t|\alpha_{t-1})f(\alpha_{t-1}|Y_{t-1})$ if $t = t'$ and $f(\alpha_t|Y_{t-1})$ otherwise. It might be useful to introduce $f_*(\alpha_t|\alpha_{t-1})$ when $f(\alpha_t|Y_t)$ is far from $f(\alpha_t|Y_{t-1})$. Thus, the filters and smoothers discussed in this chapter are very flexible.

An attempt has been also made to reduce computational burden for smoothing. Smoothing is N' times more computer-intensive than filtering, because at each time period the order of computation is N for filtering and $N \times N'$ for smoothing. In equation (6.22), we do not necessarily choose $N' = N$. To reduce the computational disadvantage for smoothing, from the Monte Carlo studies (i.e., Table 6.8) we have obtained the result that we may take N' less than N.

For comparison with the smoothing procedures discussed in Section 6.3.2, the smoother based on the two-filter formula, which is developed by Kitagawa (1996), is discussed in Appendix 6.4. We have shown that using the sampling density the smoother is also rewritten in the same fashion. For Simulations I – III and V, the simulations studies are examined. As a result, it is found that the smoother based on the two-filter formula shows a good performance.

Non-Recursive Algorithm: As for a less computational procedure, Carlin, Polson and Stoffer (1992) and Carter and Kohn (1994, 1996) proposed the nonlinear non-Gaussian state space modeling with Gibbs sampler in a Bayesian framework. The state space models which they investigated cannot be applied to all the general nonlinear and non-Gaussian state-space models. That is, they dealt with the state-space models such that we can easily generate random draws from the target density. In such a sense, their models are quite specific. Geweke and Tanizaki (1999, 2001) extended them to any state space models. This non-recursive algorithm, called GM in Section 6.5, is also discussed in Section 6.4. From the Monte Carlo results, we can see that GM does not necessarily perform well, depending on nonlinearity and nonnormality of the system. That is, we have obtained that GM shows a good performance for Simulations I – III and V but a poor performance for Simulations IV and VI.

Empirical Example: In Section 6.6, the Japanese stock market is investigated as an empilical application of the nonlinear non-Gaussian state space model. Estimation of the unknown parameter is quite difficult for the state space model. The difficulty arises from the maximization problem of the likelihood function (6.17), which problem is discussed in Section 6.3.4. Moreover, it has been shown in the Monte Carlo studies that GM is really good for Simulations II and III (see Table 6.6). In the empirical example, we have taken the same functional form as Simulations II and III. From these facts, therefore, GM in Section 6.5 is utilized together with the Bayes approach. We consider that the percent change of the Nikkei stock average consists of the trend and irregular components. The two components are separately estimated by using the state space model. We find that (i) the trend componet is very small but it is important for the business cycle and (ii) the volatility at time t, included in the irregular component, depends on the positive or negative sign of the stock return at time $t-1$ and the number of nontrading days between trading days t and $t-1$. The estimate of the trend component, $\hat{\alpha}_{t|n}$, clearly indicates the Japanese business cycle, where it is positive in the bubble period and negative in the recent serious recession. The volatility, the

estimate of $\sqrt{\mathrm{E}(\exp(\beta_t)|Y_n)}$, clearly increases after the bubble burst in 1990.

Appendix 6.1: Density-Based Recursive Algorithm

The prediction equation (6.13) and the updating equation (6.14) are derived as follows (see, for example, Kitagawa (1987), Kramer and Sorenson (1988), and Harvey (1989)):

(Prediction equation)

$$\begin{aligned} f(\alpha_t|Y_{t-1}) &= \int f(\alpha_t, \alpha_{t-1}|Y_{t-1})\, \mathrm{d}\alpha_{t-1} \\ &= \int f(\alpha_t|\alpha_{t-1}, Y_{t-1}) f(\alpha_{t-1}|Y_{t-1})\, \mathrm{d}\alpha_{t-1} \\ &= \int f_\alpha(\alpha_t|\alpha_{t-1}) f(\alpha_{t-1}|Y_{t-1})\, \mathrm{d}\alpha_{t-1}, \end{aligned}$$

(Updating equation)

$$\begin{aligned} f(\alpha_t|Y_t) &= f(\alpha_t|y_t, Y_{t-1}) = \frac{f(\alpha_t, y_t|Y_{t-1})}{f(y_t|Y_{t-1})} = \frac{f(\alpha_t, y_t|Y_{t-1})}{\int f(\alpha_t, y_t|Y_{t-1})\, \mathrm{d}\alpha_t} \\ &= \frac{f(y_t|\alpha_t, Y_{t-1}) f(\alpha_t|Y_{t-1})}{\int f(y_t|\alpha_t, Y_{t-1}) f(\alpha_t|Y_{t-1})\, \mathrm{d}\alpha_t} \\ &= \frac{f_y(y_t|\alpha_t) f(\alpha_t|Y_{t-1})}{\int f_y(y_t|\alpha_t) f(\alpha_t|Y_{t-1})\, \mathrm{d}\alpha_t}, \end{aligned}$$

for $t = 1, 2, \cdots, n$. In the third equality of the prediction equation, $f(\alpha_t|\alpha_{t-1}, Y_{t-1}) = f_\alpha(\alpha_t|\alpha_{t-1})$ is utilized. Similarly, in the fifth equality of the updating equation, $f(y_t|\alpha_t, Y_{t-1}) = f_y(y_t|\alpha_t)$ is taken. Equations (6.13) and (6.14) show a recursive algorithm of the density functions, which is called the density-based recursive filtering algorithm in this chapter. $f_\alpha(\alpha_t|\alpha_{t-1})$ is obtained from the transition equation (6.2) if the distribution of the error term η_t is specified, and also $f_y(y_t|\alpha_t)$ is derived from the measurement equation (6.1) given the specific distribution of the error term ϵ_t. Thus, the filtering density $f(\alpha_t|Y_t)$ is obtained recursively, given the distributions $f_\alpha(\alpha_t|\alpha_{t-1})$ and $f_y(y_t|\alpha_t)$.

Note as follows. If the past information Y_{t-1} is included in the transition equation (6.2), i.e., $\alpha_t = p_t(\alpha_{t-1}, \eta_t; Y_{t-1})$, $f_\alpha(\alpha_t|\alpha_{t-1})$ becomes $f_\alpha(\alpha_t|\alpha_{t-1}, Y_{t-1})$. Similarly, if the past information Y_{t-1} is in the measurement equation (6.1), i.e., $y_t = h_t(\alpha_t, \epsilon_t; Y_{t-1})$, $f_y(y_t|\alpha_t)$ should be replaced by $f_y(y_t|\alpha_t, Y_{t-1})$. Thus, the transition and measurement equations may depend on the past information.

The density-based fixed-interval smoothing algorithm (6.15) is represented as (see, for example, Kitagawa (1987) and Harvey (1989)):

$$f(\alpha_t|Y_n) = \int f(\alpha_t, \alpha_{t+1}|Y_n)\, \mathrm{d}\alpha_{t+1} = \int f(\alpha_{t+1}|Y_n) f(\alpha_t|\alpha_{t+1}, Y_n)\, \mathrm{d}\alpha_{t+1}$$

$$
\begin{aligned}
&= \int f(\alpha_{t+1}|Y_n) f(\alpha_t|\alpha_{t+1}, Y_t)\, \mathrm{d}\alpha_{t+1} = \int f(\alpha_{t+1}|Y_n) \frac{f(\alpha_t, \alpha_{t+1}|Y_t)}{f(\alpha_{t+1}|Y_t)}\, \mathrm{d}\alpha_{t+1} \\
&= \int f(\alpha_{t+1}|Y_n) \frac{f_\alpha(\alpha_{t+1}|\alpha_t) f(\alpha_t|Y_t)}{f(\alpha_{t+1}|Y_t)}\, \mathrm{d}\alpha_{t+1} \\
&= f(\alpha_t|Y_t) \int \frac{f(\alpha_{t+1}|Y_n) f_\alpha(\alpha_{t+1}|\alpha_t)}{f(\alpha_{t+1}|Y_t)}\, \mathrm{d}\alpha_{t+1},
\end{aligned}
$$

for $t = n-1, n-2, \cdots, 1$. We use $f(\alpha_t|\alpha_{t+1}, Y_n) = f(\alpha_t|\alpha_{t+1}, Y_t)$ in the third equality and $f(\alpha_t, \alpha_{t+1}|Y_t) = f_\alpha(\alpha_{t+1}|\alpha_t) f(\alpha_t|Y_t)$ in the fifth equation, which comes from the prediction equation shown above. The density function $f_\alpha(\alpha_{t+1}|\alpha_t)$ obtained from the transition equation (6.2) and the density functions $f(\alpha_{t+1}|Y_t)$ and $f(\alpha_t|Y_t)$ which are obtained from equations (6.13) and (6.14) in the filtering algorithm yield the above density-based fixed-interval smoothing algorithm (6.15), which is a backward recursion from $f(\alpha_{t+1}|Y_n)$ to $f(\alpha_t|Y_n)$. Note that the smoothing density at time $t = n$ (i.e., the endpoint in the smoothing algorithm) is equivalent to the filtering density at time $t = n$. Thus, the fixed-interval smoothing is derived together with the filtering algorithm given by equations (6.13) and (6.14).

Appendix 6.2: Recursive and Non-Recursive Algorithms

In Sections 6.3 and 6.4, we introduce two density-based algorithms on prediction, filtering and smoothing The conventional recursive algorithms are represented by equations (6.13) – (6.15) in Section 6.3. Equations (6.31) and (6.33) in Section 6.4 indicate the non-recursive algorithms. In this appendix, it is shown that both algorithms are equivalent, That is, we can derive equation (6.13) from equation (6.29), equation (6.14) from equation (6.31) and equation (6.15) from equation (6.33), respectively.

One-Step Ahead Prediction: Equation (6.13) is rewritten as:

$$
\begin{aligned}
f(\alpha_t|Y_{t-1}) &= \int f(A_t|Y_{t-1})\, \mathrm{d}A_{t-1} = \int f_\alpha(\alpha_t|\alpha_{t-1}) f(A_{t-1}|Y_{t-1})\, \mathrm{d}A_{t-1} \\
&= \iint f_\alpha(\alpha_t|\alpha_{t-1}) f(A_{t-1}|Y_{t-1})\, \mathrm{d}A_{t-2}\, \mathrm{d}\alpha_{t-1} \\
&= \int f_\alpha(\alpha_t|\alpha_{t-1}) \Big(\int f(A_{t-1}|Y_{t-1})\, \mathrm{d}A_{t-2} \Big)\, \mathrm{d}\alpha_{t-1} \\
&= \int f_\alpha(\alpha_t|\alpha_{t-1}) f(\alpha_{t-1}|Y_{t-1})\, \mathrm{d}\alpha_{t-1}. \qquad (6.48)
\end{aligned}
$$

The second equality in equation (6.48) uses the following two equations:

$$
\begin{aligned}
f(A_t, Y_{t-1}) &= f_\alpha(A_t) f_y(Y_{t-1}|A_{t-1}) = f_\alpha(\alpha_t|\alpha_{t-1}) f_\alpha(A_{t-1}) f_y(Y_{t-1}|A_{t-1}), \\
f(A_t|Y_{t-1}) &= \frac{f_\alpha(A_t) f_y(Y_{t-1}|A_{t-1})}{f(Y_{t-1})}.
\end{aligned}
$$

Thus, it can be easily shown that equation (6.13) is equivalent to equation (6.29).

Filtering: Equation (6.14) is transformed as:

$$\begin{aligned} f(\alpha_t|Y_t) &= \frac{\int f(A_t, Y_t)\, \mathrm{d}A_{t-1}}{\int f(A_t, Y_t)\, \mathrm{d}A_t} = \frac{\int f_y(y_t|\alpha_t) f_\alpha(\alpha_t|\alpha_{t-1}) f(A_{t-1}, Y_{t-1})\, \mathrm{d}A_{t-1}}{\iint f_y(y_t|\alpha_t) f_\alpha(\alpha_t|\alpha_{t-1}) f(A_{t-1}, Y_{t-1})\, \mathrm{d}A_{t-1}\, \mathrm{d}\alpha_t} \\ &= \frac{f_y(y_t|\alpha_t)\Big(\int f_\alpha(\alpha_t|\alpha_{t-1}) f(A_{t-1}|Y_{t-1})\, \mathrm{d}A_{t-1}\Big)}{\int f_y(y_t|\alpha_t)\Big(\int f_\alpha(\alpha_t|\alpha_{t-1}) f(A_{t-1}|Y_{t-1})\, \mathrm{d}A_{t-1}\Big)\, \mathrm{d}\alpha_t} \\ &= \frac{f_y(y_t|\alpha_t) f(\alpha_t|Y_{t-1})}{\int f_y(y_t|\alpha_t) f(\alpha_t|Y_{t-1})\, \mathrm{d}\alpha_t.} \end{aligned} \tag{6.49}$$

Note that $f(A_t, Y_t) = f_y(y_t|\alpha_t) f_\alpha(\alpha_t|\alpha_{t-1}) f(A_{t-1}, Y_{t-1})$ in the second equality of equation (6.49). $f(\alpha_t|Y_{t-1}) = \int f_\alpha(\alpha_t|\alpha_{t-1}) f(A_{t-1}|Y_{t-1})\, \mathrm{d}A_{t-1}$ is substituted into the fourth equality, which equation comes from the second equality of equation (6.48). Thus, equation (6.14) is derived from equation (6.31).

Smoothing: Let us define $Y_t^+ = \{y_t, y_{t+1}, \cdots, y_n\}$. Suppose that the joint density function of A_{t+1}^+ and Y_{t+1}^+ is given by:

$$f(A_{t+1}^+, Y_{t+1}^+) = \prod_{s=t+2}^{n} f_\alpha(\alpha_s|\alpha_{s-1}) \prod_{s=t+1}^{n} f_y(y_s|\alpha_s),$$

which implies that the joint density of A_n and Y_n is represented as follows:

$$f(A_n, Y_n) = f(A_t, Y_t) f_\alpha(\alpha_{t+1}|\alpha_t) f(A_{t+1}^+, Y_{t+1}^+),$$

which is utilized in the second and eighth equalities of equation (6.50). Equation (6.33) is represented as:

$$\begin{aligned} f(\alpha_t|Y_n) &= \frac{1}{f(Y_n)} \iint f(A_n, Y_n)\, \mathrm{d}A_{t-1}\, \mathrm{d}A_{t+1}^+ \\ &= \frac{1}{f(Y_n)} \iint f(A_t, Y_t) f_\alpha(\alpha_{t+1}|\alpha_t) f(A_{t+1}^+, Y_{t+1}^+)\, \mathrm{d}A_{t-1}\, \mathrm{d}A_{t+1}^+ \\ &= \frac{1}{f(Y_n)} \Big(\int f(A_t, Y_t)\, \mathrm{d}A_{t-1}\Big)\Big(\int f_\alpha(\alpha_{t+1}|\alpha_t) f(A_{t+1}^+, Y_{t+1}^+)\, \mathrm{d}A_{t+1}^+\Big) \\ &= \frac{f(Y_t)}{f(Y_n)} f(\alpha_t|Y_t) \int f_\alpha(\alpha_{t+1}|\alpha_t) f(A_{t+1}^+, Y_{t+1}^+)\, \mathrm{d}A_{t+1}^+ \\ &= \frac{f(Y_t)}{f(Y_n)} f(\alpha_t|Y_t) \int \frac{f_\alpha(\alpha_{t+1}|\alpha_t) f(A_{t+1}^+, Y_{t+1}^+)}{\int f_\alpha(\alpha_{t+1}|\alpha_t) f(A_t, Y_t)\, \mathrm{d}A_t} \\ &\qquad \times \Big(\int f_\alpha(\alpha_{t+1}|\alpha_t) f(A_t, Y_t)\, \mathrm{d}A_t\Big)\, \mathrm{d}A_{t+1}^+ \end{aligned}$$

$$
\begin{aligned}
&= f(\alpha_t|Y_t) \int\int \frac{f_\alpha(\alpha_{t+1}|\alpha_t) f(A^+_{t+1}, Y^+_{t+1})}{f(Y_n) f(\alpha_{t+1}|Y_t)} \\
&\qquad \times \Big(\int f_\alpha(\alpha_{t+1}|\alpha_t) f(A_t, Y_t)\, \mathrm{d}A_t \Big)\, \mathrm{d}A^+_{t+2}\, \mathrm{d}\alpha_{t+1} \\
&= f(\alpha_t|Y_t) \int \frac{\iint f(A^+_{t+1}, Y^+_{t+1}) f_\alpha(\alpha_{t+1}|\alpha_t) f(A_t, Y_t)\, \mathrm{d}A_t\, \mathrm{d}A^+_{t+2}}{f(Y_n)} \\
&\qquad \times \frac{f_\alpha(\alpha_{t+1}|\alpha_t)}{f(\alpha_{t+1}|Y_t)}\, \mathrm{d}\alpha_{t+1} \\
&= f(\alpha_t|Y_t) \int \frac{\iint f(A_n, Y_n)\, \mathrm{d}A_t\, \mathrm{d}A^+_{t+2}}{f(Y_n)} \frac{f_\alpha(\alpha_{t+1}|\alpha_t)}{f(\alpha_{t+1}|Y_t)}\, \mathrm{d}\alpha_{t+1} \\
&= f(\alpha_t|Y_t) \int \frac{f(\alpha_{t+1}|Y_n) f_\alpha(\alpha_{t+1}|\alpha_t)}{f(\alpha_{t+1}|Y_t)}\, \mathrm{d}\alpha_{t+1}. \qquad (6.50)
\end{aligned}
$$

In the fourth equality, the first equality of equation (6.49) is utilized. Note that $f(Y_t) = \int f(A_t,\ Y_t)\, \mathrm{d}A_t$. The ninth equality comes from the first equality, i.e., $\int\int f(A_n,\ Y_n)/f(Y_n)\, \mathrm{d}A_t\ \ \mathrm{d}A^+_{t+2} = f(\alpha_{t+1}|Y_n)$. Thus, it is shown that equation (6.33) is exactly equivalent to equation (6.15).

Thus, it has been shown that both recursive and non-recursive algorithms are equivalent. In other words, we can derive equation (6.13) from equation (6.29), equation (6.14) from equation (6.31) and equation (6.15) from equation (6.33), respectively.

Appendix 6.3: Linear and Normal System

State-Space Model: Consider the case where the system is linear and normal, i.e.,

$$
\begin{aligned}
&\text{(Measurement equation)} \qquad y_t = Z_t\alpha_t + d_t + S_t\epsilon_t, \qquad (6.51) \\
&\text{(Transition equation)} \qquad \alpha_t = T_t\alpha_{t-1} + c_t + R_t\eta_t, \qquad (6.52) \\
&\qquad\qquad \begin{pmatrix} \epsilon_t \\ \eta_t \end{pmatrix} \sim N\Big(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} H_t & 0 \\ 0 & Q_t \end{pmatrix} \Big),
\end{aligned}
$$

where Z_t, d_t, S_t, T_t, c_t, R_t, H_t and Q_t are assumed to be known for all time $t = 1, 2, \cdots, T$. Define conditional mean and variance as $\alpha_{t|s} \equiv \mathrm{E}(\alpha_t|Y_s)$ and $\Sigma_{t|s} \equiv \mathrm{V}(\alpha_t|Y_s)$ for $s = t-1, t, n$. Under the above setup, optimal prediction, filtering and smoothing are represented as the standard linear recursive algorithms, which are easily derived from the first and second moments of density functions (6.13) – (6.15). See, for example, Tanizaki (1996).

Filtering: The density-based filtering algorithm is given by equations (6.13) and (6.14). This algorithm reduces to the following standard linear recursive algorithm:

$$\alpha_{t|t-1} = T_t\alpha_{t-1|t-1} + c_t, \qquad (6.53)$$

$$\Sigma_{t|t-1} = T_t \Sigma_{t-1|t-1} T_t' + R_t Q_t R_t', \tag{6.54}$$
$$y_{t|t-1} = Z_t \alpha_{t|t-1} + d_t, \tag{6.55}$$
$$F_{t|t-1} = Z_t \Sigma_{t|t-1} Z_t' + S_t H_t S_t', \tag{6.56}$$
$$K_t = \Sigma_{t|t-1} Z_t' F_{t|t-1}^{-1}, \tag{6.57}$$
$$\alpha_{t|t} = \alpha_{t|t-1} + K_t(y_t - y_{t|t-1}), \tag{6.58}$$
$$\Sigma_{t|t} = \Sigma_{t|t-1} - K_t F_{t|t-1} K_t', \tag{6.59}$$

for $t = 1, 2, \cdots, n$. Equations (6.53) and (6.54) are derived from equation (6.13), while equations (6.58) and (6.59) come from equation (6.14). Given the initial values $\alpha_{0|0}$ and $\Sigma_{0|0}$, the filtering mean and variance at time t (i.e., $\alpha_{t|t}$ and $\Sigma_{t|t}$) are recursively computed based on equations (6.53) – (6.59).

Under the normality assumption in addition to the linearity assumption, we derive the filtering algorithm (6.53) – (6.59). When α_0, ϵ_t and η_t are normally distributed, it is known that $f(\alpha_t|Y_s)$, $s = t, t-1$, is expressed by the Gaussian distribution:

$$f(\alpha_t|Y_s) = \Phi(\alpha_t - \alpha_{t|s},\ \Sigma_{t|s}),$$

where $\Phi(\alpha_t - \alpha_{t|s},\ \Sigma_{t|s})$ denotes the normal density with mean $\alpha_{t|s}$ and variance $\Sigma_{t|s}$, i.e.,

$$\Phi(\alpha_t - \alpha_{t|s},\ \Sigma_{t|s}) = (2\pi)^{-k/2} |\Sigma_{t|s}|^{-1/2} \exp\Big(-\frac{1}{2}(\alpha_t - \alpha_{t|s})' \Sigma_{t|s}^{-1} (\alpha_t - \alpha_{t|s})\Big).$$

Note as follows:

$$\Phi(\alpha_t - \alpha_{t|s}, \Sigma_{t|s}) \equiv N(\alpha_{t|s},\ \Sigma_{t|s}).$$

Then, from equation (6.13), one-step ahead prediction density $f(\alpha_t|Y_{t-1})$ is given by:

$$\begin{aligned} f(\alpha_t|Y_{t-1}) &= \int f_\alpha(\alpha_t|\alpha_{t-1}) f(\alpha_{t-1}|Y_{t-1})\, \mathrm{d}\alpha_{t-1} \\ &= \int \Phi(\alpha_t - T_t\alpha_{t-1} - c_t,\ R_t Q_t R_t')\ \Phi(\alpha_{t-1} - \alpha_{t-1|t-1},\ \Sigma_{t-1|t-1})\, \mathrm{d}\alpha_{t-1} \\ &= \Phi(\alpha_t - T_t\alpha_{t-1|t-1} - c_t,\ T_t \Sigma_{t-1|t-1} T_t' + R_t Q_t R_t') \\ &\equiv \Phi(\alpha_t - \alpha_{t|t-1},\ \Sigma_{t|t-1}), \end{aligned} \tag{6.60}$$

where $f_\alpha(\alpha_t|\alpha_{t-1})$ and $f(\alpha_{t-1}|Y_{t-1})$ are rewritten as follows:

$$\begin{aligned} f_\alpha(\alpha_t|\alpha_{t-1}) &= \Phi(\alpha_t - T_t\alpha_{t-1} - c_t,\ R_t Q_t R_t'), \\ f(\alpha_{t-1}|Y_{t-1}) &= \Phi(\alpha_{t-1} - \alpha_{t-1|t-1},\ \Sigma_{t-1|t-1}). \end{aligned}$$

We show that the third equality in equation (6.60) holds, i.e.,

$$\begin{aligned} &\int \Phi(\alpha_t - T_t\alpha_{t-1} - c_t,\ R_t Q_t R_t')\ \Phi(\alpha_{t-1} - \alpha_{t-1|t-1},\ \Sigma_{t-1|t-1})\, \mathrm{d}\alpha_{t-1} \\ &\quad = \Phi(\alpha_t - T_t\alpha_{t-1|t-1} - c_t,\ T_t \Sigma_{t-1|1-1} T_t' + R_t Q_t R_t'). \end{aligned}$$

For simplicity of discussion, each variable is re-defined as follows:

$$
\begin{aligned}
x &= \alpha_{t-1} - \alpha_{t-1|t-1}, \\
\Sigma_{xx} &= \Sigma_{t-1|t-1}, \\
y &= \alpha_t - T_t\alpha_{t-1|t-1} - c_t, \\
\Sigma_{yy} &= R_t Q_t R_t', \\
A &= T_t.
\end{aligned}
$$

Substituting each variable, the equality to be proved is given by:

$$
\int \Phi(y - Ax,\ \Sigma_{yy})\ \Phi(x,\ \Sigma_{xx})\ \mathrm{d}x = \Phi(y,\ A\Sigma_{xx}A' + \Sigma_{yy}).
$$

The two normal distributions $\Phi(y - Ax,\ \Sigma_{yy})$ and $\Phi(x,\ \Sigma_{xx})$ are written as:

$$
\begin{aligned}
\Phi(y - Ax,\ \Sigma_{yy}) &= (2\pi)^{-g/2}|\Sigma_{yy}|^{-1/2} \exp\left(-\frac{1}{2}(y - Ax)'\Sigma_{yy}^{-1}(y - Ax)\right), \\
\Phi(x,\ \Sigma_{xx}) &= (2\pi)^{-k/2}|\Sigma_{xx}|^{-1/2} \exp\left(-\frac{1}{2}x'\Sigma_{xx}^{-1}x\right).
\end{aligned}
$$

The dimensions of y and x are given by $g \times 1$ and $k \times 1$, respectively. A denotes a $g \times k$ matrix. Note that here we have $g = k$. To derive updating equations (6.58) and (6.59), we need to use $g \neq k$. Therefore, for now, g should be distinguished from k.

A product of these two normal densities is transformed into:

$$
\begin{aligned}
&\Phi(y - Ax,\ \Sigma_{yy})\ \Phi(x,\ \Sigma_{xx}) \\
&\quad = (2\pi)^{-g/2}|\Sigma_{yy}|^{-1/2} \exp\left(-\frac{1}{2}(y - Ax)'\Sigma_{yy}^{-1}(y - Ax)\right) \\
&\qquad\quad \times(2\pi)^{-k/2}|\Sigma_{xx}|^{-1/2} \exp\left(-\frac{1}{2}x'\Sigma_{xx}^{-1}x\right) \\
&\quad = (2\pi)^{-(g+k)/2}|\Sigma_{yy}|^{-1/2}\ |\Sigma_{xx}|^{-1/2} \\
&\qquad\quad \times\exp\left(-\frac{1}{2}\begin{pmatrix} y - Ax \\ x \end{pmatrix}'\begin{pmatrix} \Sigma_{yy} & 0 \\ 0 & \Sigma_{xx} \end{pmatrix}^{-1}\begin{pmatrix} y - Ax \\ x \end{pmatrix}\right) \\
&\quad = (2\pi)^{-(g+k)/2}|\Sigma_{yy}|^{-1/2}\ |\Sigma_{xx}|^{-1/2} \\
&\qquad\quad \times\exp\left(-\frac{1}{2}\begin{pmatrix} y \\ x \end{pmatrix}'\begin{pmatrix} I_g & -A \\ 0 & I_k \end{pmatrix}'\begin{pmatrix} \Sigma_{yy} & 0 \\ 0 & \Sigma_{xx} \end{pmatrix}^{-1}\begin{pmatrix} I_g & -A \\ 0 & I_k \end{pmatrix}\begin{pmatrix} y \\ x \end{pmatrix}\right) \\
&\quad = (2\pi)^{-(g+k)/2}|\Sigma_{yy}|^{-1/2}\ |\Sigma_{xx}|^{-1/2} \\
&\qquad\quad \times\exp\left(-\frac{1}{2}\begin{pmatrix} y \\ x \end{pmatrix}'\begin{pmatrix} I_g & A \\ 0 & I_k \end{pmatrix}'^{-1}\begin{pmatrix} \Sigma_{yy} & 0 \\ 0 & \Sigma_{xx} \end{pmatrix}^{-1}\begin{pmatrix} I_g & A \\ 0 & I_k \end{pmatrix}^{-1}\begin{pmatrix} y \\ x \end{pmatrix}\right) \\
&\quad = (2\pi)^{-(g+k)/2}|\Sigma_{yy}|^{-1/2}\ |\Sigma_{xx}|^{-1/2} \\
&\qquad\quad \times\exp\left(-\frac{1}{2}\begin{pmatrix} y \\ x \end{pmatrix}'\left(\begin{pmatrix} I_g & A \\ 0 & I_k \end{pmatrix}\begin{pmatrix} \Sigma_{yy} & 0 \\ 0 & \Sigma_{xx} \end{pmatrix}\begin{pmatrix} I_g & A \\ 0 & I_k \end{pmatrix}'\right)^{-1}\begin{pmatrix} y \\ x \end{pmatrix}\right)
\end{aligned}
$$

$$= (2\pi)^{-(g+k)/2}|\Sigma_{yy}|^{-1/2}\,|\Sigma_{xx}|^{-1/2}$$
$$\times\exp\left(-\frac{1}{2}\begin{pmatrix} y \\ x \end{pmatrix}'\begin{pmatrix} A\Sigma_{xx}A'+\Sigma_{yy} & A\Sigma_{xx} \\ \Sigma_{xx}A' & \Sigma_{xx} \end{pmatrix}^{-1}\begin{pmatrix} y \\ x \end{pmatrix}\right).$$

Note that, in deriving the above equation, the inverse of the following matrix is used.

$$\begin{pmatrix} I_g & -A \\ 0 & I_k \end{pmatrix}^{-1} = \begin{pmatrix} I_g & A \\ 0 & I_k \end{pmatrix}.$$

Furthermore, we have:

$$\begin{vmatrix} A\Sigma_{xx}A'+\Sigma_{yy} & A\Sigma_{xx} \\ \Sigma_{xx}A' & \Sigma_{xx} \end{vmatrix} = |\Sigma_{xx}||\Sigma_{yy}|.$$

Accordingly, the joint distribution of x and y is represented by the following bivariate normal distribution.

$$\begin{pmatrix} y \\ x \end{pmatrix} \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} A\Sigma_{xx}A'+\Sigma_{yy} & A\Sigma_{xx} \\ \Sigma_{xx}A' & \Sigma_{xx} \end{pmatrix}\right), \tag{6.61}$$

The marginal density of y is given by:

$$f(y) = \Phi(y,\ A\Sigma_{xx}A'+\Sigma_{yy}),$$

which implies that

$$f(\alpha_t|Y_{t-1}) = \Phi(\alpha_t - T_t\alpha_{t-1|t-1} - c_t,\ T_t\Sigma_{t-1|1-1}T_t' + R_tQ_tR_t').$$

Thus, the third equality in equation (6.60) is derived.

Comparing each argument (moment) in the two normal densities given by the fourth line and the fifth line in equation (6.60), we can derive the prediction equations (6.53) and (6.54).

Next, to derive the updating equations, equation (6.14) is calculated as follows:

$$\begin{aligned} f(\alpha_t|Y_t) &= \frac{f_y(y_t|\alpha_t)f(\alpha_t|Y_{t-1})}{\int f_y(y_t|\alpha_t)f(\alpha_t|Y_{t-1})\,\mathrm{d}\alpha_t} \\ &= \frac{\Phi(y_t - Z_t\alpha_t - d_t,\ S_tH_tS_t')\,\Phi(\alpha_t - \alpha_{t|t-1},\ \Sigma_{t|t-1})}{\int \Phi(y_t - Z_t\alpha_t - d_t,\ S_tH_tS_t')\,\Phi(\alpha_t - \alpha_{t|t-1},\ \Sigma_{t|t-1})\,\mathrm{d}\alpha_t} \\ &= \Phi(\alpha_t - \alpha_{t|t-1} - K_t(y_t - y_{t|t-1}),\ \Sigma_{t|t-1} - K_tF_{t|t-1}K_t') \\ &\equiv \Phi(\alpha_t - \alpha_{t|t},\ \Sigma_{t|t}), \end{aligned} \tag{6.62}$$

where

$$\begin{aligned} y_{t|t-1} &\equiv \mathrm{E}(y_t|Y_{t-1}) = Z_t\alpha_{t|t-1} + d_t, \\ F_{t|t-1} &\equiv \mathrm{V}(y_t|Y_{t-1}) = Z_t\Sigma_{t|t-1}Z_t' + S_tH_tS_t', \\ K_t &= \Sigma_{t|t-1}Z_t'F_{t|t-1}^{-1}. \end{aligned}$$

Note that $f_y(y_t|\alpha_t)$ and $f(\alpha_t|Y_{t-1})$ are rewritten as:

$$f_y(y_t|\alpha_t) = \Phi(y_t - Z_t\alpha_t - d_t,\ S_t H_t S_t'),$$
$$f(\alpha_t|Y_{t-1}) = \Phi(\alpha_t - \alpha_{t|t-1},\ \Sigma_{t|t-1}).$$

We show that the third equality in equation (6.62) holds, i.e.,

$$\Phi(y_t - Z_t\alpha_t - d_t,\ S_t H_t S_t')\ \Phi(\alpha_t - \alpha_{t|t-1},\ \Sigma_{t|t-1})$$
$$= \Phi(\alpha_t - \alpha_{t|t-1} - K_t(y_t - y_{t|t-1}),\ \Sigma_{t|t-1} - K_t F_{t|t-1} K_t')\ \Phi(y_t - y_{t|t-1},\ F_{t|t-1}).$$

Similarly, we re-define each variable as follows:

$$x = \alpha_t - \alpha_{t|t-1},$$
$$\Sigma_{xx} = \Sigma_{t|t-1},$$
$$y = y_t - Z_t\alpha_{t|t-1} - d_t,$$
$$\Sigma_{yy} = S_t H_t S_t',$$
$$A = Z_t,$$

where x is a $k \times 1$ vector, y is a $g \times 1$ vector, and A is a $g \times k$ matrix.

By taking into account the following three equations,

$$y_{t|t-1} = Z_t\alpha_{t|t-1} + d_t,$$
$$F_{t|t-1} = Z_t\Sigma_{t|t-1}Z_t' + S_t H_t S_t',$$
$$K_t = \Sigma_{t|t-1}Z_t' F_{t|t-1}^{-1},$$

the equation to be proved is represented by:

$$\Phi(y - Ax,\ \Sigma_{yy})\ \Phi(x,\ \Sigma_{xx})$$
$$= \Phi(x - \Sigma_{xx}A'(A\Sigma_{xx}A' + \Sigma_{yy})^{-1}y,\ \Sigma_{xx} - \Sigma_{xx}A'(A\Sigma_{xx}A' + \Sigma_{yy})^{-1}A\Sigma_{xx})$$
$$\times\Phi(y,\ A\Sigma_{xx}A' + \Sigma_{yy}).$$

We can prove the above equation in the exactly same fashion, and accordingly the joint density of x and y (i.e., $f(x, y)$) follows the same distribution as equation (6.61).

The conditional density of x given y, i.e., $f(x|y)$, and the marginal density of y, i.e., $f(y)$, are derived as:

$$f(x, y) = f(x|y)f(y)$$
$$= \Phi(x - \Sigma_{xx}A'(A\Sigma_{xx}A' + \Sigma_{yy})^{-1}y,\ \Sigma_{xx} - \Sigma_{xx}A'(A\Sigma_{xx}A' + \Sigma_{yy})^{-1}A\Sigma_{xx})$$
$$\times\Phi(y,\ A\Sigma_{xx}A' + \Sigma_{yy}),$$

where

$$f(x|y) = \Phi(x - \Sigma_{xx}A'(A\Sigma_{xx}A' + \Sigma_{yy})^{-1}y,\ \Sigma_{xx} - \Sigma_{xx}A'(A\Sigma_{xx}A' + \Sigma_{yy})^{-1}A\Sigma_{xx}),$$
$$f(y) = \Phi(y,\ A\Sigma_{xx}A' + \Sigma_{yy}).$$

Therefore, the following equation is obtained.

$$\begin{aligned}&\Phi(y_t - Z_t\alpha_t - d_t,\ S_t H_t S_t')\ \Phi(\alpha_t - \alpha_{t|t-1},\ \Sigma_{t|t-1}) \\ &\quad = \Phi(\alpha_t - \alpha_{t|t-1} - K_t(y_t - y_{t|t-1}),\ \Sigma_{t|t-1} - K_t F_{t|t-1} K_t')\ \Phi(y_t - y_{t|t-1},\ F_{t|t-1}).\end{aligned}$$

Note that we have the following:

$$\Phi(y_t - y_{t|t-1},\ F_{t|t-1}) = \int f_y(y_t|\alpha_t) f(\alpha_t|Y_{t-1})\, \mathrm{d}\alpha_t.$$

Therefore, the updating equations (6.58) and (6.59) are obtained, comparing each argument in equation (6.62). Thus, the filtering algorithm is represented by (6.53) – (6.59).

When the state space model is linear and normal as in equations (6.51) and (6.52), the Kalman filter estimate is optimal in the sense that it minimizes the mean square error. When the normality assumption is dropped, there is no longer any guarantee that the Kalman filter gives the conditional mean of the state vector. However, it is still an optimal estimator in the sense that it minimizes the mean square error within the class of all linear estimators (see Harvey (1989)). See Anderson and Moore (1979), Gelb (1974), Jazwinski (1970) and Tanizaki (1996) for the Kalman filter algorithm.

Smoothing: The first and second moments of the smoothing density (6.15) give us the following backward recursive algorithm:

$$C_t = \Sigma_{t|t} T_{t+1}' \Sigma_{t+1|t}^{-1}, \tag{6.63}$$

$$\alpha_{t|n} = \alpha_{t|t} + C_t(\alpha_{t+1|n} - \alpha_{t+1|t}), \tag{6.64}$$

$$\Sigma_{t|n} = \Sigma_{t|t} + C_t(\Sigma_{t+1|n} - \Sigma_{t+1|t})C_t', \tag{6.65}$$

for $t = n-1, n-2, \cdots, 1$. Given $\alpha_{t|t}$, $\Sigma_{t|t}$, $\alpha_{t+1|t}$ and $\Sigma_{t+1|t}$, smoothing mean and variance at time t (i.e., $\alpha_{t|n}$ and $\Sigma_{t|n}$) is obtained recursively, using equations (6.63) – (6.65). The smoothing algorithm (6.63) – (6.65) is also derived from the density-based recursive algorithm (6.15), given the normality assumption of the error terms.

Likelihood Function: When Z_t, d_t, S_t, T_t, c_t, R_t, H_t and Q_t depends on an unknown parameter, the following log-likelihood function is maximized with respect to the parameter:

$$\begin{aligned}\log f(Y_n) = &-\frac{n}{2}\log(2\pi) - \frac{1}{2}\sum_{t=1}^{n}\log|F_{t|t-1}| \\ &- \frac{1}{2}\sum_{t=1}^{n}(y_t - y_{t|t-1})' F_{t|t-1}^{-1}(y_t - y_{t|t-1}),\end{aligned} \tag{6.66}$$

which is also obtained from equation (6.17). Note that the conditional distribution of y_t given Y_{t-1} is represented as $y_t|Y_{t-1} \sim N(y_{t|t-1}, F_{t|t-1})$, where both $y_{t|t-1}$ and $F_{t|t-1}$ are obtained from the above standard filtering algorithm. Therefore, we do not need extra computation to evaluate the log-likelihood function (6.66).

Appendix 6.4: Two-Filter Formula

Kitagawa (1996) suggested the Monte Carlo smoother based on the two-filter formula. In this appendix, we show that the same approach used in Section 6.3 can be applied. Define $Y_t^+ \equiv \{y_t, y_{t+1}, \cdots, y_n\}$, where we have $Y_n = Y_{t-1} \cup Y_t^+$. The fixed-interval smoothing density $f(\alpha_t|Y_n)$ is represented as:

$$f(\alpha_t|Y_n) \propto f(Y_t^+|\alpha_t) f(\alpha_t|Y_{t-1}), \tag{6.67}$$

where $f(Y_t^+|\alpha_t)$ is recursively obtained as follows:

$$f(Y_t^+|\alpha_t) = f_y(y_t|\alpha_t) \int f(Y_{t+1}^+|\alpha_{t+1}) f_\alpha(\alpha_{t+1}|\alpha_t)\, \mathrm{d}\alpha_{t+1}, \tag{6.68}$$

for $t = n-1, n-2, \cdots, 1$. The initial condition is given by: $f(Y_n^+|\alpha_n) = f_y(y_n|\alpha_n)$.

First, we consider evaluating $f(Y_t^+|\alpha_t)$ in the backward recursion. Let $f_*(\alpha_t)$ be the sampling density and $\alpha_{i,t}^*$ be the ith random draw of α_t generated from $f_*(\alpha_t)$. From equation (6.68), the density $f(Y_t^+|\alpha_t)$ evaluated at $\alpha_t = \alpha_{i,t}^*$ is rewritten as:

$$\begin{aligned} f(Y_t^+|\alpha_{i,t}^*) &= f_y(y_t|\alpha_{i,t}^*) \int \frac{f(Y_{t+1}^+|\alpha_{t+1}) f_\alpha(\alpha_{t+1}|\alpha_{i,t}^*)}{f_*(\alpha_{t+1})} f_*(\alpha_{t+1})\, \mathrm{d}\alpha_{t+1} \\ &\approx f_y(y_t|\alpha_{i,t}^*) \frac{1}{N''} \sum_{j=1}^{N''} \frac{f(Y_{t+1}^+|\alpha_{j,t+1}^*) f_\alpha(\alpha_{j,t+1}^*|\alpha_{i,t}^*)}{f_*(\alpha_{j,t+1}^*)}, \end{aligned} \tag{6.69}$$

for $t = n-1, n-2, \cdots, 1$. In the second line of the above equation, the integration is evaluated by $\alpha_{j,t+1}^*$, $j = 1, 2, \cdots, N''$, where N'' is not necessarily equal to N. Thus, $f(Y_t^+|\alpha_{i,t}^*)$ is recursively obtained for $t = n-1, n-2, \cdots, 1$. Note that the sampling density $f_*(\alpha_t)$ may depend on the state variable at time $t-1$, i.e., $f_*(\alpha_t|\alpha_{t-1})$.

Next, given $f(Y_t^+|\alpha_{i,t}^*)$, we generate random draws of α_t from $f(\alpha_t|Y_n)$. We can rewrite equation (6.67) as follows:

$$f(\alpha_t|Y_n) \propto q_6(\alpha_t) f(\alpha_t|Y_{t-1}), \tag{6.70}$$

where $q_6(\alpha_t) \propto f(Y_t^+|\alpha_t)$. In this case, we have to take the sampling density as $f_*(\alpha_t) = f(\alpha_t|Y_{t-1})$, i.e., $\alpha_{i,t}^* = \alpha_{i,t|t-1}$. Moreover, we need to compute $f_*(\alpha_{t+1}^*) = f(\alpha_{t+1}|Y_t)$ evaluated at $\alpha_{t+1}^* = \alpha_{j,t+1|t}$ in the denominator of equation (6.69). As it is shown in equation (6.22), evaluation of $f(\alpha_{j,t+1|t}|Y_t)$ becomes N' times more computer-intensive. Therefore, it is not realistic to take the sampling density as $f_*(\alpha_t) = f(\alpha_t|Y_{t-1})$.

Alternatively, as discussed in Section 6.3, we consider generating random draws from the joint density of α_t and α_{t-1} given Y_n. Substituting equation (6.13) into equation (6.67) and eliminating the integration with respect to α_{t-1}, equation (6.67) is rewritten as:

$$f(\alpha_t, \alpha_{t-1}|Y_n) \propto q_7(\alpha_t, \alpha_{t-1}) f_*(\alpha_t) f(\alpha_{t-1}|Y_{t-1}), \tag{6.71}$$

where $q_7(\alpha_t, \alpha_{t-1}) \propto f(Y_t^+|\alpha_t) f_\alpha(\alpha_t|\alpha_{t-1}) / f_*(\alpha_t)$.

As shown in equation (6.69), we can evaluate $f(Y_t^+|\alpha_t)$ at $\alpha_t = \alpha_{i,t}^*$, but it is hard and time-consuming to obtain the supremum of $f(Y_t^+|\alpha_t)$ because the summation is included in equation (6.69). Accordingly it is not possible to compute the supremum of $q_6(\alpha_t)$ and $q_7(\alpha_t, \alpha_{t-1})$. Therefore, it is difficult to apply RS to this smoother. We may apply IR and MH to the Monte Carlo smoother based on the two-filter formula.

Taking an example of the IR smoother based on equation (6.71), a random number of α_t from $f(\alpha_t|Y_n)$ is generated as follows. Define the probability weight $\omega(\alpha_{i,t}^*, \alpha_{i,t-1|t-1})$ which satisfies $\omega(\alpha_{i,t}^*, \alpha_{i,t-1|t-1}) \propto q_7(\alpha_{i,t}^*, \alpha_{i,t-1|t-1})$ and $\sum_{i=1}^N \omega(\alpha_{i,t}^*, \alpha_{i,t-1|t-1}) = 1$. From equation (6.67), the jth smoothing random draw $\alpha_{j,t|n}$ is resampled from $\alpha_{1,t}^*, \alpha_{2,t}^*, \cdots, \alpha_{N,t}^*$ with the corresponding probability weights $\omega(\alpha_{1,t}^*, \alpha_{1,t-1|t-1})$, $\omega(\alpha_{2,t}^*, \alpha_{2,t-1|t-1}), \cdots, \omega(\alpha_{N,t}^*, \alpha_{N,t-1|t-1})$. Computing time of the IR smoother based on equation (6.71) is the order of $N \times N''$, while that of the IR smoother with equation (6.70) is $N \times N' \times N''$. Thus, for reduction of computational burden, use of equation (6.71) is superior to that of equation (6.70).

One of the computational techniques is shown as follows. The dimension of Y_t^+ increases as t is small, which implies that $f(Y_t^+|\alpha_{i,t}^*)$ decreases as t goes to the initial time period. Therefore, practically we have some computational difficulties such as underflow errors. To avoid the computational difficulties, we can modify equation (6.69) as: $s_t(\alpha_{i,t}^*) \propto f_y(y_t|\alpha_{i,t}^*) \sum_{j=1}^{N''} s_{t+1}(\alpha_{j,t+1}^*) f_\alpha(\alpha_{j,t+1}^*|\alpha_{i,t}^*) / f_*(\alpha_{j,t+1}^*)$, where $s_t(\alpha_{i,t}^*) \propto f(Y_t^+|\alpha_{i,t}^*)$. For instance, $s_t(\alpha_t)$ may be restricted to $\sum_{i=1}^N s_t(\alpha_{i,t}^*) = 1$ for all t. Note that the proportional relation $q_7(\alpha_{i,t}^*, \alpha_{i,t-1|t-1}) \propto s_t(\alpha_{i,t}^*) f_\alpha(\alpha_{i,t}^*|\alpha_{i,t-1|t-1}) \,/\, f_*(\alpha_{i,t}^*)$ still holds.

References

Anderson, B.D.O. and Moore, J.B., 1979, *Optimal Filtering*, Prentice-Hall, New York.

Aoki, M., 1987, *State Space Modeling of Time Series*, Springer-Verlag.

Aoki, M., 1990, *State Space Modeling of Time Series* (Second, Revised and Enlarged Edition), Springer-Verlag.

Arnold, S.F., 1993, "Gibbs Sampling," in *Handbook of Statistics*, Vol.9, edited by Rao, C.R., pp.599 – 625, North-Holland.

Belsley, D.A., 1973, "On the determination of Systematic Parameter Variation in the Linear Regression Model," *Annals of Economic and Social Measurement*, Vol.2, pp.487 – 494.

Belsley, D.A. and Kuh, E., 1973, "Time-Varying Parameter Structures: An Overview," *Annals of Economic and Social Measurement*, Vol.2, No.4, pp.375 – 379.

Bohachevsky, I.O., Johnson, M.E. and Stein, M.L., 1986, "Generalized Simulated Annealing for Function Optimization," *Technometrics*, Vol.28, No.3, pp.209 – 217.

Bollerslev, T., Engle, R.F. and Nelson, D.B., 1994, "ARCH Models," in *Handbook of Econometrics, Vol.4*, edited by Engle, R.F. and McFadden, D.L., pp.2959 – 3038, North-Holland.

Boswell, M.T., Gore, S.D., Patil, G.P. and Taillie, C., 1993, "The Art of Computer Generation of Random Variables," in *Handbook of Statistics, Vol.9*, edited by Rao, C.R., pp.661 – 721, North-Holland.

Brockwell, P.A. and Davis, R.A., 1987, *Time Series Theory and Models*, Springer-Verlag.

Brooks, S.P. and Morgan, B.J.T., 1995, "Optimization Using Simulated Annealing," *The Statistician*, Vol.44, No.2, pp.241 – 257.

Burmeister, E. and Wall, K.D., 1982, "Kalman Filtering Estimation of Unobserved Rational Expectations with an Application to the German Hyperinflation," *Journal of Econometrics*, Vol.20, pp.255 – 284.

Burridge, P. and Wallis, K.F., 1988, "Prediction Theory for Autoregressive Moving Average Processes," *Econometric Reviews*, Vol.7, No.1, pp.65 – 95.

Carlin, B.P., Polson, N.G. and Stoffer, D.S., 1992, "A Monte Carlo Approach to Non-normal and Nonlinear State Space Modeling," *Journal of the American Statistical Association*, Vol.87, No.418, pp.493 – 500.

Carter, C.K. and Kohn, R., 1994, "On Gibbs Sampling for State Space Models," *Biometrika*, Vol.81, No.3, pp.541 – 553.

Carter, C.K. and Kohn, R., 1996, "Markov Chain Monte Carlo in Conditionally Gaussian State Space Models," *Biometrika*, Vol.83, No.3, pp.589 – 601.

Chib, S. and Greenberg, E., 1995, "Understanding the Metropolis-Hastings Algorithm," *The American Statistician*, Vol.49, No.4, pp.327 – 335.

Chib, S. and Greenberg, E., 1996, "Markov Chain Monte Carlo Simulation Methods in Econometrics," *Econometric Theory*, Vol.12, No.4, pp.409 – 431.

Chow, G.C., 1983, *Econometrics*, McGraw-Hill Book Company.

Conrad, W. and Corrado, C., 1979, "Application of the Kalman Filter to Revisions in Monthly Retail Sales Estimates," *Journal of Economic Dynamic and Control*, Vol.1, pp.177 – 198.

Cooley, T.F., 1977, "Generalized Least Squares Applied to Time Varying Parameter Models: A Comment," *Annals of Economic and Social Measurement*, Vol.6, No.3, pp.313 – 314.

Cooley, T.F. and Prescott, E.C., 1976, "Estimation in the presence of stochastic parameter variation," *Econometrica*, Vol.44, pp.167 – 183.

Cooley, T.F., Rosenberg, B. and Wall, K.D., 1977, "A Note on Optimal Smoothing for Time Varying Coefficient Problems," *Annals of Economic and Social Measurement*, Vol.6, No.4, pp.453 – 456.

Cooper, J.P., 1973, "Time-Varying Regression Coefficients: A Mixed Estimation Approach and Operational Limitations of the General Markov Structure," *Annals of Economic and Social Measurement*, Vol.2, No.4, pp.525 – 530.

Dempster, A.P., Laird, N.M. and Rubin, D.B., 1977, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society*, Ser.B, Vol.39, pp.1 – 38 (with discussion) .

Diebold, F.X. and Nerlove, M., 1989, "Unit Roots in Economic Time Series: A Selective Survey," in *Advances in Econometrics*, Vol.8, pp.3 – 69, JAI Press.

Doucet, A., Godsill, S. and Andrieu, C., 2000, "On Sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, Vol.10, pp.197 – 208.

Engle, R.F., 1982, "Autoregressive Conditional Heteroscedasticity with Estimates of Variance of U.K. Inflation," *Econometrica*, Vol.50, pp.987 – 1008.

Engle, R.F. and Watson, M.W., 1987, "The Kalman Filter: Applications to Forecasting and Rational expectations Models," in *Advances in Econometrics, Fifth World Congress, Vol.I*, Cambridge University Press.

Gardner, G., Harvey, A.C. and Phillips, G.D.A., 1980, "An Algorithm for Maximum Likelihood Estimation Autoregressive-Moving Average Models by means of Kalman Filtering," *Applied Statistics*, Vol.29, No.3, pp.311 – 322.

Gelb, A. (Ed.), 1974, *Applied Optimal Estimation*, MIT Press.

Gelfand, A.E. and Smith, A.F.M., 1990, "Sampling-Based Approaches to Calculating Marginal Densities," *Journal of the American Statistical Association*, Vol.85, No.410, pp.398 – 409.

Geman, S. and Geman D., 1984, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.Pami-6, No.6, pp.721 – 741.

Geweke, J., 1996, "Monte Carlo Simulation and Numerical Integration," in *Handbook of Computational Economics, Vol.1*, edited by Amman, H.M., Kendrick, D.A. and Rust, J., pp.731 – 800, North-Holland.

Geweke, J., 1997, "Posterior Simulators in Econometrics," in *Advances in Economics and Econometrics: Theory and Applications*, Vol.3, edited by Kreps, D. and Wallis, K.F., pp.128 – 165, Cambridge University Press.

Geweke, J. and Tanizaki, H., 1999, "On Markov Chain Monte-Carlo Methods for Nonlinear and Non-Gaussian State-Space Models," *Communications in Statistics, Simulation and Computation*, Vol.28, No.4, pp.867 – 894.

Geweke, J. and Tanizaki, H., 2001, "Bayesian Estimation of State-Space Model Using the Metropolis-Hastings Algorithm within Gibbs Sampling," *Computational Statistics and Data Analysis*, Vol.37, No.2, pp.151-170.

Ghysels, E., Harvey, A.C. and Renault, E., 1996, "Stochastic Volatility," in *Handbook of Statistics, Vol.14*, edited by Maddala, G.S. and Rao, C.R., pp.119 – pp.191, North-Holland.

Goffe, W.L., Ferrier, G.D. and Rogers, J., 1994, "Global Optimization of Statistical Functions with Simulated Annealing," *Journal of Econometrics*, Vol.60, No.1&2, pp.65 – 99.

Gordon, N.J., Salmond, D.J. and Smith, A.F.M., 1993, "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation," *IEE Proceedings-F*, Vol.140, No.2, pp.107 – 113.

Glosten, L.R., Jagannathan, R. and Runkle, D.E., 1993, "On the Relation between the Expected Value and the Volatility of the Nominal Excess Return on Stocks," *The Journal of Finance*, Vol.48, No.5, pp.1779 – 1801.

Hall, R.E., 1978, "Stochastic Implications of the Life Cycle-Permanent Income Hypothesis: Theory and Evidence," *Journal of Political Economy*, Vol.86, No.6, pp.971 – 987.

Hall, R.E. 1990, *The Rational Consumer*, The MIT Press.

Hamilton, J.D., 1989, "A New Approach to the Economic Analysis of Nonstationary Time Series and the Business Cycle," *Econometrica*, Vol.57, pp.357 – 384.

Hamilton, J.D., 1990, "Analysis of Time Series Subject to Changes in Regime," *Journal of Econometrics*, Vol.45, pp.39 – 70.

Hamilton, J.D., 1991, "A Quasi-Bayesian Approach to Estimating Parameters for Mixtures of Normal Distributions," *Journal of Business and Economic Statistics*, Vol.9, pp.27 – 39.

Hamilton, J.D., 1993, "Estimation, Inference and Forecasting of Time Series Subject to Changes in Regime," in *Handbook of Statistics, Vol.11*, edited by Maddala, G.S., Rao, C.R. and Vinod, H.D., pp.231 – 260, North-Holland.

Hamilton, J.D., 1994, *Time Series Analysis*, Princeton University Press.

Hannan, E.J. and Deistler, M., 1988, *The Statistical Theory of Linear System*, John Wiley & Sons.

Härdle, W., 1990, *Applied Nonparametric Regression*, Cambridge University Press.

Harvey, A.C., 1981, *Time Series Models*, Philip Allen Publishers Limited, Oxford.

Harvey, A.C., 1987, "Applications of the Kalman Filter in Econometrics," in *Advances in Econometrics, Fifth World Congress, Vol.I*, Cambridge University Press.

Harvey, A.C., 1989, *Forecasting, Structural Time Series Models and the Kalman Filter*, Cambridge University Press.

Howrey, E.P., 1978, "The Use of Preliminary Data in Econometric Forecasting," *The Review of Economics and Statistics*, Vol.60, pp.193 – 200.

Howrey, E.P., 1984, "Data Revision, Reconstruction, and Prediction: An Application to Inventory Investment," *The Review of Economics and Statistics*, Vol.66, pp.386 – 393.

Hürzeler, M. and Künsch, H.R., 1998, "Monte Carlo Approximations for General State-Space Models," *Journal of Computational and Graphical Statistics*, Vol.7, pp.175 – 193.

Izenman, A.J., 1991, "Recent Developments in Nonparametric Density Estimation," *Journal of the American Statistical Association*, Vol.86, No.413, pp.205 – 224.

Jazwinski, A.H., 1970, *Stochastic Processes and Filtering Theory*, Academic Press, New York.

Kirchen, A., 1988, *Schäzung zeitveränderlicher Strukturparameter in ökonometrischen Prognosemodellen*, Frankfurt/Main: Athenäum.

Kitagawa, G., 1987, "Non-Gaussian State-Space Modeling of Nonstationary Time Series," *Journal of the American Statistical Association*, Vol.82, pp.1032 – 1063 (with discussion).

Kitagawa, G., 1996, "Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State-Space Models," *Journal of Computational and Graphical Statistics*, Vol.5, No.1, pp.1 – 25.

Kitagawa, G., 1998, "A Self-Organizing State-Space Model," *Journal of the American Statistical Association*, Vol.93, No.443, pp.1203 – 1215.

Kitagawa, G. and Gersch, W., 1996, *Smoothness Priors Analysis of Time Series* (Lecture Notes in Statistics, No.116), Springer-Verlag.

Kirkpatrick, S., Gelatt, C.D., Jr. and Vecchi, M.P., 1983, "Optimization by Simulated Annealing," *Science*, Vol.220, No.4598, pp.671 – 680.

Kong, A., Liu, J.S. and Chen, R., 1994, "Sequential Imputations and Bayesian Missing Data Problems," *Journal of the American Statistical Association*, Vol.89, pp.278 – 288.

Kramer, S.C. and Sorenson, H.W., 1988, "Recursive Bayesian Estimation Using Piece-wise Constant Approximations," *Automatica*, Vol.24, No.6, pp.789 – 801.

Laird, N., 1993, "The EM Algorithm," in *Handbook of Statistics, Vol.9*, edited by Rao, C.R., pp.661 – 721, North-Holland.

Liu, J.S., 1996, "Metropolized Independent Sampling with Comparisons to Rejection Sampling and Importance Sampling," *Statistics and Computing*, Vol.6, pp.113 – 119.

Liu, J.S. and Chen, R., 1995, "Blind Deconvolution via Sequential Imputations," *Journal of the American Statistical Association*, Vol.90, pp.567 – 576.

Liu, J.S. and Chen, R., 1998, "Sequential Monte Carlo Methods for Dynamic Systems," *Journal of the American Statistical Association*, Vol.93, pp.1032 – 1044.

Liu, J.S., Chen, R. and Wong, W.H.G., 1998, "Rejection Control and Sequential Importance Sampling," *Journal of the American Statistical Association*, Vol.93, pp.1022 – 1031.

Mariano, R.S. and Tanizaki, H., 1995, "Prediction of Final Data with Use of Preliminary and/or Revised Data," *Journal of Forecasting*, Vol.14, No.4, pp.351 – 380.

Mariano, R.S. and Tanizaki, H., 2000, "Simulation-Based Inference in Nonlinear State-Space Models: Application to Testing Permanent Income Hypothesis," in *Simulation-Based Inference in Econometrics: Methods and Applications*,

Chap.9, edited by Mariano, R.S., Weeks, M. and Schuermann, T., pp.218 – 234, Cambridge University Press.

McNelis, P.D. and Neftci, S.N., 1983, "Policy-dependent Parameters in the Presence of Optimal Learning: An Application of Kalman Filtering to the Fair and Sargent Supply-side Equations," *The Review of Economics and Statistics*, Vol.65, pp.296 – 306.

Nelson, D.B., 1991, "Conditional Heteroskedasticity in Asset Returns: A New Approach," *Econometrica*, Vol.59, No.2, pp. 347-370.

Nicholls, D.F. and Pagan, A.R., 1985, "Varying Coefficient Regression," in *Handbook of Statistics, Vol.5*, edited by Hannan, E.J., Krishnaiah, P.R. and Rao, M.M., pp.413 – 449, North-Holland.

O'Hagan, A., 1994, *Kendall's Advanced Theory of Statistics*, Vol.2B (Bayesian Inference), Edward Arnold.

Pagan, A.R., 1975, "A Note on the Extraction of Components from Time Series," *Econometrica*, Vol.43, pp.163 – 168.

Prakasa Rao, B.L.S., 1983, *Nonparametric Functional Estimation*, Academic Press.

Rund, P.A., 1991, "Extensions of Estimation Methods Using the EM Algorithm," *Journal of Econometrics*, Vol.49, pp.305 – 341.

Sant, D.T., 1977, "Generalized Least Squares Applied to Time Varying Parameter Models," *Annals of Economic and Measurement*, Vol.6, No.3, pp.301 – 311.

Sarris, A.H., 1973, "A Bayesian Approach to Estimation of Time Varying Regression Coefficients," *Annals of Economic and Social Measurement*, Vol.2, No.4, pp.501 – 523.

Shumway, R.H. and Stoffer, D.S., 1982, "An Approach to Time Series Smoothing and Forecasting Using the EM Algorithm," *Journal of Time Series Analysis*, Vol.3, PP.253 – 264.

Silverman, B.W., 1986, *Density Estimation for Statistics and Data Analysis* (Monographs on Statistics and Applied Probability 26), Chapman & Hall.

Smith, A.F.M. and Roberts, G.O., 1993, "Bayesian Computation via Gibbs Sampler and Related Markov Chain Monte Carlo Methods," *Journal of the Royal Statistical Society*, Ser.B, Vol.55, No.1, pp.3 – 23.

Tanizaki, H., 1989, "The Kalman Filter Model under the Assumption of the First-Order Autoregressive Process in the Disturbance Terms," *Economics Letters*, Vol.31, No.2, pp.145 – 149.

Tanizaki, H., 1993a, "Kalman Filter Model with Qualitative Dependent Variable," *The Review of Economics and Statistics*, Vol.75, No.4, pp.747 – 752.

Tanizaki, H., 1993b, *Nonlinear Filters: Estimation and Applications* (Lecture Notes in Economics and Mathematical Systems, No.400), Springer-Verlag.

Tanizaki, H., 1996, *Nonlinear Filters: Estimation and Applications* (Second, Revised and Enlarged Edition), Springer-Verlag.

Tanizaki, H., 1997, "Nonlinear and Nonnormal Filters Using Monte Carlo Methods," *Computational Statistics and Data Analysis*, Vol.25, No.4, pp.417 – 439.

Tanizaki, H., 1999, "On the Nonlinear and Nonnormal Filter Using Rejection Sampling," *IEEE Transactions on Automatic Control*, Vol.44, No.2, pp.314 – 319.

Tanizaki, H., 2000, "Time-Varying Parameter Model Revisited," *Kobe University Economic Review*, Vol.45, pp.41 – 57.

Tanizaki, H., 2001a, "Nonlinear and Non-Gaussian State Space Modeling Using Sampling Techniques," *Annals of the Institute of Statistical Mathematics*, Vol.53, No.1, pp.63 – 81.

Tanizaki, H., 2001b, "Estimation of Unknown Parameters in Nonlinear and Non-Gaussian State Space Models," *Journal of Statistical Planning and Inference*, Vol.96, No.2, pp.301 – 323.

Tanizaki, H., 2003, "Nonlinear and Non-Gaussian State-Space Modeling with Monte Carlo Techniques: A Survey and Comparative Study," in *Handbook of Statistics, Vol.21 (Stochastic Processes: Theory and Applications)*, Chap.22, edited by Rao, C.R. and Shanbhag, D.N., pp.871 – 929, North-Holland.

Tanizaki, H. and Mariano, R.S., 1994, "Prediction, Filtering and Smoothing in Nonlinear and Nonnormal Cases Using Monte-Carlo Integration," *Journal of Applied Econometrics*, Vol.9, No.2, pp.163 – 179 (in *Econometric Inference Using Simulation Techniques*, Chap.12, edited by van DijK, H.K., Manfort, A. and Brown, B.W., pp.245 – 261, 1995, John Wiley & Sons).

Tanizaki, H. and Mariano, R.S., 1996, "Nonlinear Filters Based on Taylor Series Expansions," *Communications in Statistics, Theory and Methods*, Vol.25, No.6, pp.1261 – 1282.

Tanizaki, H. and Mariano, R.S., 1998, "Nonlinear and Non-Gaussian State-Space Modeling with Monte Carlo Simulations," *Journal of Econometrics*, Vol.83, No.1&2, pp.263 – 290.

Tanner, M.A. and Wong, W.H., 1987, "The Calculation of Posterior Distributions by Data Augmentation," *Journal of the American Statistical Association*, Vol.82, No.398, pp.528 – 550 (with discussion).

Taylor, S.J., 1994, "Modeling Stochastic Volatility: A Review and Comparative Study," *Mathematical Finance*, Vol.4, No.2, pp.183 – 204.

Tierney, L., 1994, "Markov Chains for Exploring Posterior Distributions," *The Annals of Statistics*, Vol.22, No.4, pp.1701 – 1762.

Ullah, A., 1988, "Non-parametric Estimation of Econometric Functionals," *Canadian Journal of Economics*, Vol.21, No.3, pp.625 – 658.

Watanabe, T., 1999, "A Non-linear Filtering Approach to Stochastic Volatility Models with an Application to Daily Stock Returns," *Journal of Applied Econometrics*, Vol.14, No.2, pp.101 – 121.

Watanabe, T., 2000a, "Bayesian Analysis of Dynamic Bivariate Mixture Models: Can They Explain the Behavior of Returns and Trading Volume?" *Journal of Business and Economic Statistics*, Vol.18, No.2, pp.199 – 210.

Watanabe, T., 2000b, "Excess Kurtosis of Conditional Distribution for Daily Stock Returns: The Case of Japan," *Applied Economics Letters*, Vol.7, No.6, pp.353 – 355.

Part III

Nonparametric Statistical Methods

Chapter 7

Difference between Two-Sample Means

This chapter is based on Tanizaki (1997), which is substantially revised. Nonparametric tests dealing with two samples include score tests (e.g., Wilcoxon rank sum test, normal score test, logistic score test and so on) and Fisher's randomization test. Since in general the nonparametric tests require a large amount of computational burden, there are few studies on small sample properties although asymptotic properties from various aspects were studied in the past. In this chapter, the various nonparametric tests dealing with difference between two-sample means are compared with the *t* test, which is a conventional parametric test, through Monte Carlo experiments. Also, we consider testing structural changes as an application to the regression analysis.

7.1 Introduction

There are various kinds of **nonparametric test**s (**distribution-free test**s), i.e., **score test**s, Fisher's test and so on. However, almost all the studies in the past are related to asymptotic properties. In this chapter, we examine small sample properties of nonparametric two-sample tests by Monte Carlo experiments.

One of the features of nonparametric tests is that we do not have to impose any assumption on the underlying distribution. From no restriction on the distribution, it can be expected that the nonparametric tests are less powerful than the conventional **parametric test**s such as *t* test. However, Hodges and Lehmann (1956) and Chernoff and Savage (1958) showed that the **Wilcoxon rank sum test** is as powerful as the *t* test under the location-shift alternatives and moreover that the Wilcoxon test is sometimes much more powerful than the *t* test. Especially, the remarkable fact about the Wilcoxon test is that it is about 95 percent as powerful as the usual *t* test for normal data, which is discussed in Section 7.3. Chernoff and Savage (1958) proved that **Pitman's asymptotic relative efficiency** of the **normal score test** relative to the *t* test is greater than one under the location-shift alternatives (see Section 7.3 for Pitman's

asymptotic relative efficiency). This implies that except for normal population the power of the normal score test is always larger than that of the *t* test. According to Mehta and Patel (1992), the normal score test is less powerful than the Wilcoxon test if the tails of the underlying distributions are diffuse.

Fisher's randomization test examines whether there is difference between two-sample means. The *t* test, which is a parametric test, is also used when we test whether two-sample means are different. Bradley (1968) showed that the *t* test does not depend on the functional form of the underlying distribution in a large sample if there exists the fourth moment, which implies that the *t* test is asymptotically a nonparametric test. The score tests are similar to the Fisher test except that the score test statistics are based on the sum of score while the Fisher test statistic is difference between two-sample means. Both test statistics are discretely distributed and we have to obtain all the possible combinations for testing.

It is quite difficult to obtain all the possible combinations, and accordingly computational time becomes quite large. Mehta and Patel (1983, 1986a, 1986b), Mehta, Patel and Tsiatis (1984), Mehta, Patel and Gray (1985), Mehta, Patel and Wei (1988) made a program on the Fisher permutation test (a generalization of the Fisher two-sample test treated in this chapter, i.e., independence test by $r \times c$ contingency table) using a network algorithm. *StatXact* is a computer software on nonparametric inference, which computes the exact probability using a nonparametric test. There, the network algorithm that Mehta and Patel (1983, 1986a, 1986b), Mehta, Patel and Tsiatis (1984), Mehta, Patel and Gray (1985), Mehta, Patel and Wei (1988) developed is used for a permutation program. The two source codes which obtain all the possible combinations are introduced in Appendix 7.1, where the source code are written by C language because binary operation and recursion are utilized.

In this chapter, we consider small sample properties of two-sample nonparametric tests (i.e., the score tests and the Fisher test) by comparing with the *t* test which is the usual parametric test.

7.2 Overview of Nonparametric Tests

It is well known for testing two-sample means that the *t* test gives us a uniform powerful test under normality assumption but not under nonnormality. We consider a distribution-free test in this chapter, which is also called a nonparametric test. Normal score Test, Wilcoxon (1945) rank sum test, and Fisher (1935) test are famous nonparametric tests, which are similar tests. We have two-sample groups. We test if two samples are generated from the same distribution. Let $x_1, x_2, \cdots, x_{n1}$ be mutually independently distributed as $F(x)$, and $y_1, y_2, \cdots, y_{n2}$ be mutually independently distributed as $G(x)$, where $n1$ denotes the sample size of Group 1 and $n2$ is given by that of Group 2. $F(x)$ and $G(x)$ are continuous distribution functions. For simplicity of discussion, in this chapter all the values of X_i and Y_j are assumed to be different. Under the assumptions, we consider the null hypothesis of no difference between two-sample

means. The null hypothesis H_0 is represented by:

$$H_0 \ : \ F(x) = G(x).$$

Both the score tests and the Fisher test can be applied under the alternative of location shift. For the nonparametric test in the case where we test if the functional form of the two distributions is different, we have the runs test (Kendall and Stuart (1979)), which is not discussed in this book. One possible alternative hypothesis H_1 is given by:

$$H_1 \ : \ F(x) = G(x - \theta), \qquad \theta > 0,$$

where a shift of the location parameter θ is tested.

We consider randomly taking $n1$ samples out of N samples, mixing two groups, where $N = n1 + n2$ is defined. Then, we have ${}_NC_{n1}$ combinations. Each event of ${}_NC_{n1}$ combinations occurs with equal probability $1/{}_NC_{n1}$. For both the score tests and the Fisher test, all the possible combinations are compared with the original two samples.

7.2.1 Score Tests

For the score tests, the two samples $\{x_i\}_{i=1}^{n1}$ and $\{y_j\}_{j=1}^{n2}$ are converted into the data ranked by size. Let $\{Rx_i\}_{i=1}^{n1}$ and $\{Ry_j\}_{j=1}^{n2}$ be the ranked samples corresponding to $\{x_i\}_{i=1}^{n1}$ and $\{y_j\}_{j=1}^{n2}$, respectively. That is, for all i, Rx_i takes one of the integers from 1 to $n1 + n2$. The score test statistic s_0 is represented by:

$$s_0 = \sum_{i=1}^{n1} a(Rx_i), \tag{7.1}$$

where $a(\cdot)$ is a function to be specified, which is called the **score function**.

For all the possible combinations of taking $n1$ samples out of N samples (i.e., ${}_NC_{n1}$ combinations), we can compute the sum of the scores shown in equation (7.1). Then, we have ${}_NC_{n1}$ combinations. Let the sum of the scores be s_m, $m = 1, 2, \cdots, {}_NC_{n1}$. Note that at least one of s_m, $m = 1, 2, \cdots, {}_NC_{n1}$, is equal to s_0. s_m occurs with equal probability (i.e., $1/{}_NC_{n1}$) for all the combinations. Comparing s_0 and s_m, the following probabilities can be computed by counting:

$$P(s < s_0) = \frac{\text{the number of } s_m \text{ which satisfies } s_m < s_0,\ m = 1, 2, \cdots, {}_NC_{n1}}{{}_NC_{n1}},$$
$$P(s = s_0) = \frac{\text{the number of } s_m \text{ which satisfies } s_m = s_0,\ m = 1, 2, \cdots, {}_NC_{n1}}{{}_NC_{n1}},$$
$$P(s > s_0) = \frac{\text{the number of } s_m \text{ which satisfies } s_m > s_0,\ m = 1, 2, \cdots, {}_NC_{n1}}{{}_NC_{n1}},$$

where s is taken as a random variable generated from the score test statistic.

If $P(s < s_0)$ is small enough, s_0 is located at the right tail of the distribution, which implies $F(x) < G(x)$ for all x. Similarly, if $P(s > s_0)$ is small enough, s_0 is located at

the left tail of the distribution, which implies $F(x) > G(x)$ for all x. Therefore, in the case of the null hypothesis $H_0 : F(x) = G(x)$ and the alternative $H_1 : F(x) \neq G(x)$, the null hypothesis is rejected at the 10% significance level when $P(s < s_0) \leq 0.05$ or $P(s > s_0) \leq 0.05$.

We can consider various score tests by specifying the function $a(\cdot)$. The score tests examined in this chapter are Wilcoxon rank sum test, normal score test, logistic score test, and Cauchy score test.

Wilcoxon Rank Sum Test: One of the most famous nonparametric tests is the Wilcoxon rank sum test. Wilcoxon test statistic w_0 is the score test defined as $a(Rx_i) = Rx_i$, which is as follows:

$$w_0 = \sum_{i=1}^{n1} Rx_i.$$

$P(w < w_0)$, $P(w = w_0)$ and $P(w > w_0)$ are computed, where w denotes a random variable of the Wilcoxon test statistic.

In the past, it was too difficult to obtain the exact distribution of w, from computational point of view. Therefore, under the null hypothesis, we have tested utilizing the fact that w has approximately normal distribution with mean $\mathrm{E}(w)$ and variance $\mathrm{V}(w)$:

$$\mathrm{E}(w) = \frac{n1\ (N+1)}{2}, \qquad \mathrm{V}(w) = \frac{n1\ (N-n1)(N+1)}{12}.$$

Accordingly, in this case, the following statistic was used for the Wilcoxon test statistic:

$$aw = \frac{w - \mathrm{E}(w)}{\sqrt{\mathrm{V}(w)}},$$

which is called the **asymptotic Wilcoxon test** statistic in this chapter. aw is asymptotically distributed as a standard normal random variable. Mann and Whitney (1947) demonstrated that the normal approximation is quite accurate when $n1$ and $n2$ are larger than 7 (see Mood, Graybill and Boes (1974)).

Hodges and Lehmann (1956) showed that Pitman's asymptotic relative efficiency of the Wilcoxon test relative to the t test is quite good. They obtained the result that the asymptotic relative efficiency is always greater than or equal to 0.864 under the null hypothesis of location shift. This result implies that the Wilcoxon test is not too poor, compared with the t test, and moreover that the Wilcoxon test may be sometimes much better than the t test. Especially, they showed that the relative efficiency of the Wilcoxon test is 1.33 when the density function $f(x)$ takes the following:

$$f(x) = \frac{x^2 \exp(-x)}{\Gamma(3)},$$

where $\Gamma(3)$ is a gamma function with parameter 3. In general, for the distributions with large tails, the Wilcoxon test is more powerful than the t test. See Section 7.3 for the asymptotic relative efficiency.

All the past studies are concerned with asymptotic properties. In Section 7.4, we examine the small sample cases, i.e., $n1 = n2 = 9, 12, 15$.

Normal Score Test: The normal score test statistic ns_0 is:

$$ns_0 = \sum_{i=1}^{n1} \Phi^{-1}\Big(\frac{Rx_i - 0.5}{N}\Big),$$

where $\Phi(\cdot)$ is a standard normal distribution. 0.5 in the numerator of p_i indicates the continuity correction. Thus, $P(ns < ns_0)$, $P(ns = ns_0)$ and $P(ns > ns_0)$ are obtained, where ns denotes a random variable of the normal score test statistic.

The score test that $a(\cdot)$ in equation (7.1) is assumed to be $a(x) = \Phi^{-1}\Big(\frac{x - 0.5}{N}\Big)$ is called the normal score test. We can interpret the Wilcoxon test as the score test assumed to be a uniform distribution for the inverse function of $a(\cdot)$. That is, the score test defined as $a(x) = (x - 0.5)/N$ is equivalent to the Wilcoxon test, so-called the **uniform score test**.

Chernoff and Savage (1958) proved that the asymptotic relative efficiency of the normal score test relative to the t test is greater than or equal to one, i.e., that the power of the normal score test is equivalent to that of the t test under normality assumption and the power of the normal score test is greater than that of the t test otherwise.

Logistic Score Test: The logistic score test statistic ls_0 is given by:

$$ls_0 = \sum_{i=1}^{n1} F^{-1}\Big(\frac{Rx_i - 0.5}{N}\Big),$$

where $F(x) = \dfrac{1}{1 + e^{-x}}$, which is a logistic distribution. Again, $P(ls < ls_0)$, $P(ls = ls_0)$ and $P(ls > ls_0)$ are obtained, where ls denotes a random variable of the logistic score test statistic.

Cauchy Score Test: The Cauchy score test statistic cs_0 is represented as:

$$cs_0 = \sum_{i=1}^{n1} F^{-1}\Big(\frac{Rx_i - 0.5}{N}\Big),$$

where $F(x) = 1/2 + (1/\pi)\tan^{-1} x$, which is a Cauchy distribution. $P(cs < cs_0)$, $P(cs = cs_0)$ and $P(cs > cs_0)$ are computed, where cs denotes a random variable of the Cauchy score test statistic.

Thus, by specifying a functional form for $a(\cdot)$, various score tests can be constructed. In this chapter, however, the four score tests discussed above and the Fisher test introduced in the following section are compared.

7.2.2 Fisher's Randomization Test

While the Wilcoxon test statistic is the rank sum of the two samples, the Fisher test statistic uses difference between two-sample means, i.e., $\bar{x} - \bar{y}$ for the two samples $\{x_i\}_{i=1}^{n1}$ and $\{y_j\}_{j=1}^{n2}$. Thus, the test statistic is given by:

$$f_0 = \bar{x} - \bar{y}.$$

Again, we consider mixing two groups, taking $n1$ samples randomly out of N samples, and dividing N data into two groups, where $N = n1+n2$ is defined,. Then, we have ${}_NC_{n1}$ combinations. For all the possible combinations (i.e., ${}_NC_{n1}$ combinations taking $n1$ out of N), we compute difference between the sample means. Let f_m be the difference between two-sample means, obtained from the mth combination. Note that at least one out of f_m, $m = 1, 2, \cdots, {}_NC_{n1}$, is equal to f_0. For all m, f_m occurs with equal probability (i.e., $1/{}_NC_{n1}$). Comparing f_0 and f_m, we can compute $P(f < f_0)$, $P(f = f_0)$ and $P(f > f_0)$, where f is a random variable generated from the Fisher test statistic.

Fisher's two-sample test is the same type as the score tests in the sense that all the possible combinations are utilized for testing, but the Fisher test uses more information than the score tests because the score tests utilize the ranked data as the test statistics while the Fisher test uses the original data. It might be expected that the Fisher test is more powerful than the score tests. Moreover, Bradley (1968) stated that the Fisher test and the t test are asymptotically equivalent because both of them use the difference between two-sample means as the test statistic. See Appendix 7.2 for asymptotic equivalence between Fisher and t tests. Hoeffding (1952) showed that the Fisher test is asymptotically as powerful as the t test even under normality assumption. Therefore, it can be shown that the asymptotic relative efficiency of the Fisher test is sometimes better or worse, compared with the Wilcoxon test.

As mentioned above, the Fisher test statistic is given by the difference between two-sample means, which is rewritten as:

$$\begin{aligned} f_0 &= \bar{x} - \bar{y} \\ &= \left(\frac{1}{n1} + \frac{1}{n2}\right) \sum_{i=1}^{n1} x_i - \frac{1}{n2}\left(\sum_{i=1}^{n1} x_i + \sum_{j=1}^{n2} y_j\right). \end{aligned}$$

This implies that the Fisher test statistic is equivalent to $\sum_{i=1}^{n1} x_i$, because the second term in the second line is a sum of all the data and accordingly it is independent of choosing $n1$ data out of all the N data. Thus, the Fisher test statistic is given by $\sum_{i=1}^{n1} x_i$, which is the sum of Group 1 data, while the score test statistic is represented by $\sum_{i=1}^{n1} a(Rx_i)$, which is the sum of a function of Group 1 ranked data.

7.3 Asymptotic Relative Efficiency

Pitman's asymptotic relative efficiency is defined as follows. We consider two kinds of testing procedures, i.e., Tests 1 and 2. Let $N_1 = n1_1+n2_1$ be the sample size required to obtain the same power as another test for the sample size $N_2 = n1_2+n2_2$. The subscript in N, $n1$ and $n2$ represents Test 1 or 2. Then, the limit of N_2/N_1 is called Pitman's asymptotic relative efficiency of Test 1 relative to test 2 (see, for example, Kendall and Stuart (1979)), where $n1_2/N_2 = n1_1/N_1$ and $n2_2/N_2 = n2_1/N_1$ are assumed.

7.3.1 Score Test

In this section, we consider the asymptotic relative efficiency of score tests.

Let $X_1, X_2, \cdots, X_{n1}$ be mutually independently distributed random variables with the density function $f(x)$ and $Y_1, Y_2, \cdots, Y_{n2}$ be mutually independently distributed random variables with the density function $g(y)$. Consider the null hypothesis $f(z) = g(z)$ against the alternative one $g(z) \neq f(z)$.

Let Rx_i be the ranked data of X_i. As shown in the previous section, the score test statistic is given by:

$$s = \sum_{i=1}^{n1} a(Rx_i),$$

where $a(\cdot)$ is called the score function, which is specified as the inverse of a cumulative distribution function.

To obtain the asymptotic relative efficiency, the above problem is reformulated as follows. We sort $X_1, X_2, \cdots, X_{n1}$ and $Y_1, Y_2, \cdots, Y_{n2}$ by size. Denote the ordered sample by Z_1, Z_2, $\cdots$, Z_N, where $Z_1 < Z_2 < \cdots < Z_N$ holds. That is, Z_i is equivalent to one of $X_1, X_2, \cdots, X_{n1}$ and $Y_1, Y_2, \cdots, Y_{n2}$ for all $i = 1, 2, \cdots, N$. Then, the test statistic s is rewritten as:

$$s = \sum_{i=1}^{N} a(Rz_i)\epsilon_i,$$

where Rz_i denotes the ranked data of Z_i and ϵ_i is given by:

$$\epsilon_i = \begin{cases} 1, & \text{if } Z_i \text{ is in Group 1 sample, i.e., if } Z_i \text{ is one of } X_1, X_2, \cdots, X_{n1}, \\ 0, & \text{if } Z_i \text{ is in Group 2 sample, i.e., if } Z_i \text{ is one of } Y_1, Y_2, \cdots, Y_{n2}. \end{cases}$$

Moreover, p_i is defined as:

$$p_i = \frac{Rz_i - 0.5}{N},$$

for $i = 1, 2, \cdots, N$. p_i is between zero and one. Note that 0.5 in the numerator of p_i implies the continuity correction. Define the following function $c(\cdot)$ which satisfies:

$$c(p_i) = c\Big(\frac{Rz_i - 0.5}{N}\Big) = a(Rz_i).$$

Using $c(p_i)$, the test statistic s is represented as:

$$s = \sum_{i=1}^{N} c(p_i)\epsilon_i. \tag{7.2}$$

Based on the test statistic (7.2), hereafter we obtain the asymptotic distribution of s.

Asymptotic Distribution under the Null Hypothesis: First, we obtain the asymptotic distribution of s under $H_0 : f(z) = g(z)$.

To derive the mean and variance of s, the mean and variance of ϵ_i and the covariance between ϵ_i and ϵ_j have to be obtained. Under the null hypothesis $f(z) = g(z)$, consider choosing the $n1$ data out of the N data. In this case, as mentioned above, there are ${}_NC_{n1}$ combinations. Therefore, the probability which ϵ_i takes 1 is given by: $P(\epsilon_i = 1) = {}_{N-1}C_{n1-1}/{}_NC_{n1} = n1/N$ for all $i = 1, 2, \cdots, N$, and the probability which ϵ_i takes 0 is: $P(\epsilon_i = 0) = 1 - P(\epsilon_i = 1) = n2/N$ for $i = 1, 2, \cdots, N$. Thus, the expectation of ϵ_i is:

$$\mathrm{E}(\epsilon_i) = 1 \times P(\epsilon_i = 1) + 0 \times P(\epsilon_i = 0) = \frac{n1}{N},$$

for all $i = 1, 2, \cdots, N$. Similarly, $\mathrm{E}(\epsilon_i^2)$ is computed as:

$$\mathrm{E}(\epsilon_i^2) = 1^2 \times P(\epsilon_i = 1) + 0^2 \times P(\epsilon_i = 0) = \frac{n1}{N},$$

for $i = 1, 2, \cdots, N$. Moreover, the covariance between ϵ_i and ϵ_j for $i \neq j$ is given by:

$$\begin{aligned}\mathrm{E}(\epsilon_i\epsilon_j) &= 1 \times 1 \times P(\epsilon_i = 1, \epsilon_j = 1) + 1 \times 0 \times P(\epsilon_i = 1, \epsilon_j = 0) \\ &\quad + 0 \times 1 \times P(\epsilon_i = 0, \epsilon_j = 1) + 0 \times 0 \times P(\epsilon_i = 0, \epsilon_j = 0) \\ &= 1 \times 1 \times P(\epsilon_i = 1, \epsilon_j = 1) = \frac{{}_{N-2}C_{n1-2}}{{}_NC_{n1}} = \frac{n1\ (n1-1)}{N(N-1)},\end{aligned}$$

for all $i \neq j$. The number of combinations which both Z_i and Z_j belong to Group 1 is given by ${}_{N-2}C_{n1-2}$.

Accordingly, mean of s is derived as follows:

$$\mathrm{E}(s) = \mathrm{E}(\sum_{i=1}^{N} c(p_i)\epsilon_i) = \sum_{i=1}^{N} c(p_i)\mathrm{E}(\epsilon_i) = \frac{n1}{N}\sum_{i=1}^{N} c(p_i) = n1\ \bar{c},$$

where $\bar{c} \equiv (1/N)\sum_{i=1}^{N} c(p_i)$ is defined. $\mathrm{E}(s^2)$ is:

$$\begin{aligned}\mathrm{E}(s^2) &= \mathrm{E}\Big((\sum_{i=1}^{N} c(p_i)\epsilon_i)^2\Big) = \mathrm{E}\Big((\sum_{i=1}^{N} c(p_i)\epsilon_i)(\sum_{j=1}^{N} c(p_j)\epsilon_j)\Big) \\ &= \mathrm{E}\Big(\sum_{i=1}^{N}\sum_{j=1}^{N} c(p_i)c(p_j)\epsilon_i\epsilon_j\Big) = \mathrm{E}\Big(\sum_{i=1}^{N} c(p_i)^2\epsilon_i^2\Big) + \mathrm{E}\Big(\sum_{\substack{i=1 \\ i\neq j}}^{N}\sum_{j=1}^{N} c(p_i)c(p_j)\epsilon_i\epsilon_j\Big)\end{aligned}$$

$$
\begin{aligned}
&= \sum_{i=1}^{N} c(p_i)^2 \mathrm{E}(\epsilon_i^2) + \sum_{\substack{i=1 \\ i \neq j}}^{N} \sum_{j=1}^{N} c(p_i) c(p_j) \mathrm{E}(\epsilon_i \epsilon_j) \\
&= \frac{n1}{N} \sum_{i=1}^{N} c(p_i)^2 + \frac{n1\,(n1-1)}{N(N-1)} \sum_{\substack{i=1 \\ i \neq j}}^{N} \sum_{j=1}^{N} c(p_i) c(p_j) \\
&= \frac{n1}{N} \sum_{i=1}^{N} c(p_i)^2 - \frac{n1\,(n1-1)}{N(N-1)} \sum_{i=1}^{N} c(p_i)^2 + \frac{n1\,(n1-1)}{N(N-1)} \sum_{i=1}^{N} \sum_{j=1}^{N} c(p_i) c(p_j) \\
&= \frac{n1\,(N-n1)}{N(N-1)} \sum_{i=1}^{N} c(p_i)^2 + \frac{n1\,(n1-1)}{N(N-1)} \sum_{i=1}^{N} \sum_{j=1}^{N} c(p_i) c(p_j) \\
&= \frac{n1\,(N-n1)}{N(N-1)} \sum_{i=1}^{N} c(p_i)^2 + \frac{n1\,(n1-1)N}{N-1} \overline{c}^2.
\end{aligned}
$$

Note that we have the fourth line, substituting $\mathrm{E}(\epsilon_i^2)$ and $\mathrm{E}(\epsilon_i \epsilon_j)$ in the third line. The second term in the fourth line is equal to the second and third terms in the fifth line. Moreover, the first term in the sixth line is equivalent to the first and second terms in the fifth line. The second term in the sixth line is also equal to the second term in the seventh line, because $\sum_{i=1}^{N} \sum_{j=1}^{N} c(p_i) c(p_j) = \left(\sum_{i=1}^{N} c(p_i)\right)\left(\sum_{j=1}^{N} c(p_j)\right)$ and $\overline{c} = (1/N) \sum_{i=1}^{N} c(p_i)$.

Utilizing the first and second moments of s, i.e., $\mathrm{E}(s)$ and $\mathrm{E}(s^2)$, variance of s is obtained as:

$$
\begin{aligned}
\mathrm{V}(s) = \mathrm{E}(s^2) - (\mathrm{E}(s))^2 &= \frac{n1\,(N-n1)}{N(N-1)} \sum_{i=1}^{N} c(p_i)^2 + \frac{n1\,(n1-1)N}{N-1} \overline{c}^2 - (n1\,\overline{c})^2 \\
&= \frac{n1\,(N-n1)}{(N-1)} \frac{1}{N} \sum_{i=1}^{N} c(p_i)^2 - \frac{n1\,(N-n1)}{N-1} \overline{c}^2 \\
&= \frac{n1\,(N-n1)}{N(N-1)} \sum_{i=1}^{N} (c(p_i)^2 - \overline{c})^2 = \frac{n1\,(N-n1)}{N} \sigma_c^2,
\end{aligned}
$$

where $\sigma_c^2 \equiv \sum_{i=1}^{N} (c(p_i)^2 - \overline{c})^2 / (N-1)$ is defined.

Define $\overline{s} = s/N$, $n1 = \omega N$ and $n2 = N - n1 = (1-\omega)N$, where ω is assumed to be constant. Then, the mean and variance of $\overline{s}$ is derived as:

$$
\mathrm{E}(\overline{s}) = \omega \overline{c}, \qquad \mathrm{V}(\overline{s}) = \frac{\omega(1-\omega)}{N} \sigma_c^2.
$$

Therefore, by the central limit theorem, as N goes to infinity we have the following result:

$$
\frac{\overline{s} - \omega \overline{c}}{\sqrt{\omega(1-\omega)\sigma_c^2/N}} = \frac{(1/N) \sum_{i=1}^{N} a(Rz_i)\epsilon_i - \omega \overline{c}}{\sqrt{\omega(1-\omega)\sigma_c^2/N}} \longrightarrow N(0,1). \tag{7.3}
$$

In addition, for sufficiently large N, we have the following properties:

$$\overline{c} \equiv \frac{1}{N}\sum_{i=1}^{N} c(p_i) \longrightarrow \int_0^1 c(t)\,\mathrm{d}t,$$
$$\sigma_c^2 \equiv \frac{1}{N-1}\sum_{i=1}^{N}(c(p_i)-\overline{c})^2 \longrightarrow \int_0^1 c(t)^2\,\mathrm{d}t - \Big(\int_0^1 c(t)\,\mathrm{d}t\Big)^2.$$

Note that the score function is defined on the interval between zero and one and that it is assumed to be integrable on the interval. Remember that $p_i = (Rz_i - 0.5)/N$ is between zero and one, where Rz_i indicates the ranked data corresponding to Z_i. Thus, under the null hypothesis $H_0 : g(z) = f(z)$, the score test statistic (7.2) is asymptotically distributed as (7.3).

Asymptotic Distribution under the Alternative Hypothesis: Under the alternative hypothesis $H_1 : g(z) \neq f(z)$, especially $g(z) = f(z-\theta)$, we consider the conditional distribution of $\epsilon_1, \epsilon_2, \cdots, \epsilon_N$ given $Z_1, Z_2, \cdots, Z_N$. When N is large, $\epsilon_1, \epsilon_2, \cdots, \epsilon_N$ given $Z_1, Z_2, \cdots, Z_N$ are approximated to be mutually independently distributed, because the correlation coefficient between ϵ_i and ϵ_j is given by $-1/(N-1)$ and it goes to zero as N is large. Remember that the correlation coefficient is defined as: $\mathrm{Cov}(\epsilon_i, \epsilon_j)/\sqrt{\mathrm{V}(\epsilon_i)\mathrm{V}(\epsilon_j)}$, where $\mathrm{V}(\epsilon_i) = \mathrm{E}(\epsilon_i^2) - (\mathrm{E}(\epsilon_i)) = \dfrac{n1\,(N-n1)}{N^2}$ and $\mathrm{Cov}(\epsilon_i, \epsilon_j) = \mathrm{E}(\epsilon_i\epsilon_j) - \mathrm{E}(\epsilon_i)\mathrm{E}(\epsilon_j) = -\dfrac{n1\,(N-n1)}{N^2(N-1)}$. Under the alternative hypothesis $g(z) \neq f(z)$, the conditional distribution of ϵ_i given $Z_1 = z_1, Z_2 = z_2, \cdots, Z_N = z_N$ is given by:

$$P(\epsilon_i = 1|z_1, z_2, \cdots, z_N) = P(\epsilon_i = 1|z_i) = \frac{\omega f(z_i)}{\omega f(z_i) + (1-\omega)g(z_i)}.$$

Note that the density function of z_i is given by: $\omega f(z_i) + (1-\omega)g(z_i)$, which implies that z_i is generated from $f(\cdot)$ with probability ω and $g(\cdot)$ with probability $1-\omega$. As for a joint probability of ϵ_i and Z_i, we have $P(\epsilon_i = 1, Z_i < z_i) = \omega\int_{-\infty}^{z_i} f(t)\,\mathrm{d}t$ and $P(\epsilon_i = 0, Z_i < z_i) = (1-\omega)\int_{-\infty}^{z_i} g(t)\,\mathrm{d}t$, which implies that $P(\epsilon_i = 1, z_i) = \omega f(z_i)$ and $P(\epsilon_i = 0, z_i) = (1-\omega)g(z_i)$. Therefore, we can obtain $P(\epsilon_i = 1|z_i)$ as shown above.

Now, consider $g(z) = f(z-\theta)$ for small θ. Then, $\mathrm{E}(\epsilon_i|z_1, z_2, \cdots, z_N)$ is obtained as follows:

$$\begin{aligned}
\mathrm{E}(\epsilon_i|z_1, z_2, \cdots, z_N) &= 1 \times P(\epsilon_i = 1|z_1, z_2, \cdots, z_N) + 0 \times P(\epsilon_i = 0|z_1, z_2, \cdots, z_N) \\
&= \frac{\omega f(z_i)}{\omega f(z_i) + (1-\omega)g(z_i)} = \frac{\omega f(z_i)}{\omega f(z_i) + (1-\omega)f(z_i-\theta)} \\
&= \frac{\omega f(z_i)}{f(z_i) - \theta(1-\omega)\Big(\dfrac{f(z_i) - f(z_i-\theta)}{\theta}\Big)}
\end{aligned}$$

$$\approx \frac{\omega f(z_i)}{f(z_i) - \theta(1-\omega) f'(z_i)} = \frac{\omega}{1 - \theta(1-\omega) f'(z_i)/f(z_i)}$$
$$\approx \omega\Big(1 + \theta(1-\omega)\frac{f'(z_i)}{f(z_i)}\Big). \tag{7.4}$$

Note in the fourth line that the definition of the derivative is given by: $f'(z_i) = \lim_{\theta \to 0} \dfrac{f(z_i) - f(z_i - \theta)}{\theta}$. The last approximation (the fifth line) comes from the fact that $1/(1-r) = 1+r+r^2+\cdots \approx 1+r$ for small r. That is, we consider that $\theta(1-\omega) f'(z_i)/f(z_i)$ is small enough (actually, θ is assumed to be close to zero).

Now, define $Q(z)$ as follows:

$$Q(z) = \omega \int_{-\infty}^{z} f(t)\, \mathrm{d}t + (1-\omega) \int_{-\infty}^{z} g(t)\, \mathrm{d}t,$$

which is interpreted as the probability $P(Z < z)$ under the alternative hypothesis. Note that $Q(z)$ represents the mixture of two distributions $f(\cdot)$ and $g(\cdot)$. Then, we obtain the following relationship:

$$\frac{i}{N} = \frac{1}{N}\{\text{the number of } X_j \text{ and } Y_k,\ j = 1, 2, \cdots, n1 \text{ and } k = 1, 2, \cdots, n2,$$
$$\text{which satisfies } X_j \le z_i \text{ and } Y_k \le z_i\}$$
$$\approx \frac{n1}{N} \int_{-\infty}^{z_i} f(t)\, \mathrm{d}t + \frac{n2}{N} \int_{-\infty}^{z_i} g(t)\, \mathrm{d}t$$
$$\approx \omega \int_{-\infty}^{z_i} f(t)\, \mathrm{d}t + (1-\omega) \int_{-\infty}^{z_i} g(t)\, \mathrm{d}t = Q(z_i). \tag{7.5}$$

Remember that z_i is the ith largest value of $\{z_1, z_2, \cdots, z_N\}$. Noting that $g(z) = f(z-\theta)$, $Q(z)$ is rewritten as:

$$\begin{aligned} Q(z) &= \omega \int_{-\infty}^{z} f(t)\, \mathrm{d}t + (1-\omega) \int_{-\infty}^{z} f(t-\theta)\, \mathrm{d}t \\ &= \int_{-\infty}^{z} f(t)\, \mathrm{d}t - \theta(1-\omega) \int_{-\infty}^{z} \frac{f(t) - f(t-\theta)}{\theta}\, \mathrm{d}t \\ &\approx F(z) - \theta(1-\omega) \int_{-\infty}^{z} f'(t)\, \mathrm{d}t \\ &= F(z) - \theta(1-\omega) f(z), \end{aligned} \tag{7.6}$$

where the third line approximately holds for small θ, which also comes from the definition of the derivative. Thus, the relationship between $Q(z)$ and $F(z)$ is given by: $Q(z) - F(z) = -\theta(1-\omega) f(z)$.

Without loss of generality, we consider $Rz_i = i$, i.e., $p_i = (i - 0.5)/N$. Then, we have the following:

$$a(Rz_i) = a(i) = c\Big(\frac{i - 0.5}{N}\Big) = c(p_i).$$

Consider the conditional expectation of $(1/N)\sum_{i=1}^{N} c(p_i)\epsilon_i$ given $z_1, z_2, \cdots, z_N$ as follows:

$$\begin{aligned}
&\mathrm{E}\Big(\frac{1}{N}\sum_{i=1}^{N} c(p_i)\epsilon_i | z_1, z_2, \cdots, z_N\Big) \\
&= \frac{1}{N}\sum_{i=1}^{N} c(p_i)\,\mathrm{E}(\epsilon_i | z_1, z_2, \cdots, z_N) \\
&= \frac{1}{N}\sum_{i=1}^{N} c(p_i)\,\omega\Big(1+\theta(1-\omega)\frac{f'(z_i)}{f(z_i)}\Big) \\
&= \frac{\omega}{N}\sum_{i=1}^{N} c(p_i) + \theta\omega(1-\omega)\frac{1}{N}\sum_{i=1}^{N} c(Q(z_i))\frac{f'(z_i)}{f(z_i)} \\
&\approx \frac{\omega}{N}\sum_{i=1}^{N} c(p_i) + \theta\omega(1-\omega)\frac{1}{N}\sum_{i=1}^{N} c(F(z_i))\frac{f'(z_i)}{f(z_i)} \\
&\qquad\qquad - \theta^2\omega(1-\omega)^2\frac{1}{N}\sum_{i=1}^{N} c'(F(z_i))f'(z_i) \\
&\approx \omega\frac{1}{N}\sum_{i=1}^{N} c(p_i) + \theta\omega(1-\omega)\frac{1}{N}\sum_{i=1}^{N} c(F(z_i))\frac{f'(z_i)}{f(z_i)} \qquad (7.7)
\end{aligned}$$

In the second equality, the expectation (7.4) is substituted. Since $i/N \approx Q(z_i)$ comes from equation (7.5) and it is approximately equal to $(i-0.5)/N = p_i$, we have $c(p_i) \approx c(Q(z_i))$ in the third equality. In the fifth line, we utilize the following approximation:

$$\begin{aligned}
c(Q(z)) &\approx c(F(z)) + c'(F(z))(Q(z) - F(z)) \\
&= c(F(z)) - \theta(1-\omega)c'(F(z))f(z),
\end{aligned}$$

where $c(Q(z))$ is linearized around $Q(z) = F(z)$ in the first line and equation (7.6) is used in the second line. Moreover, as N goes to infinity, the summations in equation (7.7) are approximated as follows:

$$\frac{1}{N}\sum_{i=1}^{N} c(p_i) \longrightarrow \int_0^1 c(t)\,\mathrm{d}t, \qquad \frac{1}{N}\sum_{i=1}^{N} c(F(z_i))\frac{f'(z_i)}{f(z_i)} \longrightarrow \int c(F(z))f'(z)\,\mathrm{d}z,$$

which do not depend on z_i. This fact implies that the conditional expectation is equal to the unconditional one for sufficiently large N, i.e.,

$$\lim_{N\to\infty}\mathrm{E}\Big(\frac{1}{N}\sum_{i=1}^{N} c(p_i)\epsilon_i \,\Big|\, z_1, z_2, \cdots, z_N\Big) = \lim_{N\to\infty}\mathrm{E}\Big(\frac{1}{N}\sum_{i=1}^{N} c(p_i)\epsilon_i\Big).$$

Therefore, replacing the two summations by the corresponding integrations, the unconditional expectation of $(1/N)\sum_{i=1}^{N} c(p_i)\epsilon_i$ is approximately given by:

$$\mathrm{E}\Big(\frac{1}{N}\sum_{i=1}^{N} c(p_i)\epsilon_i\Big) \approx \omega\int_0^1 c(t)\,\mathrm{d}t + \theta\omega(1-\omega)\int c(F(z))f'(z)\,\mathrm{d}z. \qquad (7.8)$$

Ignoring all the terms which include θ, we can obtain the variance as follows:

$$\mathrm{V}\Big(\frac{1}{N}\sum_{i=1}^{N} c(p_i)\epsilon_i\Big) \approx \frac{\omega(1-\omega)}{N}\Big(\int_0^1 c(t)^2\,\mathrm{d}t - \Big(\int_0^1 c(t)\,\mathrm{d}t\Big)^2\Big). \tag{7.9}$$

When N is large, by the central limit theorem, $(1/N)\sum_{i=1}^{N} c(p_i)\epsilon_i$ approaches the normal distribution with mean (7.8) and variance (7.9).

Now, consider the following statistic:

$$T_1 = \frac{(1/N)\sum_{i=1}^{N} c(p_i)\epsilon_i - \omega\int_0^1 c(t)\,\mathrm{d}t}{\omega(1-\omega)\int c(F(z))f'(z)\,\mathrm{d}z.} \tag{7.10}$$

Then, mean and variance of T_1 are given by:

$$\mathrm{E}(T_1) \approx \theta, \tag{7.11}$$

$$\mathrm{V}(T_1) \approx \frac{1}{\omega(1-\omega)N}\Big(\frac{\int_0^1 c(t)^2\,\mathrm{d}t - \Big(\int_0^1 c(t)\,\mathrm{d}t\Big)^2}{\Big(\int c(F(z))f'(z)\,\mathrm{d}z\Big)^2}\Big). \tag{7.12}$$

Thus, under the assumption of $g(z) = f(z-\theta)$, the test statistic based on the score test, T_1, approximately normally distributed with mean (7.11) and variance (7.12). We want to derive the asymptotic relative efficiency of the score test relative to the *t* test. Therefore, next, we discuss the *t* test.

7.3.2 *t* Test

Let $X_1, X_2, \cdots, X_{n1}$ be mutually independently distributed random variables with mean μ_1 and variance σ^2, and $Y_1, Y_2, \cdots, Y_{n2}$ be mutually independently distributed random variables with mean μ_2 and σ^2. Define $\overline{X} = (1/n1)\sum_{i=1}^{n1} X_i$ and $\overline{Y} = (1/n2)\sum_{j=1}^{n2} Y_j$. Then, we have:

$$\mathrm{E}(\overline{X}) = \mu_1, \quad \mathrm{V}(\overline{X}) = \frac{\sigma^2}{n1}, \qquad \mathrm{E}(\overline{Y}) = \mu_2, \quad \mathrm{V}(\overline{Y}) = \frac{\sigma^2}{n2},$$

where σ^2 indicates the common variance of X and Y, i.e., $\sigma^2 = \mathrm{V}(X) = \mathrm{V}(Y)$. Therefore, mean and variance of $\overline{X} - \overline{Y}$ are given by:

$$\mathrm{E}(\overline{X} - \overline{Y}) = \mu_1 - \mu_2, \qquad \mathrm{V}(\overline{X} - \overline{Y}) = \sigma^2\Big(\frac{1}{n1} + \frac{1}{n2}\Big).$$

By the central limit theorem, we have the following asymptotic distribution:

$$\frac{(\overline{X} - \overline{Y}) - \theta}{\sigma\sqrt{1/n1 + 1/n2}} \longrightarrow N(0,1),$$

where $\mu_1 - \mu_2 \equiv \theta$ is defined. Note that when testing difference between two-sample means we practically use the following test:

$$\frac{(\overline{X} - \overline{Y}) - \theta}{\hat{\sigma}\sqrt{1/n1 + 1/n2}},$$

which approximately has the t distribution with $N - 2$ degrees of freedom, where $\hat{\sigma}^2$ denotes the consistent estimator of σ^2 and it is defined as:

$$\hat{\sigma}^2 = \frac{1}{N-2}\Big(\sum_{i=1}^{n1}(X_i - \overline{X})^2 + \sum_{j=1}^{n2}(Y_j - \overline{Y})^2\Big).$$

As N goes to infinity, the t distribution approaches the standard normal distribution under both null and alternative hypotheses, where the null hypothesis is given by $H_0 : \mu_1 = \mu_2$ while the alternative one is written as $H_1 : \mu_1 \neq \mu_2$, which are rewritten as $H_0 : \theta = 0$ and $H_1 : \theta \neq 0$. Moreover, we have the following approximation:

$$\hat{\sigma}^2\Big(\frac{1}{n1} + \frac{1}{n2}\Big) \approx \sigma^2\Big(\frac{1}{\omega N} + \frac{1}{(1-\omega)N}\Big) = \frac{\sigma^2}{\omega(1-\omega)N}.$$

Thus, the test statistic based on the t test:

$$T_2 = \overline{X} - \overline{Y}. \tag{7.13}$$

Then, mean and variance of T_2 are obtained as follows:

$$\mathrm{E}(T_2) = \theta, \tag{7.14}$$

$$\mathrm{V}(T_2) \approx \frac{1}{\omega(1-\omega)N}\sigma^2. \tag{7.15}$$

Accordingly, T_2 is approximately normally distributed with mean (7.14) and variance (7.15). We can obtain the asymptotic relative efficiency of the score test relative to the t test, comparing (7.10) – (7.12) with (7.13) – (7.15).

7.3.3 Comparison between Two Tests

We have shown two test statistics, i.e., T_1 and T_2, under the alternative hypothesis $H_1 : \theta \neq 0$. T_1 is related to the nonparametric score test, while T_2 is based on the t test. As mentioned above, the asymptotic efficiency of T_1 relative to T_2 is given by:

$$\lim_{N_1 \to \infty} \frac{N_2(N_1)}{N_1},$$

where N_1 denotes the sample size for T_1 and N_2 represents the sample size for T_2, when T_1 has the same power as T_2. That is, when T_1 has the same power as T_2 for sufficiently large N_1 and N_2, the following equation has to hold:

$$\mathrm{V}(T_1) = \mathrm{V}(T_2),$$

which implies that:

$$\frac{1}{\omega(1-\omega)N_1}\left(\frac{\int_0^1 c(t)^2\,\mathrm{d}t - \left(\int_0^1 c(t)\,\mathrm{d}t\right)^2}{\left(\int c(F(z))f'(z)\,\mathrm{d}z\right)^2}\right) = \frac{1}{\omega(1-\omega)N_2}\sigma^2,$$

for large N_1 and N_2. Remember that both T_1 and T_2 are asymptotically normally distributed. Therefore, the asymptotic relative efficiency is given by:

$$\lim_{N_1\to\infty}\frac{N_2(N_1)}{N_1} = \frac{\sigma^2\left(\int c(F(z))f'(z)\,\mathrm{d}z\right)^2}{\int_0^1 c(t)^2\,\mathrm{d}t - \left(\int_0^1 c(t)\,\mathrm{d}t\right)^2}.$$

Moreover, normalizing to be $\int_0^1 c(t)\,\mathrm{d}t = 0$, the asymptotic relative efficiency is rewritten as:

$$\lim_{N_1\to\infty}\frac{N_2(N_1)}{N_1} = \frac{\sigma^2\left(\int c(F(z))f'(z)\,\mathrm{d}z\right)^2}{\int_0^1 c(t)^2\,\mathrm{d}t} = \frac{\sigma^2\left(\int c'(F(z))f(z)^2\,\mathrm{d}z\right)^2}{\int_0^1 c(t)^2\,\mathrm{d}t}. \tag{7.16}$$

In the second equality, note that we have the following equality: $\int c(F(z))f'(z)\,\mathrm{d}z = \Big[c(F(z))f(z)\Big]_{-\infty}^{\infty} - \int c'(F(z))f(z)^2\,\mathrm{d}z = -\int c'(F(z))f(z)^2\,\mathrm{d}z$, under the conditions of $c(F(-\infty))f(-\infty) = c(F(\infty))f(\infty) = 0$. These conditions hold when $f(-\infty) = f(\infty) = 0$, and $c(t)$ is bounded between zero and one.

The asymptotic relative efficiency is numerically computed as follows:

$$\begin{aligned}\lim_{N_1\to\infty}\frac{N_2(N_1)}{N_1} &= \frac{\sigma^2\left(\int c'(F(z))f(z)^2\,\mathrm{d}z\right)^2}{\int_0^1 c(t)^2\,\mathrm{d}t} \\ &\approx \frac{\left(\frac{1}{L}\sum_{i=1}^{L}(z_i^2-\bar{z})^2\right)\left(\frac{1}{L}\sum_{i=1}^{L}c'(p_i)f(z_i)\right)^2}{\frac{1}{L}\sum_{i=1}^{L}c(p_i)^2},\end{aligned} \tag{7.17}$$

where $p_i = (i-0.5)/L$ for $i = 1, 2, \cdots, L$, z_i denotes $100\times p_i$ percent point of the distribution $F(\cdot)$ and $\bar{z}$ represents the arithmetic average of z_1, z_2, $\cdots$, z_L, i.e., $\bar{z} = (1/L)\sum_{i=1}^{L} z_i$. Note that $\sigma^2 = \int(z-\mu)^2 f(z)\,\mathrm{d}z \approx (1/L)\sum_{i=1}^{L}(z_i-\bar{z})^2$ for $\mu = \int z f(z)\,\mathrm{d}z \approx \bar{z}$.

Especially, when $c(t) = t - 0.5$ is taken, the score test reduces to the Wilcoxon rank sum test. Then, the asymptotic relative efficiency of the Wilcoxon test relative to t test is represented by:

$$\lim_{N_1\to\infty}\frac{N_2(N_1)}{N_1} = 12\sigma^2\left(\int f(z)^2\,\mathrm{d}z\right)^2,$$

which is obtained from (7.16). See Hodges and Lehmann (1956) and Lehmann (1983). σ^2 denotes the common variance of X and Y. In order to obtain the minimum value of the asymptotic relative efficiency, consider minimizing:

$$\int f(z)^2 \, dz,$$

subject to the following three conditions:

$$\int f(z) \, dz = 1, \qquad \int z f(z) \, dz = 0, \qquad \int z^2 f(z) \, dz = 1,$$

where the first condition indicates the condition which $f(z)$ is a density function, and the second and third conditions implies that $f(z)$ is normalized to be mean zero and variance one. Then, we have the solution:

$$f(z) = \frac{3\sqrt{5}}{100}(5 - z^2), \qquad \text{for } -\sqrt{5} \leq z \leq \sqrt{5}. \tag{7.18}$$

Under the above density function, therefore, the asymptotic relative efficiency of the Wilcoxon rank sum test relative to the t test is minimized, which value is given by $108/125 = 0.864$. That is, we have the following inequality:

$$\lim_{N_1 \to \infty} \frac{N_2(N_1)}{N_1} \geq \frac{108}{125} = 0.864,$$

which implies that the asymptotic relative efficiency of the Wilcoxon test relative to the t test is always greater than or equal to 0.864. This result indicates that the Wilcoxon test is not too bad, compared with the t test.

When $c(t) = \Phi^{-1}(t)$ is taken, where $\Phi(z)$ denotes the standard normal cumulative distribution function, the score test reduces to the normal score test. When $c(t) = \log(t/(1-t))$ is taken, the score test indicates the logistic score test. When $c(t) = \tan(\pi(t - 0.5))$ is taken, the score test is the Cauchy score test.

In Table 7.1, the asymptotic relative efficiency (ARE) of the score test relative to the t test, shown in equation (7.17), is computed, where *w*, *ns*, *ls* and *cs* denote the Wilcoxon test, the normal score test, the logistic score test and the Cauchy score test. (B), (C), (X), (G), (D), (L), (N), (T), (U) and (Q) indicate that the underlying population distribution is given by the bimodal distribution which consists of two normal distributions, the Cauchy distribution, the chi-square distribution with one degree of freedom, the Gumbel distribution (extreme-value distribution), the double exponential distribution (LaPlace distribution), the logistic distribution, the standard normal distribution, the t distribution with three degrees of freedom, the uniform distribution on the interval between -0.5 and 0.5, and the quadratic distribution shown in equation (7.18), respectively. These distributions are shown in the next section. That is, *w* and (B) implies the asymptotic relative efficiency of the Wilcoxon test relative to the t test when a population distribution is bimodal. It is not easy to obtain all the

Table 7.1: ARE of Score Tests Relative to t Test

	L	(B)	(C)	(X)	(G)	(D)	(L)	(N)	(T)	(U)	(Q)
w	10^2	1.188	—	—	1.200	1.437	1.065	0.943	1.481	1.000	0.863
	10^3	1.195	—	—	1.229	1.491	1.092	0.954	1.705	1.000	0.864
	10^4	1.195	—	—	1.233	1.499	1.096	0.955	1.810	1.000	0.864
	10^5	1.195	—	—	1.234	1.500	1.097	0.955	1.858	1.000	0.864
	∞	1.195	∞	∞	1.234	1.500	1.097	0.955	1.900	1.000	0.864
ns	10^2	1.524	—	—	1.320	1.236	1.031	1.000	1.295	3.050	1.232
	10^3	1.516	—	—	1.339	1.268	1.045	1.000	1.474	4.429	1.256
	10^4	1.515	—	—	1.341	1.272	1.047	1.000	1.562	5.868	1.265
	10^5	1.515	—	—	1.342	1.273	1.047	1.000	1.603	7.331	1.268
ls	10^2	1.627	—	—	1.348	1.152	1.000	1.006	1.205	4.500	1.401
	10^3	1.601	—	—	1.352	1.166	1.000	0.994	1.352	8.005	1.452
	10^4	1.597	—	—	1.353	1.168	1.000	0.992	1.429	—	1.474
	10^5	1.596	—	—	1.353	1.168	1.000	0.992	1.467	—	1.482
cs	10^2	1.936	1.000	—	1.230	0.324	0.430	0.741	0.231	—	4.570
	10^3	0.531	1.000	—	0.375	0.061	0.086	0.193	0.032	—	5.157
	10^4	0.108	1.000	—	0.088	0.010	0.014	0.040	0.004	—	5.403
	10^5	0.018	1.000	—	0.018	0.001	0.002	0.007	0.000	—	5.511

cases analytically. Therefore, in Table 7.1 we numerically compute the asymptotic relative efficiencies, using equation (7.17), where $L = 10^2, 10^3, 10^4, 10^5$ is taken. — indicates that the numerically computed value is greater than 10.0. As for ARE of the Wilcoxon test (i.e., w) relative to the t test, (C), (X), (D) and (T) are much greater than one, and therefore the Wilcoxon test is very powerful when (C), (X), (D) or (T) is a population distribution. The AREs of the normal score test (i.e., ns) relative to the t test are greater than one for all the cases of the underlying population. For cs, the AREs of (B), (G), (D), (L), (N) and (T) are close to zero when $L = 10^5$. Thus, the Cauchy score test shows a poor performance in large sample.

7.4 Power Comparison (Small Sample Properties)

In Section 7.2, we have introduced the score tests (i.e., Wilcoxon rank sum test, normal score test, logistic score test and Cauchy score test) and the Fisher test. In Section 7.3, the asymptotic relative efficiencies have been discussed. In this section, we examine small sample properties of the score tests and the Fisher test through Monte-Carlo experiments. Assuming a specific distribution for Group 1 sample $\{x_i\}_{i=1}^{n1}$ and Group 2 sample $\{y_j\}_{j=1}^{n2}$ and generating random draws, we compare the nonparametric tests and the t test with respect to the empirical size and the sample power. The underlying distributions examined in this section are as follows.

Bimodal Distribution (B): $f(x) = \frac{1}{2}N(\mu_1, \sigma_1^2) + \frac{1}{2}N(\mu_2, \sigma_2^2)$,

for $(\mu_1, \sigma_1) = (1, 1)$ and $(\mu_2, \sigma_2) = (-1, 0.5)$, where $N(\mu, \sigma^2)$ denotes the following function:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{1}{2\sigma^2}(x-\mu)^2\Big).$$

Note that we have $\mathrm{E}(X) = 0$ and $\mathrm{V}(X) = 1.625$.

Cauchy Distribution (C): $f(x) = \dfrac{1}{\pi(1+x^2)}$,
where $\mathrm{E}(X)$ and $\mathrm{V}(X)$ do not exist, i.e., both are infinity.

Chi-Square Distribution (X): $f(x) = \dfrac{1}{\Gamma(k/2)} 2^{-k/2} x^{k/2-1} e^{-x/2}$,
for $0 < x$, where $\mathrm{E}(X) = k$ and $\mathrm{V}(X) = 2k$. In this section, $k = 1$ is taken.

Gumbel (Extreme-Value) Distribution (G): $F(x) = \exp(-e^{-x})$,
where $\mathrm{E}(X) = 0.5772156599$ and $\mathrm{V}(X) = \pi^2/6$.

LaPlace (Double Exponential) Distribution (D): $f(x) = \dfrac{1}{2}\exp(-|x|)$,
where $\mathrm{E}(X) = 0$ and $\mathrm{V}(X) = 2$.

Logistic Distribution (L): $F(x) = \dfrac{1}{1+\exp(-x)}$,
where $\mathrm{E}(X) = 0$ and $\mathrm{V}(X) = \pi^2/3$.

Normal Distribution (N): $f(x) = \dfrac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$,
where $\mathrm{E}(X) = 0$ and $\mathrm{V}(X) = 1$.

***t*(3) Distribution (T):** $f(x) = \dfrac{\Gamma(\frac{k+1}{2})}{\Gamma(\frac{k}{2})} \dfrac{1}{\sqrt{k\pi}} \dfrac{1}{(1+x^2/k)^2}$,
where $\mathrm{E}(X) = 0$ and $\mathrm{V}(X) = k/(k-2)$. In this section, $k = 3$ is taken.

Uniform Distribution (U): $f(x) = \dfrac{1}{4}$,
for $-2 < x < 2$, where $\mathrm{E}(X) = 0$ and $\mathrm{V}(X) = 4/3$.

Quadratic Distribution (Q): $f(x) = \dfrac{3(5-x^2)}{20\sqrt{5}}$,
for $-\sqrt{5} < x < \sqrt{5}$, where $\mathrm{E}(X) = 0$ and $\mathrm{V}(X) = 1$. As discussed on p.408, the asymptotic efficiency of the Wilcoxon test relative to the t test shows the smallest value when population follows this distribution.

7.4.1 Setup of the Monte Carlo Studies

The simulation procedure is as follows. We generate random draws for x_i and y_j, where $\mathrm{E}(x_i) = \mu_1$ for $i = 1, 2, \cdots, n1$ and $\mathrm{E}(y_j) = \mu_2$ for $j = 1, 2, \cdots, n2$. (B), (C), (X), (G), (D), (L), (N), (T), (U) and (Q) are examined for population distribution. For each underlying distribution, variances are assumed to equal between two samples. Then, the t test discussed in Section 7.3.2 is compared with the nonparametric tests introduced in Section 7.2.

The null hypothesis is $H_0 : F(x) = G(x)$. We compare the sample powers for a shift in location parameter $\mu_1 - \mu_2 = \theta$. Therefore, the alternative hypothesis is given by $H_1 : F(x) = G(x - \theta)$. We perform Monte-Carlo experiments in the following cases: $n1 = n2 = 9$, 12, 15, $\mu_1 - \mu_2 = \theta = 0.0$, 0.5, 1.0 and $\alpha = 0.10$, 0.05, 0.01, α denotes the significance level.

The results are in Table 7.2, where t, w, aw, ns, ls, cs and f represent the t test, the Wilcoxon test, the asymptotic Wilcoxon test, the normal score test, the logistic score test, the Cauchy score test and the Fisher randomization test, respectively. In Tables 7.2, note as follows.

(i) Perform G simulation runs, where $G = 10^4$ is taken. Given the significance level $\alpha = 0.10,\ 0.05,\ 0.01$, each value in Table 7.2 is the number of rejections divided by the number of simulation runs (i.e., G), which represents the following probabilities: $P(t < -t_0) < \alpha$, $P(w \geq w_0) < \alpha$, $P(aw < -aw_0) < \alpha$, $P(ns \geq ns_0) < \alpha$, $P(ls \geq ls_0) < \alpha$, $P(cs \geq cs_0) < \alpha$ and $P(f \geq f_0) < \alpha$, where t_0, w_0, aw_0, ns_0, ls_0, cs_0 and f_0 are the corresponding test statistics obtained from the original data for each simulation run. For the t distribution and the normal distribution, given the significance level, we have the following critical values:

α		0.10	0.05	0.01
t Dist.	$n1 = n2 = 9$	1.3368	1.7459	2.5835
	$n1 = n2 = 12$	1.3212	1.7171	2.5083
	$n1 = n2 = 15$	1.3125	1.7011	2.4671
$N(0, 1)$ Dist.		1.2816	1.6449	2.3263

Let $\hat{p}_t$, $\hat{p}_w$, $\hat{p}_{aw}$, $\hat{p}_{ns}$, $\hat{p}_{ls}$, $\hat{p}_{cs}$ and $\hat{p}_f$ be the number of rejections divided by the number of simulation runs (i.e., G) for each test. They indicate the probabilities which reject the null hypothesis under the alternative one, or equivalently, the sample powers.

(ii) The estimated variance of each value in Table 7.2 is estimated by: $\mathrm{V}(\hat{p}_k) \approx \hat{p}_k(1 - \hat{p}_k)/G$ for $k = t,\ w,\ aw,\ ns,\ ls,\ cs,\ f$ and $G = 10^4$.

(iii) In the case of $\theta = 0$ of Table 7.2, $^{\bullet}$, $^{\bullet\bullet}$, $^{\bullet\bullet\bullet}$, $^{\circ}$, $^{\circ\circ}$ and $^{\circ\circ\circ}$ represent comparison with the significance level α. We put the superscript $^{\bullet}$ when $(\hat{p}_k - \alpha)/\sqrt{\mathrm{V}(\hat{p}_k)}$, $k = w,\ aw,\ ns,\ ls,\ cs,\ f$, is greater than 1.6449, $^{\bullet\bullet}$ when it is greater than 1.9600, and $^{\bullet\bullet\bullet}$ when it is greater than 2.5758, where $\mathrm{V}(\hat{p}_k)$ is given by $\mathrm{V}(\hat{p}_k) =$

$\alpha(1-\alpha)/G$ under the null hypothesis $H_0 : p_k = \alpha$, where p_k denotes the true size or power, and k takes w, aw, ns, ls, cs or f. 1.6449, 1.9600 and 2.5758 correspond to 95%, 97.5% and 99.5% points of the standard normal distribution, respectively. We put the superscript $^\circ$ if $(\hat{p}_k - \alpha)/\sqrt{\mathrm{V}(\hat{p}_k)}$, $k = w$, aw, ns, ls, cs, f, is less than -1.6449, the superscript $^{\circ\circ}$ if it is less than -1.9600, and the superscript $^{\circ\circ\circ}$ if it is less than -2.5758. Remember that $(\hat{p}_k - \alpha)/\sqrt{\mathrm{V}(\hat{p}_k)} \sim N(0,1)$ asymptotically holds under the null hypothesis $H_0 : p_k = \alpha$ and the alternative one $H_1 : p_k \neq \alpha$. Therefore, in the case of $\theta = 0$, the values with $^\bullet$ or $^\circ$ indicate that the empirical size is statistically different from the true size. Thus, the number of $^\bullet$ or $^\circ$ increases as the empirical size is far from the true size.

(iv) However, in the case of $\theta = 0.5, 1.0$ (i.e., $\theta \neq 0$) of Table 7.2, $^\bullet$, $^{\bullet\bullet}$, $^{\bullet\bullet\bullet}$, $^\circ$, $^{\circ\circ}$ and $^{\circ\circ\circ}$ indicate comparison with the t test. We put the superscript $^\bullet$ when $(\hat{p}_k - \hat{p}_t)/\sqrt{\mathrm{V}(\hat{p}_k) + \mathrm{V}(\hat{p}_t)}$, $k = w$, aw, ns, ls, cs, f, is greater than 1.6449, the superscript $^{\bullet\bullet}$ when it is greater than 1.9600, and the superscript $^{\bullet\bullet\bullet}$ when it is greater than 2.5758. The two variances are approximated as: $\mathrm{V}(\hat{p}_k) \approx \hat{p}_k(1-\hat{p}_k)/G$ and $\mathrm{V}(\hat{p}_t) \approx \hat{p}_t(1-\hat{p}_t)/G$. We put the superscript $^\circ$ if $(\hat{p}_k - \hat{p}_t)/\sqrt{\mathrm{V}(\hat{p}_k) + \mathrm{V}(\hat{p}_t)}$, $k = w$, aw, ns, ls, cs, f, is less than -1.6449, the superscript $^{\circ\circ}$ if it is less than -1.9600, and the superscript $^{\circ\circ\circ}$ if it is less than -2.5758. Note that in large sample we have the following result: $(\hat{p}_k - \hat{p}_t)/\sqrt{\mathrm{V}(\hat{p}_k) + \mathrm{V}(\hat{p}_t)} \sim N(0,1)$ under the null hypothesis $H_0 : \theta = 0$ and the alternative one $H_1 : \theta \neq 0$. Therefore, in the case of $\theta \neq 0$, the values with $^\bullet$ indicate more powerful test than the t test. In addition, the number of $^\bullet$ shows degree of the sample power. Contrarily, the values with $^\circ$ represent less powerful test than the t test.

7.4.2 Results and Discussion

In Table 7.2, each value represents an empirical size when $\theta = 0$ and a sample power when $\theta \neq 0$. We summarize the two cases separately as follows.

Empirical Size ($\theta = 0$): First, we compare the nonparametric tests and the t test with respect to the empirical size, which corresponds to the case of $\theta = 0$. The empirical size is compared with the significance level α.

The Cauchy score test, cs, shows the correct sizes for most of the cases. For only two cases, i.e., $.0079^{\circ\circ}$ in $n = 12$ and $\alpha = 0.01$ of (B) and $.1077^{\bullet\bullet}$ in $n = 12$ and $\alpha = 0.10$ of (T), out of $10 \times 3 \times 3 = 90$ (i.e., 10 population distributions, $n = 9, 12, 15$, and $\alpha = .10, .05, .01$), we obtain the result that the true size is significantly different from the significance level.

In the size criterion, the second best test is given by the Fisher test, f. About 80% out of 90 cases represents that the size is not distorted. As for the normal score test, ns, and the logistic score test, ls, about 60% out of 90 performs well.

Table 7.2: Empirical Sizes and Sample Powers

	n	θ	α	f	w	ns	ls	cs	aw	t
(B)	9	0.0	.10	.1001	.0947°	.0976	.0981	.0996	.0947°	.0986
			.05	.0481	.0463°	.0494	.0489	.0493	.0463°	.0483
			.01	.0094	.0086	.0094	.0091	.0092	.0086	.0102
		0.5	.10	.3232	.3406•••	.3671•••	.3708•••	.3790•••	.3406•••	.3209
			.05	.1952	.2103•••	.2325•••	.2354•••	.2385•••	.2103•••	.1954
			.01	.0549	.0603	.0661•••	.0673•••	.0708•••	.0603	.0573
		1.0	.10	.6392	.6303	.6827•••	.6872•••	.6968•••	.6303	.6380
			.05	.4780	.4781	.5212•••	.5277•••	.5503•••	.4781	.4778
			.01	.2004	.2102	.2322•••	.2335•••	.2427•••	.2102	.2058
	12	0.0	.10	.0936°°	.0936°°	.0953	.0952	.0975	.0936°°	.0933°°
			.05	.0459°	.0406°°°	.0464°	.0459°	.0474	.0473	.0458°
			.01	.0091	.0076°°	.0092	.0089	.0079°°	.0076°°	.0093
		0.5	.10	.3582	.3848•••	.4117•••	.4195•••	.4259•••	.3848•••	.3571
			.05	.2276	.2322	.2754•••	.2805•••	.2930•••	.2522•••	.2275
			.01	.0691	.0739	.0896•••	.0922•••	.0960•••	.0739	.0704
		1.0	.10	.7200	.7221	.7710•••	.7809•••	.7706•••	.7221	.7194
			.05	.5732	.5591°°	.6304•••	.6405•••	.6492•••	.5841	.5734
			.01	.2814	.2779	.3294•••	.3370•••	.3602•••	.2779	.2864
	15	0.0	.10	.1018	.0951	.1014	.1009	.1027	.1025	.1012
			.05	.0469	.0444°°	.0497	.0498	.0501	.0444°°	.0469
			.01	.0095	.0087	.0091	.0095	.0097	.0087	.0103
		0.5	.10	.4133	.4223	.4737•••	.4829•••	.4821•••	.4385•••	.4127
			.05	.2691	.2938•••	.3318•••	.3375•••	.3461•••	.2938•••	.2686
			.01	.0931	.1034•	.1223•••	.1264•••	.1342•••	.1034•	.0953
		1.0	.10	.8042	.7967	.8513•••	.8595•••	.8228•••	.8092	.8046
			.05	.6780	.6782	.7367•••	.7462•••	.7425•••	.6782	.6782
			.01	.3808	.3838	.4431•••	.4521•••	.4778•••	.3838	.3855
(C)	9	0.0	.10	.0951	.0877°°°	.0928°°	.0936°°	.0977	.0877°°°	.0850°°°
			.05	.0481	.0420°°°	.0438°°°	.0438°°°	.0469	.0420°°°	.0299°°°
			.01	.0103	.0084	.0092	.0094	.0091	.0084	.0025°°°
		0.5	.10	.1681•••	.2200•••	.2104•••	.2053•••	.1709•••	.2200•••	.1483
			.05	.1008•••	.1255•••	.1207•••	.1179•••	.1026•••	.1255•••	.0655
			.01	.0306•••	.0321•••	.0329•••	.0329•••	.0313•••	.0321•••	.0094
		1.0	.10	.2556•••	.3967•••	.3632•••	.3478•••	.2541•••	.3967•••	.2305
			.05	.1735•••	.2602•••	.2405•••	.2293•••	.1785•••	.2602•••	.1217
			.01	.0730•••	.0916•••	.0857•••	.0824•••	.0695•••	.0916•••	.0283
	12	0.0	.10	.0962	.0927°°	.0940°°	.0942°	.0970	.0927°°	.0867°°°
			.05	.0484	.0411°°°	.0454°°	.0453°°	.0477	.0474	.0285°°°
			.01	.0102	.0091	.0098	.0095	.0102	.0091	.0024°°°
		0.5	.10	.1687•••	.2498•••	.2277•••	.2202•••	.1697•••	.2498•••	.1487
			.05	.1024•••	.1392•••	.1359•••	.1302•••	.1014•••	.1530•••	.0665
			.01	.0325•••	.0393•••	.0408•••	.0390•••	.0322•••	.0393•••	.0099
		1.0	.10	.2628•••	.4684•••	.4158•••	.3945•••	.2423	.4684•••	.2367
			.05	.1798•••	.3095•••	.2815•••	.2656•••	.1738•••	.3314•••	.1239
			.01	.0787•••	.1172•••	.1090•••	.1021•••	.0743•••	.1172•••	.0312
	15	0.0	.10	.0988	.0860°°°	.0947°	.0953	.0973	.0936°°	.0893°°°
			.05	.0458°	.0423°°°	.0447°°	.0443°°°	.0465	.0423°°°	.0271°°°
			.01	.0098	.0092	.0094	.0090	.0092	.0092	.0013°°°
		0.5	.10	.1772•••	.2775•••	.2576•••	.2470•••	.1715••	.2925•••	.1600
			.05	.1045•••	.1693•••	.1538•••	.1463•••	.1018•••	.1693•••	.0677
			.01	.0313•••	.0460•••	.0426•••	.0407•••	.0297•••	.0460•••	.0076
		1.0	.10	.2664•••	.5319•••	.4788•••	.4533•••	.2388	.5495•••	.2412
			.05	.1882•••	.4018•••	.3396•••	.3164•••	.1759•••	.4018•••	.1325
			.01	.0776•••	.1596•••	.1342•••	.1232•••	.0718•••	.1596•••	.0314

Table 7.2: Empirical Sizes and Sample Powers —< Continued >—

	n	θ	α	f	w	ns	ls	cs	aw	t
(X)	9	0.0	.10	.0981	.0940°°	.0993	.0997	.0995	.0940°°	.1017
			.05	.0522	.0466	.0495	.0502	.0510	.0466	.0463°
			.01	.0100	.0102	.0110	.0109	.0114	.0102	.0047°°°
		0.5	.10	.3658	.5711•••	.5895•••	.5839•••	.4983•••	.5711•••	.3692
			.05	.2588•••	.4255•••	.4336•••	.4287•••	.3815•••	.4255•••	.2417
			.01	.1143•••	.1841•••	.1833•••	.1805•••	.1604•••	.1841•••	.0741
		1.0	.10	.6470	.8324•••	.8280•••	.8175•••	.6375°°	.8324•••	.6525
			.05	.5371	.7188•••	.7103•••	.6883•••	.5636•••	.7188•••	.5257
			.01	.3390•••	.4515•••	.4297•••	.4216•••	.3350•••	.4515•••	.2824
	12	0.0	.10	.1029	.1007	.1021	.1020	.1042	.1007	.1073••
			.05	.0509	.0420°°°	.0507	.0514	.0535	.0482	.0469
			.01	.0095	.0092	.0112	.0111	.0106	.0092	.0062°°°
		0.5	.10	.4033	.6728•••	.6879•••	.6829•••	.5379•••	.6728•••	.4072
			.05	.2884	.5091•••	.5313•••	.5272•••	.4395•••	.5310•••	.2781
			.01	.1330•••	.2542•••	.2670•••	.2626•••	.2249•••	.2542•••	.0963
		1.0	.10	.7055	.9100•••	.9067•••	.9003•••	.6522°°°	.9100•••	.7112
			.05	.6054	.8257•••	.8266•••	.8125•••	.6112	.8421•••	.6005
			.01	.3998•••	.5864•••	.5686•••	.5476•••	.4251•••	.5864•••	.3582
	15	0.0	.10	.0968	.0916°°°	.0969	.0984	.0986	.0997	.1010
			.05	.0472	.0444°°	.0461°	.0470	.0481	.0444°°	.0447°°
			.01	.0096	.0092	.0095	.0097	.0095	.0092	.0050°°°
		0.5	.10	.4182	.7319•••	.7547•••	.7525•••	.5372•••	.7445•••	.4252
			.05	.3014	.6071•••	.6178•••	.6129•••	.4615•••	.6071•••	.2946
			.01	.1381•••	.3306•••	.3325•••	.3251•••	.2536•••	.3306•••	.1088
		1.0	.10	.7537	.9462•••	.9479•••	.9420•••	.6385°°°	.9506•••	.7582
			.05	.6554	.8972•••	.8879•••	.8785•••	.6100°°°	.8972•••	.6523
			.01	.4464•••	.7137•••	.6864•••	.6640•••	.4630•••	.7137•••	.4085
(G)	9	0.0	.10	.0951	.0877°°°	.0928°°	.0936°°	.0977	.0877°°°	.0963
			.05	.0457°°	.0420°°°	.0438°°°	.0438°°°	.0469	.0420°°°	.0450°°
			.01	.0101	.0084	.0092	.0094	.0091	.0084	.0087
		0.5	.10	.3222	.3232	.3408•••	.3402••	.3248	.3232	.3234
			.05	.2033	.2045	.2118•	.2112•	.2044	.2045	.2016
			.01	.0624•	.0578	.0633•	.0639••	.0642••	.0578	.0568
		1.0	.10	.6420	.6570••	.6703•••	.6674•••	.6021°°°	.6570••	.6427
			.05	.4978	.5100••	.5215•••	.5160•••	.4811°°	.5100••	.4955
			.01	.2311	.2300	.2374••	.2377••	.2254	.2300	.2237
	12	0.0	.10	.0967	.0927°°	.0940°°	.0942°	.0970	.0927°°	.0975
			.05	.0475	.0411°°°	.0454°°	.0453°°	.0477	.0474	.0471
			.01	.0100	.0091	.0098	.0095	.0102	.0091	.0090
		0.5	.10	.3660	.3833••	.3932•••	.3933•••	.3559°	.3833••	.3684
			.05	.2424	.2370	.2610•••	.2606•••	.2391	.2568•••	.2408
			.01	.0848	.0775	.0893••	.0891•	.0848	.0775	.0815
		1.0	.10	.7244	.7558•••	.7662•••	.7623•••	.6458°°°	.7558•••	.7252
			.05	.5858	.6051•••	.6294•••	.6243•••	.5462°°°	.6265•••	.5845
			.01	.3225•	.3249••	.3456•••	.3418•••	.3098	.3249••	.3114
	15	0.0	.10	.0950°	.0860°°°	.0947°	.0953	.0973	.0936°°	.0961
			.05	.0451°°	.0423°°°	.0447°°	.0443°°°	.0465	.0423°°°	.0449°°
			.01	.0097	.0092	.0094	.0090	.0092	.0092	.0090
		0.5	.10	.4215	.4409••	.4628•••	.4587•••	.3859°°°	.4586•••	.4243
			.05	.2862	.3017•••	.3145•••	.3147•••	.2789	.3017•••	.2851
			.01	.0991	.1010	.1074•••	.1084•••	.1025•	.1010	.0950
		1.0	.10	.7981	.8283•••	.8470•••	.8449•••	.6787°°°	.8399•••	.7994
			.05	.6868	.7296•••	.7405•••	.7344•••	.6108°°°	.7296•••	.6859
			.01	.4187	.4489•••	.4631•••	.4602•••	.3868°°°	.4489•••	.4085

Table 7.2: Empirical Sizes and Sample Powers —< Continued >—

	n	θ	α	f	w	ns	ls	cs	aw	t
(D)	9	0.0	.10	.0925°°	.0877°°°	.0928°°	.0936°°	.0977	.0877°°°	.0938°°
			.05	.0450°°	.0420°°°	.0438°°°	.0438°°°	.0469	.0420°°°	.0448°°
			.01	.0103	.0084	.0092	.0094	.0091	.0084	.0082°
		0.5	.10	.3003	.3180••	.3164•	.3093	.2659°°°	.3180••	.3038
			.05	.1879	.2021•••	.1978••	.1915	.1707°°°	.2021•••	.1856
			.01	.0591••	.0592••	.0590••	.0592••	.0562	.0592••	.0510
		1.0	.10	.5840	.6330•••	.6103•••	.5940	.4677°°°	.6330•••	.5861
			.05	.4484	.4780•••	.4625••	.4484	.3714°°°	.4780•••	.4445
			.01	.2090•••	.2196•••	.2120•••	.2046••	.1788°°	.2196•••	.1926
	12	0.0	.10	.0941°°	.0927°°	.0940°°	.0942°	.0970	.0927°°	.0965
			.05	.0484	.0411°°°	.0454°°	.0453°°	.0477	.0474	.0482
			.01	.0106	.0091	.0098	.0095	.0102	.0091	.0086
		0.5	.10	.3410	.3807•••	.3588••	.3528	.2749°°°	.3807•••	.3449
			.05	.2205	.2342••	.2336••	.2277	.1821°°°	.2509•••	.2195
			.01	.0757•	.0768•	.0807•••	.0779••	.0672	.0768•	.0696
		1.0	.10	.6686	.7312•••	.6965•••	.6813	.4842°°°	.7312•••	.6714
			.05	.5293	.5754•••	.5605•••	.5433••	.4060°°°	.5985•••	.5282
			.01	.2805••	.3028•••	.2914•••	.2795••	.2189°°°	.3028•••	.2658
	15	0.0	.10	.0956	.0860°°°	.0947°	.0953	.0973	.0936°°	.0970
			.05	.0471	.0423°°°	.0447°°	.0443°°°	.0465	.0423°°°	.0463°
			.01	.0098	.0092	.0094	.0090	.0092	.0092	.0087
		0.5	.10	.3838	.4327•••	.4179•••	.4049•••	.2882°°°	.4500•••	.3864
			.05	.2615	.2981•••	.2836•••	.2750••	.2009°°°	.2981•••	.2606
			.01	.0892	.1012•••	.0974•••	.0928••	.0723°°°	.1012•••	.0845
		1.0	.10	.7413	.8056•••	.7836•••	.7642•••	.5004°°°	.8173•••	.7428
			.05	.6181	.7009•••	.6577•••	.6344••	.4308°°°	.7009•••	.6176
			.01	.3540•	.4220•••	.3853•••	.3667•••	.2570°°°	.4220•••	.3419
(L)	9	0.0	.10	.0944°	.0877°°°	.0928°°	.0936°°	.0977	.0877°°°	.0948°
			.05	.0458°	.0420°°°	.0438°°°	.0438°°°	.0469	.0420°°°	.0456°°
			.01	.0098	.0084	.0092	.0094	.0091	.0084	.0092
		0.5	.10	.2359	.2279	.2344	.2330	.2216°°	.2279	.2367
			.05	.1370	.1307	.1363	.1353	.1297	.1307	.1365
			.01	.0366	.0338	.0363	.0359	.0356	.0338	.0345
		1.0	.10	.4445	.4384	.4423	.4384	.3864°°°	.4384	.4456
			.05	.3019	.2923	.3001	.2953	.2736°°°	.2923	.3015
			.01	.1088	.1025	.1068	.1055	.0995	.1025	.1052
	12	0.0	.10	.0933°°	.0927°°	.0940°°	.0942°	.0970	.0927°°	.0938°°
			.05	.0473	.0411°°°	.0454°°	.0453°°	.0477	.0474	.0472
			.01	.0106	.0091	.0098	.0095	.0102	.0091	.0095
		0.5	.10	.2626	.2612	.2597	.2589	.2326°°°	.2612	.2634
			.05	.1597	.1484°°	.1586	.1550	.1400°°°	.1629	.1589
			.01	.0474	.0418	.0453	.0441	.0428	.0418	.0452
		1.0	.10	.5156	.5203	.5141	.5073	.4141°°°	.5203	.5170
			.05	.3707	.3537°°	.3694	.3644	.3076°°°	.3767	.3705
			.01	.1531	.1382°	.1464	.1459	.1312°°°	.1382°	.1472
	15	0.0	.10	.0932°°	.0860°°°	.0947°	.0953	.0973	.0936°°	.0939°°
			.05	.0459°	.0423°°°	.0447°°	.0443°°°	.0465	.0423°°°	.0459°
			.01	.0095	.0092	.0094	.0090	.0092	.0092	.0092
		0.5	.10	.3012	.2906°°	.3024	.2988	.2495°°°	.3051	.3034
			.05	.1844	.1807	.1817	.1797	.1572°°°	.1807	.1841
			.01	.0526	.0472	.0507	.0508	.0472	.0472	.0506
		1.0	.10	.5895	.5924	.5931	.5852	.4393°°°	.6080••	.5911
			.05	.4381	.4510•	.4420	.4358	.3408°°°	.4510•	.4373
			.01	.1973	.1888	.1934	.1889	.1598°°°	.1888	.1944

Table 7.2: Empirical Sizes and Sample Powers —< Continued >—

	n	θ	α	f	w	ns	ls	cs	aw	t
(N)	9	0.0	.10	.0973	.0914°°°	.0966	.0963	.0988	.0914°°°	.0973
			.05	.0496	.0462°	.0482	.0484	.0487	.0462°	.0494
			.01	.0101	.0097	.0103	.0102	.0093	.0097	.0104
		0.5	.10	.4094	.3829°°°	.3972°	.3959°°	.3693°°°	.3829°°°	.4096
			.05	.2699	.2497°°°	.2623	.2619	.2509°°°	.2497°°°	.2700
			.01	.0843	.0790	.0845	.0842	.0817	.0790	.0839
		1.0	.10	.7922	.7592°°°	.7746°°°	.7721°°°	.7070°°°	.7592°°°	.7921
			.05	.6594	.6215°°°	.6360°°°	.6324°°°	.5964°°°	.6215°°°	.6585
			.01	.3579	.3254°°°	.3388°°°	.3368°°°	.3268°°°	.3254°°°	.3587
	12	0.0	.10	.1014	.0982	.1016	.1022	.0998	.0982	.1013
			.05	.0494	.0442°°°	.0492	.0498	.0500	.0492	.0492
			.01	.0102	.0084	.0095	.0097	.0095	.0084	.0098
		0.5	.10	.4734	.4549°°°	.4622	.4594°°	.4080°°°	.4549°°°	.4735
			.05	.3261	.2892°°°	.3165	.3165	.2926°°°	.3121°°	.3261
			.01	.1227	.1056°°°	.1209	.1192	.1128°°	.1056°°°	.1230
		1.0	.10	.8727	.8563°°°	.8625°°	.8602°°	.7612°°°	.8563°°°	.8725
			.05	.7726	.7356°°°	.7598°°	.7564°°°	.6851°°°	.7546°°°	.7728
			.01	.4925	.4476°°°	.4742°°°	.4732°°°	.4431°°°	.4476°°°	.4933
	15	0.0	.10	.0968	.0941°°	.0960	.0984	.0966	.1007	.0965
			.05	.0514	.0493	.0505	.0505	.0487	.0493	.0514
			.01	.0096	.0090	.0105	.0106	.0105	.0090	.0099
		0.5	.10	.5313	.5001°°°	.5222	.5215	.4443°°°	.5164°°	.5317
			.05	.3820	.3635°°°	.3744	.3754	.3283°°°	.3635°°°	.3819
			.01	.1459	.1346°°	.1432	.1438	.1284°°°	.1346°°	.1471
		1.0	.10	.9261	.9052°°°	.9165°°	.9146°°°	.7956°°°	.9129°°°	.9261
			.05	.8540	.8324°°°	.8441°°	.8396°°°	.7400°°°	.8324°°°	.8541
			.01	.6141	.5821°°°	.5987°°	.5963°°°	.5314°°°	.5821°°°	.6157
(T)	9	0.0	.10	.0992	.0933°°	.1009	.1001	.0984	.0933°°	.1000
			.05	.0499	.0454°°	.0478	.0473	.0487	.0454°°	.0468
			.01	.0084	.0085	.0092	.0093	.0090	.0085	.0054°°°
		0.5	.10	.2857	.3033••	.2996•	.2963	.2516°°°	.3033••	.2881
			.05	.1822	.1922•••	.1906•••	.1848••	.1650°	.1922•••	.1738
			.01	.0605•••	.0581•••	.0597•••	.0587•••	.0532	.0581•••	.0489
		1.0	.10	.5482	.6013•••	.5836•••	.5703•••	.4505°°°	.6013•••	.5518
			.05	.4171•	.4507•••	.4356•••	.4254•••	.3584°°°	.4507•••	.4055
			.01	.1976•••	.2024•••	.1968•••	.1921•••	.1735	.2024•••	.1723
	12	0.0	.10	.1053•	.1005	.1036	.1044	.1077••	.1005	.1072••
			.05	.0540•	.0461°	.0521	.0527	.0533	.0524	.0511
			.01	.0114	.0097	.0115	.0115	.0116	.0097	.0085
		0.5	.10	.3371	.3739•••	.3618•••	.3545••	.2772°°°	.3739•••	.3386
			.05	.2218	.2286••	.2312•••	.2243	.1859°°°	.2473•••	.2148
			.01	.0777•••	.0754•••	.0797•••	.0781•••	.0685	.0754•••	.0644
		1.0	.10	.6300	.7159•••	.6873•••	.6706•••	.4775°°°	.7159•••	.6323
			.05	.5012	.5587•••	.5502•••	.5335•••	.4021°°°	.5813•••	.4927
			.01	.2688•••	.2870•••	.2825•••	.2727•••	.2195°°°	.2870•••	.2434
	15	0.0	.10	.1024	.0906°°°	.0982	.0984	.1021	.0973	.1039
			.05	.0490	.0448°°	.0457°°	.0471	.0481	.0448°°	.0464°
			.01	.0092	.0086	.0097	.0091	.0096	.0086	.0078°°
		0.5	.10	.3538	.3968•••	.3891•••	.3810•••	.2782°°°	.4121•••	.3567
			.05	.2383	.2718•••	.2598•••	.2521•••	.1937°°°	.2718•••	.2322
			.01	.0872•••	.0943•••	.0937•••	.0911•••	.0734	.0943•••	.0737
		1.0	.10	.6768	.7717•••	.7482•••	.7305•••	.4704°°°	.7827•••	.6803
			.05	.5536	.6573•••	.6199•••	.5996•••	.4057°°°	.6573•••	.5503
			.01	.3160•••	.3709•••	.3461•••	.3332•••	.2396°°°	.3709•••	.2896

Table 7.2: Empirical Sizes and Sample Powers —< Continued >—

	n	θ	α	f	w	ns	ls	cs	aw	t
(U)	9	0.0	.10	.0949°	.0877°°°	.0928°°	.0936°°	.0977	.0877°°°	.0931°°
			.05	.0445°°	.0420°°°	.0438°°°	.0438°°°	.0469	.0420°°°	.0446°°
			.01	.0096	.0084	.0092	.0094	.0091	.0084	.0109
		0.5	.10	.3329	.3148°°	.3649•••	.3777•••	.4315•••	.3148°°	.3299
			.05	.2035	.1938°	.2291•••	.2373•••	.2686•••	.1938°	.2036
			.01	.0529°	.0503°°	.0589	.0614	.0666••	.0503°°	.0583
		1.0	.10	.6795	.6287°°°	.6946•••	.7099•••	.7768•••	.6287°°°	.6773
			.05	.5206	.4707°°°	.5415•••	.5542•••	.6180•••	.4707°°°	.5209
			.01	.2206°°	.2025°°°	.2316	.2377	.2621•••	.2025°°°	.2326
	12	0.0	.10	.0949°	.0927°°	.0940°°	.0942°	.0970	.0927°°	.0942°
			.05	.0461°	.0411°°°	.0454°°	.0453°°	.0477	.0474	.0462°
			.01	.0099	.0091	.0098	.0095	.0102	.0091	.0109
		0.5	.10	.3833	.3696°	.4318•••	.4531•••	.5485•••	.3696°	.3814
			.05	.2495	.2215°°°	.2873•••	.2990•••	.3630•••	.2421	.2495
			.01	.0783	.0647°°°	.0924•••	.0962•••	.1155•••	.0647°°°	.0821
		1.0	.10	.7822	.7281°°°	.8012•••	.8198•••	.8953•••	.7281°°°	.7803
			.05	.6422	.5721°°°	.6703•••	.6894•••	.7834•••	.5960°°°	.6423
			.01	.3274	.2817°°°	.3548••	.3697•••	.4371•••	.2817°°°	.3377
	15	0.0	.10	.0912°°°	.0860°°°	.0947°	.0953	.0973	.0936°°	.0897°°°
			.05	.0437°°°	.0423°°°	.0447°°	.0443°°°	.0465	.0423°°°	.0437°°°
			.01	.0101	.0092	.0094	.0090	.0092	.0092	.0103
		0.5	.10	.4532	.4227°°°	.5172•••	.5400•••	.6643•••	.4392°	.4513
			.05	.3044	.2877°°	.3582•••	.3788•••	.4882•••	.2877°°	.3043
			.01	.0951	.0903°°	.1205•••	.1292•••	.1700•••	.0903°°	.1002
		1.0	.10	.8556	.8004°°°	.8760•••	.8939•••	.9532•••	.8140°°°	.8551
			.05	.7484	.6926°°°	.7764•••	.7982•••	.8941•••	.6926°°°	.7481
			.01	.4520	.3990°°°	.4926•••	.5146•••	.6272•••	.3990°°°	.4601
(Q)	9	0.0	.10	.0941°°	.0877°°°	.0928°°	.0936°°	.0977	.0877°°°	.0930°°
			.05	.0452°°	.0420°°°	.0438°°°	.0438°°°	.0469	.0420°°°	.0452°°
			.01	.0098	.0084	.0092	.0094	.0091	.0084	.0106
		0.5	.10	.3879	.3512°°°	.3857	.3922	.4068•••	.3512°°°	.3848
			.05	.2473	.2243°°°	.2465	.2490	.2609••	.2243°°°	.2477
			.01	.0733	.0657°°°	.0733	.0743	.0775	.0657°°°	.0769
		1.0	.10	.7750	.7219°°°	.7637°	.7710	.7949•••	.7219°°°	.7738
			.05	.6339	.5739°°°	.6255	.6320	.6499••	.5739°°°	.6342
			.01	.3097	.2765°°°	.3034°°	.3085	.3158	.2765°°°	.3192
	12	0.0	.10	.0939°°	.0927°°	.0940°°	.0942°	.0970	.0927°°	.0931°°
			.05	.0462°	.0411°°°	.0454°°	.0453°°	.0477	.0474	.0462°
			.01	.0102	.0091	.0098	.0095	.0102	.0091	.0107
		0.5	.10	.4493	.4176°°°	.4549	.4610•	.4848•••	.4176°°°	.4483
			.05	.3026	.2623°°°	.3084	.3142•	.3305•••	.2825°°°	.3025
			.01	.1064	.0879°°°	.1079	.1102	.1155	.0879°°°	.1101
		1.0	.10	.8646	.8193°°°	.8612	.8695	.8854•••	.8193°°°	.8640
			.05	.7581	.6841°°°	.7529	.7641	.7926•••	.7040°°°	.7580
			.01	.4596	.3905°°°	.4550	.4654	.4922•••	.3905°°°	.4663
	15	0.0	.10	.0916°°°	.0860°°°	.0947°	.0953	.0973	.0936°°	.0910°°°
			.05	.0438°°°	.0423°°°	.0447°°	.0443°°°	.0465	.0423°°°	.0437°°°
			.01	.0099	.0092	.0094	.0090	.0092	.0092	.0104
		0.5	.10	.5259	.4755°°°	.5340	.5463•••	.5676•••	.4956°°°	.5244
			.05	.3765	.3375°°°	.3784	.3904••	.4155•••	.3375°°°	.3764
			.01	.1372	.1153°°°	.1405	.1448	.1571•••	.1153°°°	.1403
		1.0	.10	.9227	.8825°°°	.9234	.9305••	.9406•••	.8921°°°	.9224
			.05	.8508	.7999°°°	.8454	.8587	.8883•••	.7999°°°	.8508
			.01	.6037	.5374°°°	.6073	.6191	.6605•••	.5374°°°	.6080

Moreover, it is shown from Table 7.2 that the *t* test, *t*, is not robust. In the case where the population distribution is normal, *t* performs well. However, in the rest of the cases, *t* is not very good. Especially, when the population distribution is (C), *t* does not work at all. The empirical sizes are underestimated for most of the cases.

The empirical sizes of both the Wilcoxon test, *w*, and the asymptotic Wilcoxon test, *aw*, are underestimated for a lot of cases. However, all the cases of $\alpha = .01$ show a good performance for both tests, i.e., the empirical sizes are properly estimated in the case of $\alpha = .01$ of *w* and *aw*, but not in the case of $\alpha = .10,\ .05$ of *w* and *aw*.

In the case of the normal population (N), all the tests except for *w* and *aw* perform well. Note that *t* is a uniform powerful test in the case of (N). However, in the case of (L), (U) and (Q), we can see that *cs* is the only one which does not have size distortion.

In Table 7.1, the asymptotic relative efficiencies of *cs* relative to *t* approach zero except for the Cauchy population. However, we have obtained the result that for all the population distributions *cs* shows the best performance in the sense of the empirical size criterion.

Sample Power ($\theta \neq 0$): Next, we discuss several tests in the sample power criterion, which is related to the case of $\theta \neq 0$. The sample powers of *f*, *w*, *ns*, *ls*, *cs* and *aw* are compared with the sample power of *t*.

From Table 7.2, *ns* is almost equivalent to *ls* in the sense of the sample powers. Both have many cases which are significantly more powerful than the *t* test and few cases which are less powerful than the *t* test. For example, in all the cases of (B), (C), (G) and (T), the sample powers of *ns* and *ls* are significantly larger than those of *t*. *ns* and *ls* are less powerful than *t* when the population distribution is (N), and they are slightly less powerful than *t* when the population distribution is (L). Furthermore, Chernoff and Savage (1958) proved the theorem that the asymptotic relative efficiency of normal score test to the *t* test is more than one under the alternative hypothesis of shifting location parameter. From Table 7.2, this theorem generally holds even in the case of small sample. In the case of (L) and (Q) in Table 7.2, there is no significant difference between the *t* test and the normal score test. Thus, the *t* test and the normal score test are similar to each other, but the latter is slightly better than the former.

As for *w*, *cs* and *aw*, there are a lot of cases which are more powerful than *t* test, but at the same time there are many cases which are less powerful than *t* test. The sample powers of *cs* are smaller than those of *t* when the population is distributed as (D), (L), (N) and (T), while they are larger than those of *t* when (B), (C), (X), (U) or (Q) is taken for the underlying distribution. In particular, for (U) and (Q), *cw* shows the most powerful test of the seven tests. Moreover, from judging the sample power criterion, *cs* is inferior to *w*, *ns*, *ls* and *aw* in a lot of cases, e.g., (C), (X), (G), (D), (L), (N) and (T).

It is known that *aw* is asymptotically normal and equivalent to *w*. Even in the small sample, *w* is almost same as *aw* from Table 7.2. Therefore, we can conclude that the results in large sample are consistent with those in small sample. Moreover,

in the case of (C), (D) and (T), *w* and *aw* are more powerful than any other tests, but in the case of (U) and (Q), they are less powerful than any other tests. This result implies that *w* and *aw* are useful tests when the underlying distribution has diffuse tails.

In Table 7.2, we do not have the case where f is less powerful than *t*, but we have a few cases where f is more powerful than *t*. In other words, f is very similar to *t*. For (C), (X), (D) and (T), f is relatively superior to *t*. Especially, f is much more powerful than *t* for (C). For (N), although all the score tests are less powerful than *t* test, f has the almost same power as *t*. As discussed in Bradley (1968), it is known that both Fisher and *t* tests are asymptotically equivalent. We can see that this result is shown even in the case of small sample.

We consider the case of normal population, i.e., (N). It is well known that *t* gives us the most powerful test when the population follows a normal distribution (N). This fact is also supported by Table 7.2. That is, for the case of $\theta \neq 0$ in (N), the superscript $^\circ$ is attached with all the values except for f. Hodges and Lehmann (1956) and Chernoff and Savage (1958) showed that the Wilcoxon test is as powerful as *t* test under a shift of location parameter for any population distribution. Remember that the asymptotic relative efficiency of the Wilcoxon test relative to the *t* test is 0.955, which is close to one. See Table 7.1 for the asymptotic relative efficiency. However, it is shown from (N) in Table 7.2 that the Wilcoxon test is significantly less powerful than the *t* test. Moreover, as discussed in Bradley (1968), the *t* test is asymptotically equivalent to the Fisher test. This result is consistent with (N) in Table 7.2.

When the underlying distribution is (C), all the nonparametric tests are more powerful than the *t* test. Furthermore, the most powerful nonparametric test is given by *w* and *aw* in the case of (C).

Now, we compare the *t* test, the Wilcoxon test and the Fisher test. Fisher's two-sample test uses more information than the Wilcoxon test in the sense that the Fisher test utilizes original data while Wilcoxon test utilizes the ranked data. Accordingly, it is easily expected that in small sample the Fisher test is more powerful than the Wilcoxon test. However, the Monte-Carlo experiment shows that we have both cases; one cases are that the Wilcoxon test is more powerful than the Fisher test and the *t* test, and another ones are that the Wilcoxon test is less powerful than the Fisher test and the *t* test. The Fisher test is very similar to the *t* test in the empirical size and sample power criteria. In Table 7.1, we take some examples of the underlying distributions such that the Wilcoxon test has a larger asymptotic relative efficiency than the *t* test, which are (B), (C), (X), (G), (D), (L) and (T). This is consistent with the result obtained in Table 7.2, except for (L). As a whole, there are some cases in which the Fisher test is between the Wilcoxon test and the *t* test with respect to the order of sample power.

We examine the alternative hypothesis of $H_1 : F(x) = G(x - \theta)$, where we consider a shift in location parameter. In the case of logistic distribution, it is known that the asymptotic relative efficiency of Wilcoxon test to normal score test is 1.047 (see Kendall and Stuart (1979)), which implies that *w* is almost equivalent to *ns*. This result is consistent with *w* and *ns* in (L) of Table 7.2. When the underlying distribution

is Cauchy, it is known that the asymptotic relative efficiency of Wilcoxon test to normal score test is 1.413 (see Kendall and Stuart (1979)). In this case, the Wilcoxon test have much more power than the normal score test as n is large.

From the fact that the Fisher test is as powerful as the t test, it is known that the t test is a nonparametric test in a sense. As shown in Bradley (1968), if the original distribution has the fourth moment, the significance point of the t test does not depend on a functional form when the sample size is large. Therefore, we observe from Table 7.2 that f is similar to t in a lot of cases.

It is known that the Wilcoxon test has more asymptotic relative efficiency than the t test as the tails of distribution are large (see Mehta and Patel (1992)). It is shown from the Monte-Carlo experiments in Table 7.2 that this asymptotic property holds in the case of the small sample. That is, for (C), (D) and (T), clearly the Wilcoxon test is better than the t test.

Intuitively, it is expected from an amount of information set included in the test statistics that the Fisher test generally performs better than the Wilcoxon test in the sample power. However, judging from the results obtained in Table 7.2, sometimes the Wilcoxon test is more powerful. A reason of the results comes from the following three facts:

(i) Both the t and Fisher tests take difference between the two-sample means as the test statistics.

(ii) Both the Fisher and Wilcoxon tests are the nonparametric tests based on combination.

(iii) For distributions with fat tails, the Wilcoxon test has more asymptotic relative efficiency than the t test.

From these three facts, we can consider that the Fisher test is between the t test and the Wilcoxon test in the sense of the sample power. Accordingly, it might be appropriate that the order of the three sample powers is given by:

$$\begin{aligned} &\text{Wilcoxon} \geq \text{Fisher} \geq t, && \text{for distributions with fat tails,} \\ &\text{Wilcoxon} \leq \text{Fisher} \leq t, && \text{otherwise.} \end{aligned}$$

In a lot of population distributions, f is different from w, but f is very close to t.

As a supplement to Table 7.2, Figure 7.1 is displayed, where the sample powers are shown for $n1 = n2 = n = 12$, $\alpha = 0.1$, $\theta = 0.0, 0.1, \cdots, 1.0$ and ten population distributions. Note that in Table 7.2 only the cases of $\theta = 0.0, 0.5, 1.0$ are represented. Since each figure has seven lines, sometimes it not easy to distinguish each line. For each population distribution, we have the following features:

(B) There are three lines which we can differentiate. The upper lines are constructed from ns, ls and cs, the middle ones are w and aw, and the lower lines are f and t. Thus, ns, ls and cs are powerful, but f and t are poor.

Figure 7.1: Sample Powers: $n = 12$ and $\alpha = 0.10$

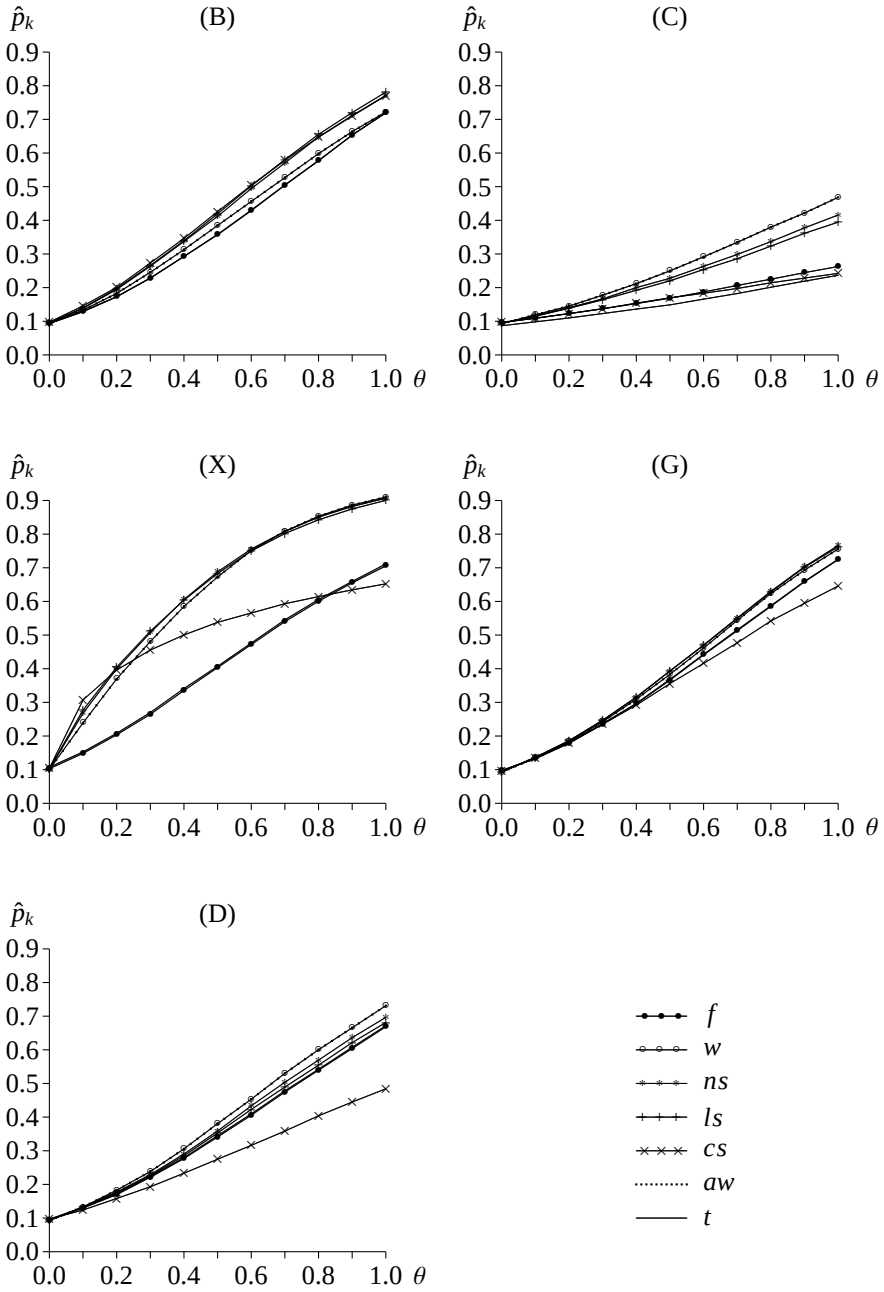
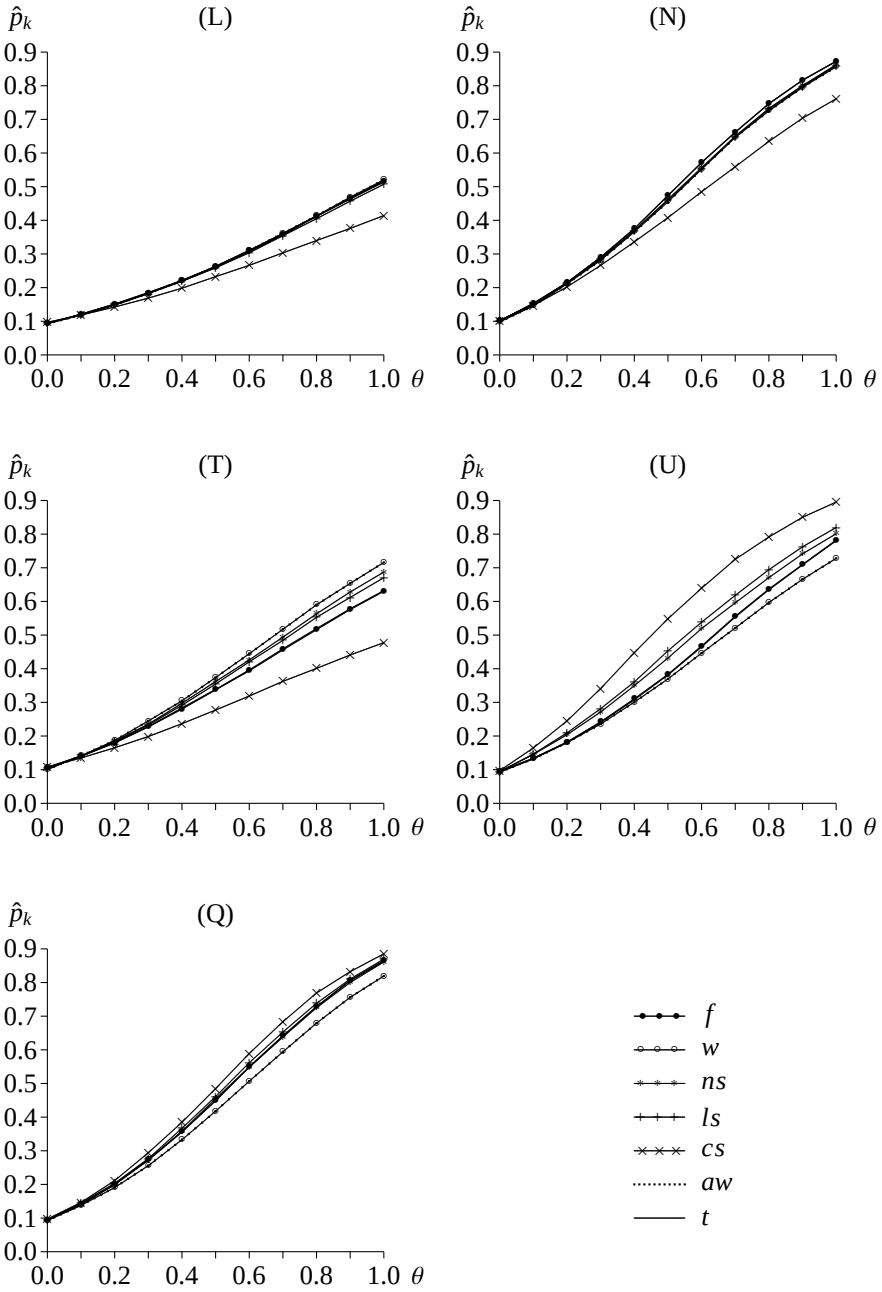


Figure 7.1: Sample Powers: $n = 12$ and $\alpha = 0.10$ —< Continued >—



(C) w and aw are very close to each other, which are the most powerful. We can distinguish the rest of lines, i.e., six lines, in (C). t is less powerful than any other tests for all θ.

(X) f is very close to t. w is on top of aw. ns and ls are very close to each other. cs is the most powerful for samll θ but the worst for large θ. Thus, in (X), we can see four lines to be distinguished. ns and ls are relatively good for all θ.

(G) The upper lines are given by w, ns, ls and aw. The middle lines are f and t. The lower line is cs.

(D) There are five lines to be distinguished. From the top, we have w, ns, ls, f and cs, where aw is on top of w and t is on f.

(L) All the lines except for cs are very close to each other. cs is less powerful than the other tests as θ is away from zero.

(N) There are three lines to be distinguished. The two lines from the top are quite close, but the first line which consists of f and t is slightly more powerful than the second one which is given by w, ns, ls and aw. cs shows the least powerful test.

(T) We can see five lines in (T). From the top, we have w, ns, ls, f and cs. We cannot distinguish w from aw and similarly f from t.

(U) There are five lines. We have cs, ls, ns, f and w from the top, where aw and t are on w and f, respectively.

(Q) The lower lines are given by w and aw. The upper one is cs. The middle lines consist of f, ns, ls and t, where ls is slightly larger than f, ns and t.

Thus, cs is extreme, because it gives us either the most powerful or the least powerful in a lot of population distributions. cs is the most powerful for (B), (U) and (Q), but it is the least powerful for (G), (D), (L), (N) and (T). It is possible to distinguish w from f, but it is not easy to see difference between f and t.

Number of Combinations: In Table 7.2, it takes about 40 seconds to obtain one line in the case of $n = 15$, where Pentium III 1GHz Dual CPU personal computer, Windows 2000 SP2 operating system and Open Watcom C/C++32 Optimizing Compiler (Version 1.0) are used. As shown in Appendix 7.1, computational burden increases by $4 - 2/n$ when n increases by one, i.e., from $n - 1$ to n. Thus, computational time extraordinarily increases as n is large. Therefore, it is not easy to obtain all the possible combinations for large n. We consider taking M combinations out of all the possible ones (i.e., ${}_{n1+n2}C_{n1}$) randomly with equal probability (i.e., $1/{}_{n1+n2}C_{n1}$). Based on the M combinations, the probabilities $P(s < s_0)$, $P(s = s_0)$ and $P(s > s_0)$ can be computed for $s = f$, w, ns, ls, cs, aw. See Appendix 7.3 for the source code of the random combinations, which is written by C language.

The cases of $M = 10^4$, 10^5, 10^6, ${}_{n1+n2}C_{n1}$ are examined, where $n1 = n2 = 15$ is taken. The results are shown in Table 7.3, where we take the case in which the

Table 7.3: Number of Combinations: (N) and $n1 = n2 = 15$

M	θ	α	f	w	ns	ls	cs
10^4	0.0	.10	.0959	.0966	.0964	.0979	.0963
		.05	.0507	.0482	.0502	.0500	.0491
		.01	.0101	.0094	.0105	.0106	.0105
	0.5	.10	.5313	.5057	.5229	.5213	.4443
		.05	.3825	.3599	.3760	.3759	.3283
		.01	.1462	.1365	.1442	.1449	.1279
	1.0	.10	.9260	.9083	.9160	.9143	.7955
		.05	.8546	.8293	.8441	.8398	.7399
		.01	.6120	.5854	.5996	.5953	.5313
10^5	0.0	.10	.0969	.0952	.0961	.0987	.0964
		.05	.0512	.0491	.0505	.0504	.0486
		.01	.0099	.0090	.0106	.0107	.0106
	0.5	.10	.5312	.5027	.5221	.5218	.4443
		.05	.3815	.3631	.3748	.3760	.3280
		.01	.1463	.1354	.1424	.1443	.1286
	1.0	.10	.9258	.9068	.9164	.9146	.7956
		.05	.8541	.8320	.8444	.8391	.7397
		.01	.6143	.5831	.5992	.5958	.5307
10^6	0.0	.10	.0971	.0941	.0961	.0984	.0966
		.05	.0514	.0493	.0507	.0505	.0488
		.01	.0096	.0090	.0104	.0104	.0105
	0.5	.10	.5316	.5002	.5222	.5215	.4441
		.05	.3818	.3635	.3745	.3757	.3286
		.01	.1461	.1346	.1433	.1440	.1291
	1.0	.10	.9261	.9052	.9166	.9146	.7955
		.05	.8537	.8324	.8441	.8397	.7397
		.01	.6141	.5821	.5992	.5967	.5313
$_{n1+n2}C_{n1}$	0.0	.10	.0968	.0941	.0960	.0984	.0966
		.05	.0514	.0493	.0505	.0505	.0487
		.01	.0096	.0090	.0105	.0106	.0105
	0.5	.10	.5313	.5001	.5222	.5215	.4443
		.05	.3820	.3635	.3744	.3754	.3283
		.01	.1459	.1346	.1432	.1438	.1284
	1.0	.10	.9261	.9052	.9165	.9146	.7956
		.05	.8540	.8324	.8441	.8396	.7400
		.01	.6141	.5821	.5987	.5963	.5314

population distribution is assumed to be normal, i.e., (N). Therefore, $M = {}_{n1+n2}C_{n1}$ in Table 7.3 is equivalent to $n = 15$ in (N) of Table 7.2. In Table 7.3, the maximum difference between $M = 10^4$ and $M = {}_{n1+n2}C_{n1}$ is 0.0056 in absolute value, which is the case of $\theta = 0.5$, $\alpha = .10$ and w. All the distances between $M = 10^5$ and $M = {}_{n1+n2}C_{n1}$ are less than 0.0026, where the maximum value 0.0026 is obtained when we choose $\theta = 0.5$, $\alpha = .10$ and w. Comparing $M = 10^6$ with $M = {}_{n1+n2}C_{n1}$, the maximum value is given by 0.0007 in absolute value (see $\theta = 0.5$, $\alpha = .01$ and cs), which is very small. Thus, the case of $M = 10^6$ is almost same as that of $M = {}_{n1+n2}C_{n1}$. Because we have ${}_{n1+n2}C_{n1} \approx 1.55 \times 10^8$ for $n1 = n2 = 15$, 10^6 corresponds to 0.6% of 1.55×10^8, which implies that only 0.6% of all the possible combinations are large enough for statistical inference. We do not have to compute the nonparametric test statistics for all the possible combinations. Therefore, it is shown from Table 7.3 that we can reduce computational burden.

7.5 Empirical Example: Testing Structural Changes

In this section, we take an example of the standard linear regression model. In a regression analysis, usually, the disturbance term is assumed to be normal and we perform testing a hypothesis. However, sometimes the normality assumption is too strong. In this chapter, loosening the normality assumption, we test a structural change without assuming any distribution for the disturbance term. Consider the following standard linear regression model:

$$y_t = x_t\beta + \epsilon_t, \qquad t = 1, 2, \cdots, n,$$

where y_t, x_t, β and ϵ_t are a dependent variable at time t, a $1 \times k$ vector of independent variable at time t, a $k \times 1$ unknown parameter vector to be estimated, and the disturbance term at time t with mean zero and variance σ^2, respectively. The sample size is given by n.

Let us define $X_{t-1} = (x_1' \; x_2' \; \cdots \; x_{t-1}')'$ and $Y_{t-1} = (y_1 \; y_2 \; \cdots \; y_{t-1})'$. β_{t-1} denotes the OLS estimate of β using the data up to time $t-1$, i.e., $\beta_{t-1} = (X_{t-1}'X_{t-1})^{-1}X_{t-1}'Y_{t-1}$. Let us define the predicted error as:

$$\omega_t = \frac{y_t - x_t\beta_{t-1}}{\sqrt{1 + x_t(X_{t-1}'X_{t-1})^{-1}x_t'}},$$

for $t = k+1, k+2, \cdots, n$. The predicted error ω_t can be estimated by recursive ordinary least squares estimation, which is distributed with mean zero and variance σ^2. This predicted error is called the **recursive residual**, which can be recursively obtained. The recursive algorithm is given by equations (6.53) – (6.59) in Appendix 6.3, p.376, which are the standard Kalman filter algorithm. The recursive residual is given by $y_t - y_{t|t-1}$ in equation (6.58), where $Z_t = x_t$, $\alpha_t = \beta_t$, $d_t = 0$, $S_t = 1$, $T_t = I_k$, $c_t = 0$ and

$Q_t = 0$. The recursive residuals ω_t, $t = k+1, k+2, \cdots, n$, are mutually independently distributed and normalized to mean zero and variance σ^2.

Based on the recursive residual ω_t, we perform testing the structural change. We can judge the structural change if the structure of the recursive residuals change in a period. Dividing the sample into two groups, We test if both $\{\omega_t\}_{t=k+1}^{n1}$ and $\{\omega_t\}_{t=n1+1}^{n}$ are generated from the same distribution. The null hypothesis is represented by H_0 : $F(\omega) = G(\omega)$ while the alternative one is H_1 : $F(\omega) \neq G(\omega)$. Let $F(\cdot)$ be the distribution of the first $n1$ recursive residuals and $G(\cdot)$ be that of the last $n2$ recursive residuals.

Why do we use the recursive residual, not the conventional OLS residual? Note that the OLS residuals have the following problem. Let e_t be the OLS residuals obtained from the regression equation $y_t = x_t\beta + \epsilon_t$, $t = 1, 2, \cdots, n$, i.e., $e_t = y_t - x_t\hat{\beta}$, where $\hat{\beta}$ denotes the OLS estimate. The OLS residuals e_t, $t = 1, 2, \cdots, n$, are not mutually independently distributed. Clearly, we have $\mathrm{E}(e_s e_t) \neq 0$ for $s \neq t$. Therefore, we utilize the recursive residuals which are mutually independently distributed.

We take an example of Japanese import function. Annual data from *Annual Report on National Accounts* (Economic and Social Research Institute, Cabinet Office, Government of Japan) is used. Let GDP_t be Gross Domestic Product (1990 price, billions of Japanese yen), M_t be Imports of Goods and Services (1990 price, billions of Japanese yen), and P_t be Terms of Trade Index, which is given by Gross Domestic Product Implicit Price Deflator divided by Imports of Goods and Services Implicit Price Deflator.

The following two import functions are estimated and the recursive residuals are computed:

$$\log M_t = \beta_1 + \beta_2 \log GDP_t + \beta_3 \log P_t,$$

$$\log M_t = \beta_1 + \beta_2 \log GDP_t + \beta_3 \log P_t + \beta_4 \log M_{t-1},$$

where β_1, β_2, β_3 and β_4 are the unknown parameters to be estimated. The estimation results by OLS are as follows:

$$\log M_t = \underset{}{-\ 6.01226} + \underset{(0.513743)}{1.28407}\ \log GDP_t + \underset{(0.040349)}{0.19698}\ \log P_t, \qquad (7.19)$$
$$\underset{(0.077940)}{}$$

$$R^2 = 0.989839, \quad \overline{R}^2 = 0.989367,$$
$$DW = 0.692777, \quad \mathrm{SER} = 0.103416,$$
$$\text{Estimation Period: } 1955 - 2000,$$

$$\log M_t = \underset{}{-\ 1.23866} + \underset{(0.816943)}{0.41114}\ \log GDP_t + \underset{(0.137635)}{0.17850}\ \log P_t + \underset{(0.055513)}{}$$
$$+\ \underset{(0.095841)}{0.61752}\ \log M_{t-1}, \qquad (7.20)$$

$$R^2 = 0.994500, \quad \overline{R}^2 = 0.994097,$$
$$DW = 1.93377 \text{ (Durbin's h} = -.012643), \quad \text{SER} = 0.073489,$$
$$\text{Estimation Period:} \quad 1956 - 2000,$$

where the values in the parentheses are the standard errors. R^2, $\overline{R}^2$, DW and SER are the coefficient of multiple determination, the adjusted R^2, Durbin=Watson statistic, and the standard error of the disturbance, respectively. The recursive residuals are obtained from 1958 to 2000 (43 periods) for equation (7.19) and from 1960 to 2000 (41 periods) for equation (7.20).

In equation (7.19), the price elasticity is estimated as 0.19698, which value is quite small. This small price elasticity represents the feature of Japanese economy. Japan has few natural resources such as fuel and food, which are necessities. Therefore, Japanese import is not sensitive to volatility in these prices. Thus, from equation (7.19), we can see that Japanese import is influenced by an income effect, rather than a price effect.

For each regression equation, we compute the recursive residuals, divide the period into two groups, and test if the recursive residuals in the first period are the same as those in the last period. The results in equations (7.19) and (7.20) are in Tables 7.4 and 7.5, respectively. In the tables, t, f, w, aw, ns, ls, cs and $F(\cdot,\cdot)$ denote the t test, the Fisher test, the Wilcoxon test, the asymptotic Wilcoxon test, the normal score test, the logistic score test, the Cauchy score test and the F test. The F test is conventionally used for testing the structural change, which is discussed in Appendix 7.4. Moreover, each test statistic is given by t_0, f_0, w_0, aw_0, ns_0, ls_0, cs_0 and F_0. The probability less than the corresponding test statistics are p-values. For the number of combinations in the nonparametric tests, $M = 10^7$ is taken when the number of all the possible combinations is greater than 10^7 (i.e., during the periods 1964 – 1993 in Table 7.4 and 1966 – 1993 in Table 7.5), and $M = {}_{n1+n2}C_{n1}$ is chosen otherwise. See Table 7.3 for comparison of the number of combinations. The results of the F test are also in Tables 7.4 and 7.5. The F test statistic is denoted by F_0 and the p-value corresponding to F_0 in given by p-values in Tables 7.4 and 7.5.

From the import function (7.19) in Table 7.4, the structural change occurs during the period 1986 – 1998 for the Fisher test, the Wilcoxon test, the normal score test, the logistic score test and the t test, the period 1987 – 1998 for the Cauchy score test, the period 1986 – 1997 for the asymptotic Wilcoxon test, and the period 1978 – 1985, 1991, 1992 for the F test, where the significance level is 1%. Figure 7.2 is drawn from Table 7.4. The lines are classified into four groups; (i) $F(3, 40)$, (ii) f and t, (iii) cs and (iv) w. ns, ls and aw. f is similar to t in the movement. w, ns, ls and aw also show a similar movement. Comparing (ii), (iii) and (iv), (ii) is the largest and (iii) is the smallest.

From the import function (7.20) in Table 7.5, the structural change occurs during the period 1986 – 1988, 1992 – 1995 for the Fisher test, the period 1987, 1988, 1993, 1994 for the Wilcoxon test, the period 1987, 1992 – 1995 for the normal score test and the logistic score test, the period 1993, 1994 for the Cauchy score test, the period

Table 7.4: Testing Structural Change by Nonparametric Tests: p-Values

Import Function (7.19)

Year	f	w	ns	ls	cs	t	aw	$F(3,40)$
1957	—	—	—	—	—	—	—	.6766
1958	.4651	.4651	.4651	.4651	.4651	.5304	.4679	.7842
1959	.3621	.2658	.3123	.3223	.3544	.3823	.2629	.7981
1960	.4184	.3014	.3442	.3548	.3993	.4303	.3000	.7798
1961	.4010	.2524	.3066	.3213	.3947	.4088	.2517	.8172
1962	.5627	.4489	.4764	.4817	.5029	.5690	.4548	.7258
1963	.6160	.5340	.5402	.5390	.5280	.6207	.5419	.6697
1964	.7113	.6684	.6474	.6371	.5718	.7139	.6774	.5181
1965	.7356	.6825	.6569	.6453	.5694	.7376	.6912	.4995
1966	.7941	.7736	.7318	.7142	.5938	.7948	.7812	.4580
1967	.7705	.7288	.6952	.6801	.5758	.7719	.7365	.4637
1968	.7933	.7667	.7245	.7069	.5812	.7942	.7738	.4933
1969	.8302	.8226	.7717	.7509	.5935	.8304	.8284	.5802
1970	.7885	.7560	.7149	.6977	.5713	.7891	.7624	.6150
1971	.7211	.6448	.6208	.6102	.5373	.7223	.6513	.6664
1972	.6521	.5350	.5331	.5301	.5106	.6535	.5406	.8017
1973	.4965	.3780	.3928	.3978	.4555	.4979	.3815	.7284
1974	.4428	.2983	.3303	.3411	.4402	.4439	.3010	.9490
1975	.5450	.4273	.4410	.4439	.4770	.5461	.4316	.9323
1976	.5766	.4568	.4629	.4634	.4822	.5771	.4610	.9618
1977	.5959	.4665	.4700	.4698	.4842	.5962	.4709	.9789
1978	.5819	.4381	.4491	.4513	.4798	.5818	.4420	.9927
1979	.5537	.3817	.4062	.4129	.4696	.5539	.3853	1.0000
1980	.6678	.5335	.5422	.5410	.5193	.6677	.5388	1.0000
1981	.7575	.6727	.6626	.6537	.5603	.7575	.6789	1.0000
1982	.8504	.8068	.7902	.7780	.6198	.8508	.8123	1.0000
1983	.9329	.9112	.9047	.8965	.7094	.9335	.9141	1.0000
1984	.9615	.9525	.9421	.9338	.7295	.9623	.9539	1.0000
1985	.9882	.9850	.9805	.9762	.7758	.9888	.9848	.9998
1986	.9991	.9974	.9982	.9982	.9947	.9993	.9970	.9886
1987	1.0000	.9997	.9999	.9999	1.0000	1.0000	.9996	.9070
1988	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	.9999	.8849
1989	1.0000	.9998	.9999	.9999	1.0000	1.0000	.9997	.9520
1990	.9999	.9994	.9998	.9998	.9999	.9999	.9990	.9866
1991	1.0000	.9996	.9999	.9999	1.0000	1.0000	.9992	.9902
1992	1.0000	.9999	1.0000	1.0000	1.0000	1.0000	.9997	.9909
1993	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	.9999	.9878
1994	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	.9999	.9883
1995	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	.9998	.9789
1996	.9999	.9999	.9998	.9998	.9988	.9999	.9989	.9309
1997	.9985	.9981	.9985	.9984	.9956	.9992	.9950	.8584
1998	.9967	.9956	.9967	.9967	.9967	.9981	.9877	—
1999	.9767	.9767	.9767	.9767	.9767	.9933	.9547	—

Table 7.5: Testing Structural Change by Nonparametric Tests: p-Values

Import Function (7.20)

Year	f	w	ns	ls	cs	t	aw	$F(4,37)$
1959	—	—	—	—	—	—	—	.9945
1960	.5854	.5854	.5854	.5854	.5854	.6916	.6323	.9924
1961	.5610	.5122	.5183	.5159	.5159	.5786	.5241	.9968
1962	.7527	.7405	.7161	.7059	.6595	.7614	.7583	.9987
1963	.7792	.7584	.7248	.7114	.6406	.7855	.7724	.9984
1964	.8449	.8400	.7934	.7746	.6597	.8467	.8499	.9969
1965	.8319	.8122	.7676	.7493	.6339	.8335	.8218	.9986
1966	.8852	.8880	.8388	.8180	.6606	.8845	.8939	.9980
1967	.8463	.8306	.7857	.7665	.6272	.8467	.8382	.9982
1968	.8550	.8204	.7757	.7568	.6155	.8549	.8276	.9981
1969	.8943	.8571	.8070	.7862	.6207	.8933	.8628	.9969
1970	.8406	.7794	.7373	.7202	.5886	.8404	.7865	.9977
1971	.7085	.6275	.5958	.5846	.5197	.7100	.6345	.9978
1972	.5698	.4727	.4630	.4603	.4683	.5721	.4777	.9983
1973	.3305	.2795	.2505	.2441	.2892	.3326	.2819	.9977
1974	.2868	.2226	.2103	.2088	.2875	.2885	.2243	.9983
1975	.4486	.3907	.3803	.3762	.3813	.4503	.3946	.9968
1976	.5212	.4427	.4191	.4110	.3929	.5224	.4474	.9949
1977	.5591	.4638	.4347	.4249	.3981	.5596	.4686	.9932
1978	.5296	.4230	.4047	.3985	.3930	.5300	.4274	.9942
1979	.4974	.3738	.3688	.3664	.3863	.4974	.3771	.9978
1980	.6843	.5666	.5849	.5872	.5699	.6834	.5724	.9828
1981	.7929	.7020	.6996	.6946	.6076	.7920	.7085	.9613
1982	.8827	.8291	.8186	.8099	.6597	.8825	.8345	.8952
1983	.9482	.9208	.9124	.9047	.7168	.9489	.9235	.6789
1984	.9480	.9147	.9082	.9010	.7190	.9489	.9177	.8102
1985	.9776	.9641	.9569	.9507	.7515	.9788	.9651	.5947
1986	.9959	.9920	.9945	.9947	.9896	.9965	.9916	.0466
1987	.9982	.9965	.9974	.9974	.9925	.9987	.9961	.0411
1988	.9959	.9917	.9945	.9948	.9904	.9967	.9913	.0393
1989	.9851	.9753	.9839	.9853	.9824	.9866	.9757	.2420
1990	.9723	.9539	.9733	.9767	.9801	.9743	.9554	.7875
1991	.9876	.9812	.9885	.9896	.9870	.9893	.9812	.7921
1992	.9953	.9925	.9950	.9952	.9912	.9964	.9918	.7926
1993	.9989	.9990	.9991	.9991	.9959	.9993	.9984	.6777
1994	.9989	.9986	.9990	.9990	.9965	.9993	.9977	.6868
1995	.9943	.9929	.9954	.9956	.9917	.9957	.9916	.5702
1996	.9661	.9665	.9770	.9786	.9777	.9703	.9675	.4068
1997	.9440	.9400	.9690	.9737	.9808	.9499	.9454	—
1998	.9817	.9805	.9902	.9902	.9915	.9861	.9771	—
1999	.9756	.9756	.9756	.9756	.9756	.9808	.9545	—

Figure 7.2: *p*-Values — Import Function (7.19): Table 7.3

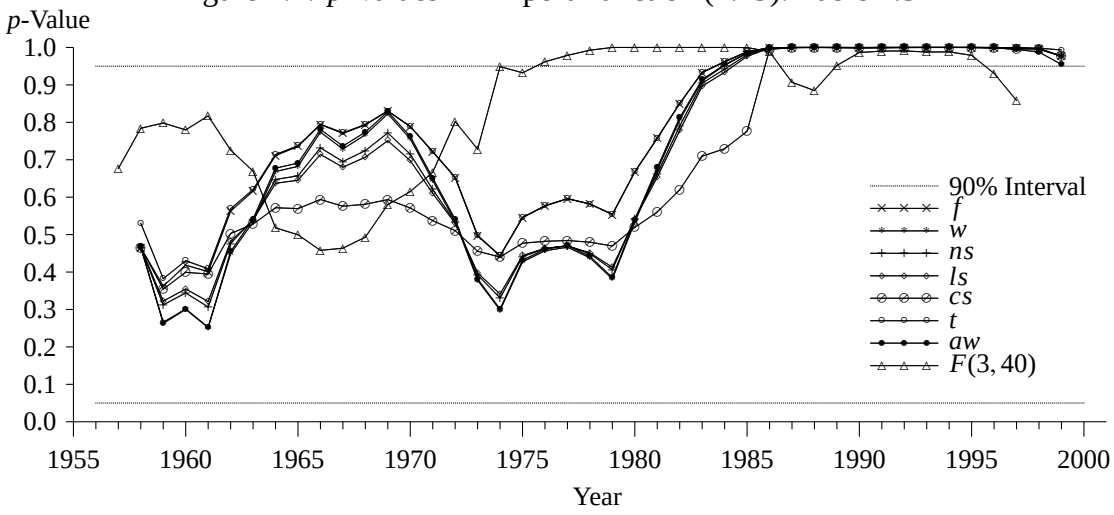
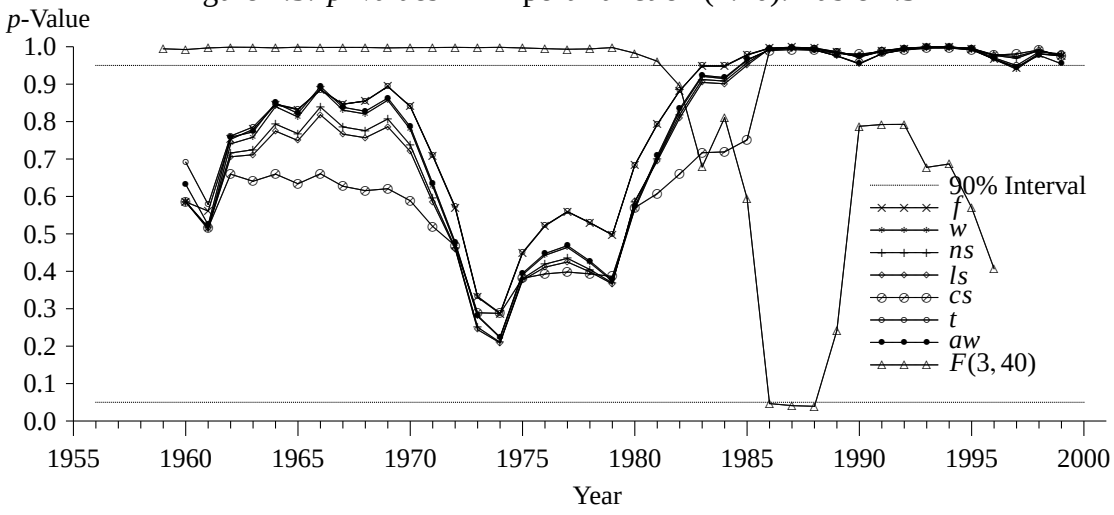


Figure 7.3: *p*-Values — Import Function (7.20): Table 7.5



1986 – 1988, 1992 – 1995 for the t test, the period 1987, 1993, 1994 for the asymptotic Wilcoxon test, and the period 1959 – 1979 for the F test, where the significance level is 1%. Figure 7.3 is based on Table 7.5. As in Figure 7.2, we can see in Figure 7.3 that we have four groups; (i) $F(4, 37)$, (ii) f and t, (iii) cs and (iv) w. ns, ls and aw. Comparing (ii), (iii) and (iv), we can see that (ii) is better than the other tests for most of the estimation periods.

For both import functions, the F test is very different from the other tests. The F test examines whether the regression coefficient changes during the period, while the other tests check if the structure of the predicted errors changes during the period. In addition, f and t show very similar movements for the p-values, which is consistent with Bradley (1968) who stated that the Fisher test and the t test are asymptotically equivalent. For almost all the periods in Figures 7.2 and 7.3, cs shows the smallest but f and t are the largest. In Table 7.2, we have seen that when the population distribution is normal both f and t show large powers but cs indicates a small power. Therefore, it might be concluded that the population distribution of the error term ϵ_t is close to a normal distribution in both equations (7.19) and (7.20).

7.6 Summary

Nonparametric test statistics have been approximated by a normal random variable from both computational and programming points of view. Recently, however, we can perform the exact test by progress of a personal computer. In this chapter, we have compared the empirical sizes and the sample powers in small sample, taking nonparametric two-sample tests, i.e, the score tests (Wilcoxon rank sum test, normal score test, logistic score test and Cauchy score test) and the Fisher test.

The results on the empirical sizes are summarized as follows. cs shows the best performance of all the nonparametric tests and the t test, because cs has no size distortion. For almost all the nonparametric tests, the empirical sizes are underestimated. Especially, the size distortion of w is observed in the most cases of all the tests.

As for the sample powers, in the case where we compare the t test, the Wilcoxon test and the Fisher test, it might be intuitively expected as follows:

(i) When the underlying distribution is normal, the t test gives us the most powerful test.

(ii) In the situations where we cannot use the t test, two nonparametric tests are more powerful than the t test, and moreover the Fisher test is better than the Wilcoxon test because the former utilizes more information than the latter. Accordingly, in the case where the underlying distribution is nonnormal, we might have the following relationship among the sample powers: $\hat{p}_t \leq \hat{p}_w \leq \hat{p}_f$.

However, the results of the Monte-Carlo simulations are: the Fisher test is as powerful as the t test and the Wilcoxon test is slightly less powerful but not too different

from the t test, even in the case where the two samples are identically and normally distributed. Moreover, when the underlying distribution is Cauchy, the Wilcoxon test is much better than the t test. In general, we have $\hat{p}_t \leq \hat{p}_f \leq \hat{p}_w$ when the population distribution is non-Gaussian. The fact proved by Chernoff and Savage (1958), which is the theorem that under the alternative hypothesis of a shifting location parameter the asymptotic relative efficiency of the normal score test relative to the t test is more than one, holds even in the small sample case, because ns is greater than t in most of the cases.

Finally, we take an example of testing structural change as an application to the nonparametric tests. we can see in Figures 7.2 and 7.3 that we have four groups; (i) $F(\cdot,\cdot)$, (ii) f and t, (iii) cs and (iv) w. ns, ls and aw. For both figures, the F test is very different from the other tests. The F test examines whether the regression coefficient changes during the period, while the other tests check if the structural change of the predicted errors occurs during the period. Moreover, f and t indicate very similar movements in the p-values, which is consistent with Bradley (1968). Note that, as discussed in Bradley (1968), the Fisher test and the t test are asymptotically equivalent. For almost all the periods in Figures 7.2 and 7.3, cs shows the smallest but f and t are the largest. In Table 7.2, we have seen that when the population distribution is normal both f and t show large powers but cs indicates a small power. Therefore, it might be concluded that the population distribution of the error terms in both Japanese import functions is close to a normal distribution.

Thus, using the nonparametric tests, we can test the hypothesis in spite of the functional form of distribution of the disturbances.

Appendix 7.1: On Generation of Combinations

It is not easy to write all the possible combinations on screen or paper. Harbison and Steele (1987) developed the source code, using the **bit operation** peculiar to C language, which is discussed in this appendix. We also introduce the source code where the **recursion** is utilized. The recursion is also the concept which characterizes C language (note that the recursion is not allowed in Fortran 77). In this appendix, the bit operation is compared with the recursion. It is shown that use of the recursion is much better than that of the bit operation from computational CPU time and simplicity of the source code.

Outline of Two Source Codes: Consider choosing $n1$ numbers out of the integers from 1 to N. We introduce two source codes to obtain the possible combinations and compare them from computational time.

Both Programs I and II indicate the source codes which take $n1$ numbers out of the positive integers up to N and display the $n1$ numbers on screen. Program I utilizes the bit operation, while Program II is based on the recursion. Note that both the bit operation and the recursion are allowed in C language, not in Fortran 77.

Table 7.6: Output of Programs I & II

Program I				Program II		
Output			Binary Number	Output		
1	2	3	000111	1	2	3
1	2	4	001011	1	2	4
1	3	4	001101	1	2	5
2	3	4	001110	1	2	6
1	2	5	010011	1	3	4
1	3	5	010101	1	3	5
2	3	5	010110	1	3	6
1	4	5	011001	1	4	5
2	4	5	011010	1	4	6
3	4	5	011100	1	5	6
1	2	6	100011	2	3	4
1	3	6	100101	2	3	5
2	3	6	100110	2	3	6
1	4	6	101001	2	4	5
2	4	6	101010	2	4	6
3	4	6	101100	2	5	6
1	5	6	110001	3	4	5
2	5	6	110010	3	4	6
3	5	6	110100	3	5	6
4	5	6	111000	4	5	6

Consider an example of the case $n1 = n2 = 3$. When we choose 3 numbers out of the integers from 1 to 6, the outputs are obtained in Table 7.6 for both Programs I and II. The outline of the two programs are concisely shown in the next paragraph.

In Program I of Table 7.6, the chosen figures in the left-hand side of Program I correspond to the digits represented by 1 in the right-hand side in Program I. In the first line, we have the binary number 000111 in the right-hand side of Program I, which implies that we choose {1, 2, 3} in the left-hand side of Program I. The second smallest binary number, which has three 1's, is given by the second line (i.e., 001011), where the corresponding figures are given by {1, 2, 4}. Similarly, in the third line, {1, 3, 4} are chosen based on the binary number 001101. This procedure is repeated until the binary number 111000 is obtained.

In Program II of Table 7.6, three numbers are taken from the six numbers 1 – 6, where the number in the right column is larger than that in the left column for each line and the number in the lower line is larger than that in the upper line for each column. That is, in the first — fourth lines, 1 and 2 are given to the first two numbers, and the last number is assigned to 3 in the first line, 4 in the second line, 5 in the third line and 6 in the fourth line.

Thus, depending on Programs I and II, the outputs should be different. Now, we discuss Programs I and II in more detail.

Program I (Bit Operation): From the N-digit binary number we take the $n1$ digits which include 1, and the numbers which correspond to the digits are chosen. This approach is introduced by Harbison and Steele (1987), which is concisely discussed as follows.

Take an example of $N = 6$ and $n1 = 3$. {3, 4, 6} is represented by the binary number 101100, where the digits taken out are given by 1. The six-digit binary number which includes three 1's and is the smallest six-digit binary number after 101100 is obtained as follows:

(i) Given the binary number which has three 1's (i.e., 101100 in this example), keep the first 1 from the right-hand side and replace the remaining 1's by 0. That is, we obtain 000100.

(ii) Add the original binary number to the binary number obtained in Step (i). In this case, we have 101100 + 000100 = 110000.

(iii) For the number obtained in Step (ii), keep the last 1 and replace the remaining 1's by 0. That is, 010000 is obtained.

(iv) Divide the number obtained in Step (iii) by the number in Step (i). We obtain 010000/000100 = 000100.

(v) Divide the number obtained in Step (iv) by 000010. Then, 000100/000010 = 000010 is obtained.

(vi) Subtract 1 from the number obtained in Step (v). Thus, we have 000010 – 000001 = 000001.

(vii) Add the number obtained in Step (vi) to that in Step (ii). Then, the next combination which we want is given by the digits with 1. In this example, we obtain 000001 + 110000 = 110001, which corresponds to the combination {1, 5, 6}.

In the case of $n1 = n2 = 3$, the initial binary number is given by 000111, which implies that we take the combination {1, 2, 3}. Repeating Steps (i) – (vii), all the binary numbers which include three 1's can be obtained.

The above approach is a very nice procedure to generate all the possible combinations (i.e., ${}_NC_{n1}$ combinations) given any $n1$ and $n2$. In C language, the bit operation (i.e., the binary number operation) is possible. Therefore, the source code to this problem is shown as follows.

comb1(n1,n2)

```
1: #define first(n) ( (unsigned int)( ( 1U<<(n) ) - 1U ) )
2: void comb1(int n1,int n2)
3: {
4:     int i,j,k,m[101];
5:     unsigned int x,s;
6:     unsigned int smallest,ripple,new_smallest,ones;
7:
```

```
 8:      i=1; x=first(n1);
 9:      while( ! (x & ~first(n1+n2)) ) {
10:             s=x; k=1;
11:             for( j=1;j<=n1+n2;j++ ) {
12:                 if( s & 1 ) {
13:                     m[k]=j;
14:                     k++;
15:                 }
16:                 s >>= 1;
17:             }
18:          for(k=1;k<=n1;k++) printf(" %2d",m[k]);
19:          printf("\n");
20:             smallest = x & -x;
21:             ripple = x + smallest;
22:             new_smallest = ripple & -ripple;
23:             ones = ( ( new_smallest/smallest ) >> 1 ) - 1;
24:             x= ripple | ones;
25:          i++;
26:      }
27: }
```

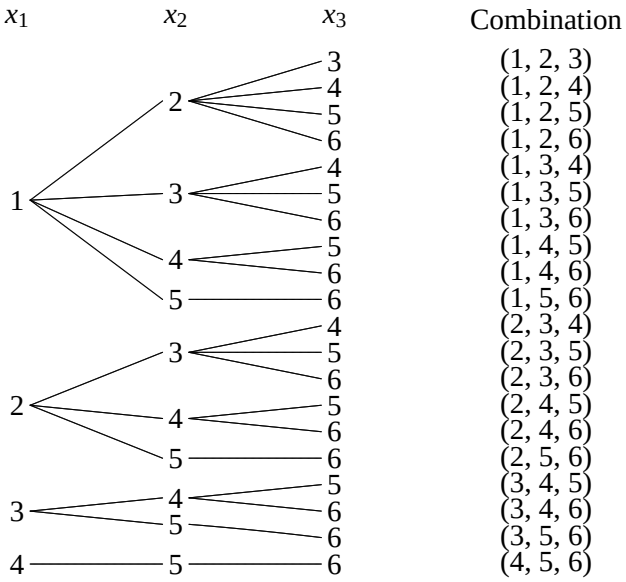
The relationship between Steps (i) – (vii) and the above source code is as follows. Line 20 corresponds to Step (i), Line 21 represents Step (ii), Line 22 is given by Step (iii), Line 23 indicates Steps (iv) – (vi), and Line 24 implies Step (vii).

Program II (Recursion): Program I is based on the binary number, but Program II utilizes the recursion. As the same example as above, i.e., $n1 = n2 = 3$, consider how to choose three different numbers out of the integers 1 – 6. The tree diagram is shown in Figure 7.4.

Suppose that three numbers out of the six integers 1 to 6 are assigned to x_1, x_2 and x_3, where $x_1 < x_2 < x_3$ is satisfied. For example, we have four combinations of $x_3 = 3, 4, \cdots, 6$ when $x_1 = 1$ and $x_2 = 2$ are taken, and there are three combinations of $x_3 = 4, 5, 6$ when $x_1 = 1$ and $x_2 = 3$ are assigned. Based on this idea, the same computational procedure is nested $n1$ times in the program which takes $n1$ numbers out of the integers 1 to N. That is, x_1 can take 1 to 4, x_2 can take $x_1 + 1$ to 5, and x_3 can take $x_2 + 1$ to 6. Thus, for each x_i of x_1, x_2, $\cdots$, x_{n1}, the same computational procedure is required.

Using the recursion, we can show the source code. The recursion as well as the bit operation are peculiar to C language. However, the Fortran compiler which allows the recursion is also available, e.g., Open WATCOM FORTRAN 77/32 Optimizing Compiler (Version 1.0). See `http://www.openwatcom.org` for the WATCOM compilers. However, since Fortran 77 does not allow the recursion in general, here we utilize C language. Because the number of nests is given by $n1$, Program II is improved computationally over Program I. Note that Lines 11 – 17 and 20 – 24 in Program I have to be performed for all the combinations, i.e., ${}_N C_{n1}$ combinations, and accordingly Program I is more computational than Program II. Based on Figure 7.4, Program II is represented as follows.

Figure 7.4: Tree Diagram



comb2(n1,n2)

```
1: void comb2(int n1,int n2)
2: {
3:     int  m[101];
4:     int  nest,column;
5:     void nest0(int nest,int column,int n1,int n2,int m[]);
6:
7:     nest=0; column=1;
8:     nest0(nest,column,n1,n2,m);
9: }
10: /* ---------------------------------------------------- */
11: void nest0(int nest,int column,int n1,int n2,int m[])
12: {
13:     int  i,k;
14:     void nest0();
15:
16:     for(i=nest+1;i<=n2+column;i++){
17:        m[column]=i;
18:        if( n1 != column )
19:           nest0(i,column+1,n1,n2,m);
20:        else {
21:           for(k=1;k<=n1;k++) printf(" %2d",m[k]);
22:           printf("\n");
23:        }
24:     }
25: }
```

Table 7.7: CPU Time (Seconds)

	Program I					Program II				
$n1$ \ $n2$	11	12	13	14	15	11	12	13	14	15
11	0.158	0.313	0.596	1.096	1.952	0.034	0.062	0.108	0.184	0.305
12	0.314	0.648	1.284	2.454	4.541	0.069	0.131	0.239	0.423	0.728
13	0.598	1.285	2.650	5.260	10.094	0.135	0.265	0.504	0.926	1.653
14	1.100	2.461	5.268	10.843	21.551	0.254	0.520	1.025	1.953	3.609
15	1.964	4.562	10.125	21.583	44.369	0.461	0.980	2.003	3.951	7.553

1) Each value in Table 7.7 represents the average of CPU time (seconds) from 1000 runs.
2) Each value in the above table represents the CPU time which corresponds to the case where the output on screen is not displayed. That is, we comment out Lines 18 and 19 for Program I and Lines 21 and 22 for Program II.
3) The above table is obtained using Pentium III 1GHz Dual CPU personal computer, Microsoft Windows 2000 (SP2) operating system and Open WATCOM C/C++32 Optimizing Compiler (Version 1.0).

In Program II, we use the recursion in which the function `nest0()` is called from the same function `nest0()`. In Line 7, the initial value 1 is given to `column`. Whenever the function `nest0()` is called, `column` increases by one. The function `nest0()` is called $n1$ times until `n1` is equal to `column`.

Comparison of CPU Time: Now we have two source codes. Here, they are compared with respect to computational time. Adding the appropriate main programs to Programs I and II, the results are obtained in Table 7.7.

From Table 7.7, we can see that computational time extraordinarily increases as either $n1$ or $n2$ increases. In the case of $n1 = n2 = n$ (i.e., diagonal elements in Table 7.7), the computational difference between ${}_{2n}C_n$ and ${}_{2n-2}C_{n-1}$ is given by ${}_{2n}C_n / {}_{2n-2}C_{n-1} = (4n-2)/n$, which implies that computational cost becomes three times for $n = 2$ and four times for sufficiently large n.

Clearly, Program II might be much easier to understand than Program I at a glance. Moreover, according to comparison of computational time, Program II is more efficient than Program I. Judging from computational time and simplicity of computer program, Program II is more practical than Program I. Therefore, we utilize Program II in Chapters 7 and 8.

Summary: Harbison and Steele (1987) have shown the computer program which obtains all the possible combinations, using the bit operation. In this appendix, alternatively we have introduced another source code, utilizing the recursion. Moreover, the two source codes have been compared with respect to computational time. As a

result, the source code with the recursion is simpler and 3 – 5 times less computational than that with the bit operation. Therefore, we can see that the former is practically more useful than the latter.

Appendix 7.2: Equivalence between Fisher and *t* Tests

In this appendix, we show the asymptotic equivalence between Fisher and t Tests. Define $U = \overline{x} - \overline{y}$ and $U_x = \sum_{i=1}^{n1} x_i$. Moreover, define $\{z_i\}_{i=1}^{n1+n2}$ such that $z_i = x_i$ for $i = 1, 2, \cdots, n1$ and $z_{n1+j} = y_j$ for $j = 1, 2, \cdots, n$. That is, we have the following:

$$\overline{z} = \frac{1}{n1+n2}\sum_{i=1}^{n1+n2} z_i = \frac{1}{n1+n2}(n1\,\overline{x} + n2\,\overline{y}).$$

The relationship between U and U_x is given by $U = \overline{x} - \overline{y} = (1/n1 + 1/n2)\sum_{i=1}^{n1} x_i - \overline{z}(1 + n1/n2) = (1/n1 + 1/n2)U_x - (1 + n1/n2)\overline{z}$. Therefore, the conditional mean and variance of U_x given $\{z_i\}_{i=1}^{n1+n2}$ are given by:

$$\begin{aligned}
&\mathrm{E}(U_x|z_1, z_2, \cdots, z_{n1+n2}) = n1\,\overline{z},\\
&\mathrm{V}(U_x|z_1, z_2, \cdots, z_{n1+n2}) = \frac{n1\,n2}{n1+n2-1}\frac{1}{n1+n2}\sum_{i=1}^{n1+n2}(z_i - \overline{z})^2,
\end{aligned}$$

which can be obtained from the results of sampling without replacement in the finite population. The similar discussion is in Section 7.3.1, where $\mathrm{E}(s)$ and $\mathrm{V}(s)$ are derived in the same way.

From $U = (1/n1 + 1/n2)U_x - (1 + n1/n2)\overline{z}$, the conditional mean and variance of U given $\{x_i\}_{i=1}^{n1}$ and $\{y_j\}_{j=1}^{n2}$ are given by:

$$\begin{aligned}
&\mathrm{E}(U|z_1, z_2, \cdots, z_{n1+n2}) = 0,\\
&\mathrm{V}(U|z_1, z_2, \cdots, z_{n1+n2}) = \frac{n1+n2}{n1\,n2\,(n1+n2-1)}\sum_{i=1}^{n1+n2}(z_i - \overline{z})^2.
\end{aligned}$$

$\sum_{i=1}^{n1+n2}(z_i - \overline{z})^2$ is rewritten as follows:

$$\begin{aligned}
\sum_{i=1}^{n1+n2}(z_i - \overline{z})^2 &= \sum_{i=1}^{n1}(x_i - \overline{x})^2 \sum_{j=1}^{n2}(y_j - \overline{y})^2 + \frac{n1\,n2}{n1+n2}(\overline{x} - \overline{y})^2\\
&= (n1+n2-2)s^2 + \frac{n1\,n2}{n1+n2}U^2,
\end{aligned}$$

where

$$s^2 = \frac{1}{n1+n2-2}\Big(\sum_{i=1}^{n1}(x_i - \overline{x})^2 + \sum_{j=1}^{n2}(y_j - \overline{y})^2\Big).$$

For large $n1$ and $n2$, when we have $\mathrm{E}(|z|^3) < \infty$ under the null hypothesis $H_0 : \theta = 0$, it is shown that the conditional distribution of U given $\{z_i\}_{i=1}^{n1+n2}$ is approximated as a normal distribution (also, see Lehmann (1986)). This equivalence is not limited to the behavior under the hypothesis. That is, for large samples, it is shown by Hoeffding (1952) and Bickel and van Zwet (1978) that the power of the randomization test is approximately equal to that of the t test. Therefore, we can obtain the asymptotic distribution as follows:

$$\frac{U}{\sqrt{\dfrac{n1+n2}{n1\ n2\ (n1+n2-1)}\displaystyle\sum_{i=1}^{n1+n2}(z_i-\overline{z})^2}} = \frac{U}{\sqrt{\dfrac{(n1+n2)(n1+n2-2)}{n1\ n2\ (n1+n2-1)}s^2+\dfrac{1}{n1+n2-1}U^2}} \longrightarrow N(0,1).$$

Let z_α be the $100 \times \alpha$ percent point of the standard normal distribution. Then, when we have the following:

$$\frac{U}{\sqrt{\dfrac{(n1+n2)(n1+n2-2)}{n1\ n2\ (n1+n2-1)}s^2+\dfrac{1}{n1+n2-1}U^2}} > z_\alpha, \tag{7.21}$$

the null hypothesis $H_0 : \theta = 0$ against the alternative one $H_1 : \theta > 0$ is rejected.

Using U, the t statistic on the two-sample mean problem is given by:

$$T = \frac{U}{\sqrt{\left(\dfrac{1}{n1}+\dfrac{1}{n2}\right)s^2}}. \tag{7.22}$$

Substituting (7.22) into (7.21), U is eliminated and it is represented by T, which is rewritten as follows:

$$T > z_\alpha \sqrt{\frac{n1+n2-2}{n1+n2-1-z_\alpha^2}} \longrightarrow z_\alpha.$$

Thus, it follows in large sample that the randomization test can be approximated by the t test. In large sample, note that both randomization and t tests are normally distributed and that they have the same percent points.

Appendix 7.3: Random Combination

`random_combination(n1,n2,num)` represents the function which chooses `num` out of ${}_{n1+n2}C_{n1}$ combinations randomly with equal probability. Therefore, it is possible that some of the `num` combinations are exactly same. The source code which chooses one of the combinations randomly is shown as follows.

random_combination(n1,n2,num)

```
1: void random_combination(int n1,int n2,int num)
2: {
3:     int    i,j,m,temp,index[1001];
4:     float  rn,urnd();
5:
6:     for(i=1;i<=n1+n2;i++) index[i]=i;
7:     for(m=1;m<=num;m++){
8:        for(i=1;i<=n1;i++) {
9:           rn=urnd();
10:          j=i+(int)(rn*(n1+n2-i+1));
11:          temp=index[j]; index[j]=index[i]; index[i]=temp;
12:       }
13:       printf(" %4d ",m);
14:       for(i=1;i<=n1;i++) printf(" %2d",index[i]);
15:       printf("\n");
16:    }
17: }
```

In the function random_combination(n1,n2,num), variables n1, n2 and num should be input. num combinations are output and printed on screen in Line 14. Initially, the integers up to n1+n2 are assigned to index[i] in Line 6. num combinations are randomly generated between Lines 7 and 16. In Lines 8 – 12, one of ${}_{n1+n2}C_{n1}$ combinations is generated, using the uniform random draw in Line 9.

Thus, utilizing the function random_combination(n1,n2,num), the cases of $M = 10^4,\ 10^5,\ 10^6$ in Table 7.3 are computed, and all the cases where the combinations exceed 10^7 in Tables 7.4 and 7.5 (i.e., 1964 – 1993 in Table 7.4 and 1966 – 1993 in Table 7.5) are also obtained.

Appendix 7.4: Testing Structural Change

In Tables 7.4 and 7.5, $F(3, 40)$ and $F(4, 37)$ are used for testing the structural change of the regression coefficients. In this appendix, the F statistic is derived in a general formulation.

Consider the following regression model:

$$y_t = d_t^- x_t\beta^- + d_t^+ x_t\beta^+ + \epsilon_t, \qquad \epsilon_t \sim N(0, \sigma^2),$$

where d_t and d_t^+ are defined by:

$$d_t^- = \begin{cases} 1, & \text{for } t = 1, 2, \cdots, n1, \\ 0, & \text{for } t = n1+1, n1+2, \cdots, N, \end{cases}$$

$$d_t^+ = \begin{cases} 0, & \text{for } t = 1, 2, \cdots, n1, \\ 1, & \text{for } t = n1+1, n1+2, \cdots, N. \end{cases}$$

In a matrix form, we can rewrite the above regression model as follows:

$$Y = (\, D_{n1}^{-} X \quad D_{n1+1}^{+} X \,)\beta^{*} + u,$$

where D_{n1}^{-}, D_{n1+1}^{+} and β^{*} are represented as:

$$D_{n1}^{-} = \begin{pmatrix} d_1^{-} & 0 & \cdots & 0 \\ 0 & d_2^{-} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & d_N^{-} \end{pmatrix} = \begin{pmatrix} I_{n1} & 0 \\ 0 & 0 \end{pmatrix}, \qquad \beta^{*} = \begin{pmatrix} \beta^{-} \\ \beta^{+} \end{pmatrix},$$

$$D_{n1+1}^{+} = \begin{pmatrix} d_1^{+} & 0 & \cdots & 0 \\ 0 & d_2^{+} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & d_N^{+} \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & I_{N-n1} \end{pmatrix}.$$

Both D_{n1}^{-} and D_{n1+1}^{+} are $N \times N$ matrices. Note that I_k denotes a $k \times k$ identity matrix.

Under the above model, we consider the null hypothesis $H_0 : R\beta^{*} = r$ and the alternative one $H_1 : R\beta^{*} \neq r$, where R and r are nonstochastic. Because $R\hat{\beta}^{*}$ has the following normal distribution:

$$R\hat{\beta}^{*} \sim N\left(r, \sigma^2 R \begin{pmatrix} (X' D_{n1}^{-} X)^{-1} & 0 \\ 0 & (X' D_{n1+1}^{+} X)^{-1} \end{pmatrix} R'\right),$$

the quadratic form is distributed as the following chi-squared distribution:

$$(R\hat{\beta}^{*} - r)' \left(\sigma^2 R \begin{pmatrix} (X' D_{n1}^{-} X)^{-1} & 0 \\ 0 & (X' D_{n1+1}^{+} X)^{-1} \end{pmatrix} R'\right)^{-1} (R\hat{\beta}^{*} - r) \sim \chi^2(q),$$

where q denotes the rank of R. Moreover, we have the following distribution:

$$\frac{\hat{u}'\hat{u}}{\sigma^2} \sim \chi^2(N - 2k),$$

where $\hat{u}$ is defined as the residual: $\hat{u} = Y - (\, D_{n1}^{-} X \quad D_{n1+1}^{+} X \,)\hat{\beta}^{*}$. Therefore, the ratio of the above two chi-squared random variables divided by the corresponding degrees of freedom is distributed as:

$$\frac{(R\hat{\beta}^{*} - r)' \left(R \begin{pmatrix} (X' D_{n1}^{-} X)^{-1} & 0 \\ 0 & (X' D_{n1+1}^{+} X)^{-1} \end{pmatrix} R'\right)^{-1} (R\hat{\beta}^{*} - r) \,\big/\, q}{\hat{u}'\hat{u}/(N - 2k)} \sim F(q, N - 2k).$$

Note that in the above random variable the numerator is independent of the denominator.

Now, taking $R = (I_k, -I_k)$ and $r = 0$, we have the following F distribution:

$$F \equiv \frac{(\hat{\beta}^{-} - \hat{\beta}^{+})' ((X' D_{n1}^{-} X)^{-1} + (X' D_{n1+1}^{+} X)^{-1})^{-1} (\hat{\beta}^{-} - \hat{\beta}^{+}) \,/\, k}{\hat{u}'\hat{u}/(N - 2k)} \sim F(k, N - 2k).$$

Note that the rank of R is k in this case. The null hypothesis $R\beta^* = r$ reduces to: $\beta^- = \beta^+$. Thus, F is used for testing if β^- is equal to β^+, where β^- denotes the regression coefficient in the first $n1$ periods, while β^+ represents that in the last $N - n1$ periods.

Thus, the F distribution with k and $N - 2k$ degrees of freedom, $F(k, N - 2k)$, is obtained. In the last row of Tables 7.4 and 7.5, the probabilities corresponding to the $F(k, N - 2k)$ test statistics are computed, using the IMSL library (`http://www.vni.com/products/imsl`) with Microsoft Fortran PowerStation Version 4.00.

References

Bickel, P.J. and van Zwet, W.R., 1978, "Asymptotic Expansions for the Power of Distribution Free Tests in the Two-Sample Problem," *Annals of Statistics*, Vol.6, No.5, pp.937 – 1004.

Bradley, J.V., 1968, *Distribution-Free Statistical Tests*, Englewood Cliffs, New Jersey: Prentice-Hall.

Chernoff, H. and Savage, I.R., 1958, "Asymptotic Normality and Efficiency of Certain Nonparametric test Statistics," *Annals of Mathematical Statistics*, Vol.29, No.4, pp.972 – 994.

Fisher, R.A., 1935, *The Design of Experiments* (eighth edition, 1966), New York: Hafner.

Harbison, S.P. and Steele Jr., G.L., 1987, *C: A Reference Manual* (second edition), Prentice-Hall.

Hodges, J.L. and Lehmann, E.L., 1956, "The Efficiency of Some Nonparametric Competitors of the t Test," *Annals of Mathematical Statistics*, Vol.27, No.2, pp.324 – 335.

Hoeffding, W., 1952, "The Large Sample Power of Tests Based on Permutations of Observations," *Annals of Mathematical Statistics*, Vol.23, No.2, pp.169 – 192.

Kendall, M. and Stuart, A., 1979, *The Advanced Theory of Statistics, Vol.2, Inference and Relationship* (fourth edition), Charles Griffin & Company Limited.

Lehmann, E.L., 1986, *Testing Statistical Hypotheses* (Second Edition), John Wiley & Sons.

Mann, H.B. and Whitney, D.R., 1947, "On a Test of Whether One of Two Random Variables Is Stochastically Larger Than the Other," *Annals of Mathematical Statistics*, Vol.18, No.1, pp.50 – 60.

Mehta, C.R. and Patel, N.R., 1983, "A Network Algorithm for Performing Fisher's Exact Test in $r \times c$ Contingency Tables," *Journal of the American Statistical Association*, Vol.78, No.382, pp.427 – 434.

Mehta, C.R., Patel, N.R. and Tsiatis, A.A., 1984, "Exact Significance Testing to Establish Treatment Equivalence for Ordered Categorical Data," *Biometrics*, Vol.40, pp.819 – 825.

Mehta, C.R., Patel, N.R. and Gray, R., 1985, "On Computing an Exact Confidence Interval for the Common Odds Ratio in Several 2×2 Contingency Tables," *Journal of the American Statistical Association*, Vol.80, No.392, pp.969 – 973.

Mehta, C.R. and Patel, N.R., 1986a, "A Hybrid Algorithm for Fisher's Exact Test in Unordered $r \times c$ Contingency Tables," *Communications in Statistics*, Vol.15, No.2, pp.387 – 403.

Mehta, C.R. and Patel, N.R., 1986b "FEXACT: A Fortran Subroutine for Fisher's Exact Test on Unordered $r \times c$ Contingency Tables," *ACM Transactions on Mathematical Software*, Vol.12, No.2, pp.154 – 161.

Mehta, C.R., Patel, N.R. and Wei, L.J., 1988, "Computing Exact Permutational Distributions with Restricted Randomization Designs," *Biometrica*, Vol.75, No.2, pp.295 – 302.

Mood, A.M., Graybill, F.A. and Boes, D.C., 1974, *Introduction to the Theory of Statistics* (third edition), McGraw-Hill.

Mehta, C.R. and Patel, N.R., 1992, *StatXact: User Manual*, CYTEL Software Corporation.

Tanizaki, H., 1997, "Power Comparison of Nonparametric Tests: Small Sample Properties from Monte-Carlo Experiments," *Journal of Applied Statistics*, Vol.24, No.5, pp.603 – 632.

Wilcoxon, F., 1945, "Individual Comparisons by Ranking Methods," *Biometrics*, Vol.1, pp.80 – 83.

Chapter 8

Independence between Two Samples

In Chapter 7, we have discussed the testing procedure on difference between two sample means, where the nonparametric tests such as the score tests and the Fisher permutation test are utilized. In this chapter, we consider some nonparametric tests (i.e., score tests and **Fisher's permutation test**) on the correlation coefficient, which is applied to a significance test on regression coefficients. Because the nonparametric tests are very computer-intensive, there are few studies on small-sample properties, although we have numerous studies on asymptotic properties with regard to various aspects. In this chapter, we aim to compare the nonparametric tests with the t test through Monte Carlo experiments, where an **independence test** between two samples and a **significance test** for regression models are taken. For both the independence and significance tests, we obtain the results through Monte Carlo experiments that the nonparametric tests perform better than the t test when the underlying sample is not Gaussian and that the nonparametric tests are as good as the t test even under the Gaussian population.

8.1 Introduction

In regression models, we assume that disturbance terms are mutually independently and identically distributed. In addition, in the case where we perform the significance test on the regression coefficients, we assume that the error terms are normally distributed. Under these assumptions, it is known that the ordinary least squares (OLS) estimator of the regression coefficients follows the t distribution with $n - k$ degrees of freedom, where n and k denote the sample size and the number of the regression coefficients.

As the sample size n increases, the t distribution approaches the standard normal distribution $N(0, 1)$. From the central limit theorem, it is known that the OLS estimator of the regression coefficient is normally distributed for a sufficiently large sample size if variance of the OLS estimator is finite. However, in the case where the error term is non-Gaussian and the sample size is small, the OLS estimator does not have the t

distribution and therefore we cannot apply the t test. To improve these problems, in this chapter we consider a significance test of the regression coefficient even in the case where the error term is non-Gaussian and the sample size is small, where some nonparametric tests (or a distribution-free test) are applied.

Generally we can regard the OLS estimator of the regression coefficient as the correlation between two samples. The nonparametric tests based on **Spearman's rank correlation** coefficient and **Kendall's rank correlation** coefficient are very famous. See, for example, Hollander and Wolfe (1973), Randles and Wolfe (1979), Conover (1980), Sprent (1989), Gibbons and Chakraborti (1992) and Hogg and Craig (1995) for the **rank correlation test**s. In this chapter, the **score test**s as well as the **permutation test** proposed by Fisher (1966) are utilized, where we compute the correlation coefficient for each of all the possible permutations and all the possible correlation coefficients are compared with the correlation coefficient based on the original data. The score tests and the permutation test can be directly applied to the regression problem.

The outline of this chapter is as follows. In Section 8.2, we introduce the nonparametric tests based on the score test and the permutation test, where we consider testing whether X is correlated with Y for the sample size n. Moreover, we show that we can directly apply the correlation test to the regression problem without any modification. In Section 8.3, we compare the powers of the nonparametric tests and the conventional t test when the underlying data are non-Gaussian. In the case where $k = 2, 3$ is taken for the number of regression coefficients, we examine whether the empirical sizes are correctly estimated when the significance level is $\alpha = 0.10, 0.05, 0.01$. In Section 8.4, an empirical example is taken, where the empirical distribution of the regression coefficient is drawn and the confidence interval is nonparametrically constructed.

8.2 Nonparametric Tests on Independence

8.2.1 On Testing the Correlation Coefficient

Let (X_1, Y_1), (X_2, Y_2), $\cdots$, (X_n, Y_n) be a random sample, where the sample size is n. Consider testing if there is a correlation between X and Y, i.e., if the correlation coefficient ρ is zero or not. The correlation coefficient ρ is defined as:

$$\rho = \frac{\mathrm{Cov}(X, Y)}{\sqrt{\mathrm{V}(X)\mathrm{V}(Y)}},$$

where $\mathrm{Cov}(X, Y)$, $\mathrm{V}(X)$ and $\mathrm{V}(Y)$ represent the covariance between X and Y, the variance of X and the variance of Y, respectively. Then, the sample correlation coefficient $\hat{\rho}$ is written as:

$$\hat{\rho} = \frac{S_{XY}}{S_X S_Y},$$

where S_{XY}, S_X and S_Y denote the sample covariance between X and Y, the sample variance of X and the sample variance of Y, which are given by:

$$S_{XY} = \frac{1}{n}\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y}), \quad S_X^2 = \frac{1}{n}\sum_{i=1}^{n}(X_i - \overline{X})^2, \quad S_Y^2 = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \overline{Y})^2.$$

$\overline{X}$ and $\overline{Y}$ represent the sample means of X and Y, i.e., $\overline{X} = (1/n)\sum_{i=1}^{n} X_i$ and $\overline{Y} = (1/n)\sum_{i=1}^{n} Y_i$.

If X is independent of Y, we have $\rho = 0$ and the joint density of X and Y is represented as a product of the two marginal densities of X and Y, i.e.,

$$f_{xy}(x, y) = f_x(x) f_y(y),$$

where $f_{xy}(x, y)$, $f_x(x)$ and $f_y(y)$ denote the joint density of X and Y, the marginal density of X and the marginal density of Y. The equation above implies that for all i and j we consider randomly taking n pairs of X_i and Y_j. Accordingly, given X_1, the possible permutations are taken as (X_1, Y_j), $j = 1, 2, \cdots, n$, where we have n permutations. Therefore, the number of all the possible permutations between X and Y are given by $n!$. For each permutation, we can compute the correlation coefficient. Thus, $n!$ correlation coefficients are obtained. The n correlation coefficients are compared with the correlation coefficient obtained from the original pairs of data. If the correlation coefficient obtained from the original data is in the tail of the empirical distribution constructed from the $n!$ correlation coefficients, the hypothesis that X is correlated with Y is rejected. The testing procedure above is distribution-free or nonparametric, which can be applicable to almost all the cases. The nonparametric test discussed above is known as a **permutation test**, which is developed by Fisher (1966). For example, see Stuart and Ord (1991).

The order of X_i, $i = 1, 2, \cdots, n$, is fixed as it is and we permute Y_j, $j = 1, 2, \cdots, n$, randomly. Based on the $n!$ correlation coefficients, we can test if X is correlated with Y. Let the $n!$ correlation coefficients be $\hat{\rho}^{(m)}$, $m = 1, 2, \cdots, n!$. If Y is independent of X, each correlation coefficient occurs with equal probability, i.e., $1/n!$. Suppose that $\hat{\rho}^{(0)}$ is the correlation coefficient obtained from the original data. The estimator of the correlation coefficient ρ, denoted by $\hat{\rho}$, is distributed as:

$$P(\hat{\rho} < \hat{\rho}^{(0)}) = \frac{\text{the number of } \hat{\rho}^{(m)} \text{ which satisfies } \hat{\rho}^{(m)} < \hat{\rho}^{(0)},\ m = 1, 2, \cdots, n!}{n!},$$

$$P(\hat{\rho} = \hat{\rho}^{(0)}) = \frac{\text{the number of } \hat{\rho}^{(m)} \text{ which satisfies } \hat{\rho}^{(m)} = \hat{\rho}^{(0)},\ m = 1, 2, \cdots, n!}{n!},$$

$$P(\hat{\rho} > \hat{\rho}^{(0)}) = \frac{\text{the number of } \hat{\rho}^{(m)} \text{ which satisfies } \hat{\rho}^{(m)} > \hat{\rho}^{(0)},\ m = 1, 2, \cdots, n!}{n!}.$$

Note that at least one of the $n!$ permutations (i.e., $\hat{\rho}^{(1)}, \hat{\rho}^{(2)}, \cdots, \hat{\rho}^{(n!)}$) is exactly equal to $\hat{\rho}^{(0)}$. See Appendix 8.1 for the source code which obtains all the permutations. Thus, the above three probabilities can be computed. The null hypothesis $H_0 : \rho = 0$ is rejected by the one-sided test if $P(\hat{\rho} < \hat{\rho}^{(0)})$ or $P(\hat{\rho} > \hat{\rho}^{(0)})$ is small enough. This test is denoted by f in Monte Carlo experiments of Section 8.3.1.

Remark 1: The sample covariance between X and Y, S_{XY}, is rewritten as:

$$S_{XY} = \frac{1}{n}\sum_{i=1}^{n} X_i Y_i - \overline{XY}.$$

The sample means $\overline{X}$ and $\overline{Y}$ take the same values without depending on the order of X and Y. Similarly, S_X^2 and S_Y^2 are also independent of the order of X and Y. Therefore, $\hat{\rho}$ depends only on $\sum_{i=1}^{n} X_i Y_i$. That is, for the empirical distribution based on the $n!$ correlation coefficients, $\hat{\rho}$ is a monotone function of $\sum_{i=1}^{n} X_i Y_i$, which implies that we have one-to-one correspondence between $\hat{\rho}$ and $\sum_{i=1}^{n} X_i Y_i$. To test no correlation between X and Y, we need to compute $\sum_{i=1}^{n} X_i Y_i$ but we do not have to compute $\hat{\rho}$. Thus, by utilizing $\sum_{i=1}^{n} X_i Y_i$ rather than $\hat{\rho}$, computational burden can be reduced.

Remark 2: As for a special case, suppose that (X_i, Y_i), $i = 1, 2, \cdots, n$, are normally distributed, i.e.,

$$\begin{pmatrix} X_i \\ Y_i \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_X \\ \mu_Y \end{pmatrix}, \begin{pmatrix} \sigma_X^2 & \rho\sigma_X\sigma_Y \\ \rho\sigma_X\sigma_Y & \sigma_Y^2 \end{pmatrix}\right).$$

Define $T = \hat{\rho}\sqrt{n-2}/\sqrt{1-\hat{\rho}^2}$, which is based on the sample correlation coefficient $\hat{\rho}$, Under the null hypothesis $H_0 : \rho = 0$, the statistic T is distributed as the following t distribution:

$$T = \frac{\hat{\rho}\sqrt{n-2}}{\sqrt{1-\hat{\rho}^2}} \sim t(n-2),$$

which derivation is discussed in Appendix 8.2. This test is denoted by t in Monte Carlo experiments of Section 8.3.1. Note that we cannot use a t distribution in the case of testing the null hypothesis $H_0 : \rho = \rho_0$. For example, see Lehmann (1986), Stuart and Ord (1991, 1994) and Hogg and Craig (1995). Generally, it is natural to consider that (X, Y) is non-Gaussian and that the distribution of (X, Y) is not known. If the underlying distribution is not Gaussian and the t distribution is applied to the null hypothesis $H_0 : \rho = 0$, the appropriate testing results cannot be obtained. However, the nonparametric permutation test can be applied even in the non-Gaussian cases, because it is distribution-free.

Under normal population and the assumption of $\rho = 0$, T has the t distribution with $n-2$ degrees of freedom. However, as a generalization, it is shown that the distribution of T under $-1 < \rho < 1$ is quite complicated, i.e., the conditional density of T given $\psi = (\rho/\sqrt{1-\rho^2})\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2}/\sigma_X$ follows a noncentral t distribution with $n-2$ degrees of freedom and noncentrality parameter $\psi = (\rho/\sqrt{1-\rho^2})\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2}/\sigma_X$, where x_i denotes a realization of X_i. The conditional density of T given ψ, i.e., the noncentral t distribution, is also derived in Appendix 8.2.

Remark 3: Let Rx_i be the ranked data of X_i, i.e., Rx_i takes one of 1, 2, $\cdots$, n. Similarly, Ry_j denotes the ranked data of Y_j, i.e., Ry_j also takes one of 1, 2, $\cdots$, n.

Define the test statistic s_0 as follows:

$$s_0 = \frac{\sum_{i=1}^{n}(a(Rx_i) - \overline{a(Rx)})(a(Ry_i) - \overline{a(Ry)})}{\sqrt{\sum_{i=1}^{n}(a(Rx_i) - \overline{a(Rx)})^2}\sqrt{\sum_{i=1}^{n}(a(Ry_i) - \overline{a(Ry)})^2}}, \tag{8.1}$$

where $a(\cdot)$ represents a function to be specified. $\overline{a(Rx)}$ and $\overline{a(Ry)}$ are given by $\overline{a(Rx)} = (1/n)\sum_{i=1}^{n} a(Rx_i)$ and $\overline{a(Ry)} = (1/n)\sum_{i=1}^{n} a(Ry_i)$. The above statistic (8.1) is equivalent to the product of $a(Rx_i)$ and $a(Ry_i)$, i.e., $\sum_{i=1}^{n} a(Rx_i)a(Ry_i)$, in the sense of the test statistic. When $a(z) = z$ is specified, s_0 is called the rank correlation coefficient, where z takes Rx_i or Ry_i.

Representatively, $a(\cdot)$ is specified as follows:

$$a(Rx_i) = c\Big(\frac{Rx_i - 0.5}{n}\Big) \equiv c(px_i), \quad a(Ry_i) = c\Big(\frac{Ry_i - 0.5}{n}\Big) \equiv c(py_i),$$

where px_i and py_i are defined as $px_i \equiv (Rx_i - 0.5)/n$ and $py_i \equiv (Ry_i - 0.5)/n$, and $c(\cdot)$ is an inverse of a distribution function. Thus, we can consider the correlation coefficient based on the **score function** $c(\cdot)$, called the **score correlation coefficient**. $c(\cdot)$ may be specified as the inverse of the standard normal distribution function, the logistic distribution function, the uniform distribution function and so on. Especially, when $c(\cdot)$ is specified as the inverse of the uniform distribution function between zero and one (i.e., when $c(x) = x$ for $0 < x < 1$, $c(x) = 0$ for $x \leq 0$ and $c(x) = 1$ for $x \geq 1$), the s_0 in the case where $a(Rx_i)$ and $a(Ry_i)$ are replaced by $c(px_i)$ and $c(py_i)$ is equivalent to the **rank correlation coefficient**. In Section 8.3.1, the inverse of the uniform distribution between zero and one (i.e., *w*), that of the standard normal distribution function (i.e., *ns*), that of the logistic distribution function (i.e., *ls*) and that of the Cauchy distribution function (i.e., *cs*) are utilized for $c(\cdot)$.

The t in the case where X_i and Y_i are replaced by Rx_i and Ry_i is asymptotically distributed as the standard normal distribution. This test is denoted by *aw* in Monte Carlo experiments of Section 8.3.1, where the empirical sizes and sample powers are obtained using the t distribution with $n - 2$ degrees of freedom, even though an asymptotic distribution of *aw* is the standard normal distribution.

Remark 4: In the case where we perform the permutation test on the correlation coefficient, we need to compute the $n!$ correlation coefficients (for example, $n!$ is equal to about 3.6 million when $n = 10$). The case of sample size n is n times more computer-intensive than that of sample size $n-1$. Thus, the permutation test discussed in this chapter is very computer-intensive.

Therefore, we need to consider less computational procedure. In order to reduce computational burden when $n!$ is large, it might be practical to choose some of the $n!$ permutations randomly and perform the same testing procedure. We can consider that all the permutations occur with equal probability, i.e., $1/n!$. Therefore, we pick up M out of the $n!$ permutations randomly and compute the probabilities $P(\hat{\rho} < \hat{\rho}^{(0)})$, $P(\hat{\rho} =$

$\hat{\rho}^{(0)}$) and $P(\hat{\rho} > \hat{\rho}^{(0)})$. For example, if either of $P(\hat{\rho} > \hat{\rho}^{(0)})$ or $P(\hat{\rho} < \hat{\rho}^{(0)})$ is smaller than $\alpha = 0.1$, the null hypothesis $H_0 : \rho = 0$ is rejected with 10% significance level. We examine whether the empirical size depends on M in Section 8.3. Thus, we can choose some out of the $n!$ permutations and compute the corresponding probabilities.

8.2.2 On Testing the Regression Coefficient

Using the exactly same approach as the nonparametric test on the correlation coefficient, discussed in Section 8.2.1, we consider the nonparametric significance test on the regression coefficients.

The regression model is given by:

$$Y_i = X_i\beta + \epsilon_i, \qquad i = 1, 2, \cdots, n,$$

where the OLS estimator of β, i.e., $\hat{\beta}$, is represented as:

$$\hat{\beta} = (X'X)^{-1}X'Y = \sum_{i=1}^{n}(X'X)^{-1}X_i'Y_i. \tag{8.2}$$

The notations are as follows:

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix}, \qquad X = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{pmatrix}, \qquad \beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{pmatrix}, \qquad \hat{\beta} = \begin{pmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \vdots \\ \hat{\beta}_k \end{pmatrix},$$

where Y_i denotes the ith element of a $n \times 1$ vector Y and X_i indicates the ith row vector of a $n \times k$ matrix X.

From the structure of equation (8.2), When X_i in Section 8.2.1 is replaced by $(X'X)^{-1}X_i'$, we can see that the same discussion as in Section 8.2.1 holds. As for only one difference, X_i is a scalar in Section 8.2.1 while $(X'X)^{-1}X_i'$ is a $k \times 1$ vector in this section. Therefore, we have $n!$ regression coefficients by changing the order of Y. This implies that the correlation between any two explanatory variables is preserved. Let $\hat{\beta}^{(m)}$, $m = 1, 2, \cdots, n!$, be the $n!$ regression coefficients and $\hat{\beta}_j^{(m)}$ be the jth element of $\hat{\beta}^{(m)}$, i.e., $\hat{\beta}^{(m)} = (\hat{\beta}_1^{(m)}, \hat{\beta}_2^{(m)}, \cdots, \hat{\beta}_k^{(m)})'$. Suppose that $\hat{\beta}_j^{(0)}$ represents the jth element of the regression coefficient vector obtained from the original data series. Under the null hypothesis $H_0 : \beta_j = 0$, the empirical distribution of $\hat{\beta}_j$, which is the jth element of the OLS estimator of β, is given by:

$$P(\hat{\beta}_j < \hat{\beta}_j^{(0)}) = \frac{\text{the number of } \hat{\beta}_j^{(m)} \text{ which satisfies } \hat{\beta}_j^{(m)} < \hat{\beta}_j^{(0)},\ m = 1, 2, \cdots, n!}{n!},$$

$$P(\hat{\beta}_j = \hat{\beta}_j^{(0)}) = \frac{\text{the number of } \hat{\beta}_j^{(m)} \text{ which satisfies } \hat{\beta}_j^{(m)} = \hat{\beta}_j^{(0)},\ m = 1, 2, \cdots, n!}{n!},$$

$$P(\hat{\beta}_j > \hat{\beta}_j^{(0)}) = \frac{\text{the number of } \hat{\beta}_j^{(m)} \text{ which satisfies } \hat{\beta}_j^{(m)} > \hat{\beta}_j^{(0)},\ m = 1, 2, \cdots, n!}{n!}.$$

For all $j = 1, 2, \cdots, k$, we can implement the same computational procedure as above and compute each probability. We can perform the significance test by examining where $\hat{\beta}_j^{(0)}$ is located among the $n!$ regression coefficients. The null hypothesis H_0 : $\beta_j = 0$ is rejected by the one-sided test if $P(\hat{\beta}_j < \hat{\beta}_j^{(0)})$ or $P(\hat{\beta}_j > \hat{\beta}_j^{(0)})$ is small enough. This nonparametric permutation test is denoted by f in Monte Carlo studies of Section 8.3.2.

Remark 5: Generally, as for the testing procedure of the null hypothesis $H_0 : \beta = \beta^*$, we may consider the nonparametric permutation test between $(X'X)^{-1}X_i'$ and $Y_i - X_i\beta^*$, because $\hat{\beta} - \beta$ is transformed into:

$$\hat{\beta} - \beta = (X'X)^{-1}X'Y - \beta = (X'X)^{-1}X'(Y - X\beta) = \sum_{i=1}^{n}(X'X)^{-1}X_i'(Y_i - X_i\beta),$$

which implies that $\hat{\beta}-\beta$ is equivalent to the correlation between $(X'X)^{-1}X_i'$ and $Y_i-X_i\beta$.

Remark 6: As for the conventional parametric significance test, the error terms ϵ_i, $i = 1, 2, \cdots, n$, are assumed to be mutually independently and normally distributed with mean zero and variance σ^2. Under the null hypothesis $H_0 : \beta_j = \beta_j^*$, the jth element of the OLS estimator (i.e., $\hat{\beta}_j$) is distributed as:

$$\frac{\hat{\beta}_j - \beta_j^*}{S\sqrt{a_{jj}}} \sim t(n-k),$$

where a_{jj} denotes the jth diagonal element of $(X'X)^{-1}$. β^* and S^2 represent $\beta^* = (\beta_1^*, \beta_2^*, \cdots, \beta_k^*)'$ and $S^2 = (Y - X\hat{\beta})'(Y - X\hat{\beta})/(n-k)$, respectively. Thus, only when ϵ_i is assumed to be normal, we can use the t distribution. However, unless ϵ_i is normal, the conventional t test gives us the incorrect inference in the small sample. As well known, in the large sample, using the central limit theorem $\sqrt{n}(\hat{\beta}_j - \beta_j)$ is asymptotically normal when the variance of ϵ_i is finite. Thus, the case of the large sample is different from that of the small sample. In this chapter, under the non-Gaussian assumption, we examine the powers of the nonparametric tests on the correlation coefficient through Monte Carlo experiments. Moreover, in the regression analysis we examine how robust the conventional t test is, when the underlying population is not Gaussian. The above test is denoted by t in Monte Carlo studies of Section 8.3.1.

Remark 7: As in Section 8.2.1, consider replacing $X_{i,j}$ and Y_i by $c(px_{i,j})$ and $c(py_i)$, where $X_{i,j}$ represents the jth element of X_i, i.e., $X_i = \{X_{i,1}, X_{i,2}, \cdots, X_{i,k}\}$. $px_{i,j}$ and py_i are defined as $px_{i,j} = (Rx_{i,j}-0.5)/n$ and $py_i = (Ry_i-0.5)/n$, where $Rx_{i,j}$ and Ry_i denote the ranked data of $X_{i,j}$ and Y_i, respectively. That is, $X_{1,j}, X_{2,j}, \cdots, X_{n,j}$ are ranked by size and accordingly $Rx_{i,j}$ takes one of the integers from 1 to n for $i = 1, 2, \cdots, n$.

Based on the **score function** $c(\cdot)$, we can examine if the regression coefficient is zero. In this case, the regression model is written as follows:

$$c(py_i) = \beta_1 c(px_{i,1}) + \beta_2 c(px_{i,2}) + \cdots + \beta_k c(px_{i,k}) + \epsilon_i.$$

Under the above regression model, we can consider the significance test of β_i. Various score functions can be taken for $c(\cdot)$. In this chapter, the score regression coefficient is denoted by w if the inverse of the uniform distribution between zero and one is taken for the score function $c(\cdot)$, ns if the inverse of the standard normal distribution is adopted, ls if the inverse of the logistic distribution is taken and cs if the inverse of the Cauchy distribution is chosen.

The t in the case where X_i and Y_i are replaced by Rx_i and Ry_i is asymptotically distributed as the standard normal distribution. This test is denoted by aw in Monte Carlo studies of Section 8.3.2. The t distribution with $n-2$ degrees of freedom is compared with the test statistic aw.

In the case where there is a constant term in the regression model, the constant term may be excluded from the regression equation, which is equivalent to the following procedure: each variable is subtracted from its arithmetic average and transformed into $c(\cdot)$. All the elements of the constant term are ones, say $x_{i,1} = 1$ for $i = 1, 2, \cdots, n$. The ranked data $Rx_{i,1})$, $i = 1, 2, \cdots, n$, take the same value. Usually, $Rx_{i,1}) = n/2$ is taken, which implies that we have $c(px_{i,1}) = 0$ for all i. Then, it is impossible to estimate β_1. Therefore, the constant term should be excluded from the regression equation. In simulation studies of Section 8.3.2, this procedure is utilized, i.e., β_1 is a constant term in Section 8.3.2, each variable of Y_i and $X_{i,j}$ for $j = 2, 3, \cdots, k$ is subtracted from its arithmetic average and the following regression equation is estimated:

$$Y_i - \overline{Y} = \sum_{j=2}^{k} \beta_j (X_{i,j} - \overline{X}_j) + \epsilon_i,$$

where $\overline{Y}$ and $\overline{X}_j$ denote the arithmetic averages of Y_i and $X_{i,j}$, $i = 1, 2, \cdots, n$, respectively. In order to perform the score tests, in Section 8.3.2 we consider the following regression model:

$$c(py_i) = \beta_2 c(px_{i,2}) + \cdots + \beta_k c(px_{i,k}) + \epsilon_i.$$

Note that the ranked data of Y_i, $X_{i,2}$, $\cdots$, $X_{i,k}$ are equivalent to those of $Y_i - \overline{Y}$, $X_{i,2} - \overline{X}_2$, $\cdots$, $X_{i,k} - \overline{X}_k$. Therefore, py_i and $px_{i,j}$ remain same.

Remark 8: In Remark 5, we have considered testing the null hypothesis $H_0 : \beta = \beta^*$. Now we discuss derivation of the confidence interval of β. The OLS estimator $\hat{\beta}$ is rewritten as:

$$\hat{\beta} = (X'X)^{-1}X'Y = \hat{\beta} + (X'X)^{-1}X'e = \hat{\beta} + \sum_{i=1}^{n} (X'X)^{-1}X_i' e_i.$$

By permuting the order of $\{e_i\}_{i=1}^n$ randomly, $\hat{\beta}$ is distributed as:

$$\hat{\beta} + \sum_{i=1}^{n} (X'X)^{-1} X_i' e_i^*,$$

where e_i^*, $i = 1, 2, \cdots, n$, denote the permutation of e_i, $i = 1, 2, \cdots, n$. The last term is interpreted as the correlation between $(X'X)^{-1}X_i'$ and e_i. Given $(X'X)^{-1}X_i'$, we permute e_i and add $\hat{\beta}$. Then, we can obtain $\hat{\beta}^{(m)}$, $m = 1, 2, \cdots, n!$. Based on the $n!$ regression coefficients, the confidence interval of β_j is constructed by sorting $\hat{\beta}_j^{(m)}$, $m = 1, 2, \cdots, n!$, by size for $j = 1, 2, \cdots, k$. For example, taking the 2.5% and 97.5% values after sorting $\hat{\beta}_j^{(m)}$, $m = 1, 2, \cdots, n!$, we can obtain the 95% confidence interval of β_j. Thus, the confidence interval of β_j is easily obtained for all $j = 1, 2, \cdots, k$. However, we should keep in mind that the amount of storage is extremely large because a $n! \times 1$ vector is required to sort $\hat{\beta}_j^{(m)}$, $m = 1, 2, \cdots, n!$. Remember that we need $10! \approx 3.6 \times 10^6$ storages even in the case of $n = 10$.

Remark 9: As discussed in Remark 4, we can reduce computational burden by choosing M out of the $n!$ permutations randomly, where we consider that each realization of $\hat{\beta}_j^{(m)}$, $m = 1, 2, \cdots, n!$, occurs with probability $1/n!$. Thus, choosing the M permutations and sorting them by size, we can obtain the testing procedure and the confidence interval. By taking some out of all the permutations randomly, it is more practical and realistic to obtain the confidence interval discussed in Remark 8.

8.3 Monte Carlo Experiments

8.3.1 On Testing the Correlation Coefficient

Each value in Table 8.1 represents the rejection rates of the null hypothesis $H_0 : \rho = 0$ against the alternative hypothesis $H_1 : \rho > 0$ (i.e., the one-sided test is chosen) by the significance level $\alpha = 0.01,\ 0.05,\ 0.10$, where the experiment is repeated G times, where $G = 10^4$ is taken in this section. That is, in Table 8.1 the number of the cases where the correlation coefficient obtained from the original observed data is greater than $1 - \alpha$ is divided by $G = 10^4$. In other words, we compute the probability of $P(\hat{\rho} < \hat{\rho}^{(0)})$, repeat the experiment G times for $G = 10^4$, and obtain the number of the cases which satisfy $P(\hat{\rho} < \hat{\rho}^{(0)}) > 1 - \alpha$ out of the G experiments, where $\hat{\rho}^{(0)}$ denotes the correlation coefficient computed from the original data. The number of the cases which satisfy $P(\hat{\rho} < \hat{\rho}^{(0)}) > 1 - \alpha$ are divided by $G = 10^4$, and the ratio corresponds to the empirical size for $\rho = 0$ or the sample power for $\rho \neq 0$. Thus, the ratio of $P(\hat{\rho} < \hat{\rho}^{(0)}) > 1 - \alpha$ relative to G simulation runs is shown in Table 8.1, where $\alpha = 0.10,\ 0.05,\ 0.01$ is examined. (B), (C), (X), (G), (D), (L), (N), (T), (U) and (Q) indicate the population distributions of (X, Y), which denote the bimodal distribution which consists of two normal distributions $0.5N(1, 1) + 0.5N(-1, 0.5^2)$,

Table 8.1: Empirical Sizes and Sample Powers ($H_0 : \rho = 0$ and $H_1 : \rho > 0$)

	n	ρ	α	f	w	ns	ls	cs	aw	t
			.10	.0970	.1113$^{\bullet\bullet\bullet}$	.1023	.1027	.1018	.1013	.1021
		0.0	.05	.0484	.0593$^{\bullet\bullet\bullet}$	.0510	.0518	.0495	.0487	.0531
			.01	.0120$^{\bullet\bullet}$	.0114	.0097	.0100	.0100	.0114	.0137$^{\bullet\bullet\bullet}$
			.10	.5105	.5302$^{\bullet}$	.5239	.5264	.4815$^{\circ\circ\circ}$	.5082	.5169
	8	0.5	.05	.3471$^{\circ\circ}$	.3801$^{\bullet\bullet}$	.3636	.3642	.3319$^{\circ\circ\circ}$	.3384$^{\circ\circ\circ}$	.3640
			.01	.1227$^{\circ\circ}$	.1261	.1257°	.1255°	.1007$^{\circ\circ\circ}$	.1261	.1338
			.10	.9885	.9699$^{\circ\circ\circ}$	.9726$^{\circ\circ\circ}$	.9726$^{\circ\circ\circ}$	.9248$^{\circ\circ\circ}$	.9666$^{\circ\circ\circ}$	.9889
		0.9	.05	.9702	.9296$^{\circ\circ\circ}$	.9302$^{\circ\circ\circ}$	.9283$^{\circ\circ\circ}$	.8639$^{\circ\circ\circ}$	.9110$^{\circ\circ\circ}$	.9733
			.01	.8400$^{\circ\circ\circ}$	.7155$^{\circ\circ\circ}$	.7244$^{\circ\circ\circ}$	.7137$^{\circ\circ\circ}$	.4775$^{\circ\circ\circ}$	.7155$^{\circ\circ\circ}$	.8696
			.10	.0993	.1024	.0987	.0988	.0984	.0962	.1023
		0.0	.05	.0501	.0548$^{\bullet\bullet}$	.0519	.0525	.0503	.0548$^{\bullet\bullet}$	.0542$^{\bullet}$
			.01	.0099	.0107	.0102	.0105	.0093	.0123$^{\bullet\bullet}$	.0122$^{\bullet\bullet}$
			.10	.5990	.6110	.6268$^{\bullet\bullet\bullet}$	.6309$^{\bullet\bullet\bullet}$	.5324$^{\circ\circ\circ}$	.5957	.6043
(B)	10	0.5	.05	.4329°	.4535	.4671$^{\bullet\bullet\bullet}$	.4668$^{\bullet\bullet\bullet}$	.3982$^{\circ\circ\circ}$	.4535	.4461
			.01	.1658$^{\circ\circ\circ}$	.1755°	.1904	.1909	.1091$^{\circ\circ\circ}$	.1949$^{\bullet}$	.1848
			.10	.9979	.9898$^{\circ\circ\circ}$	.9922$^{\circ\circ\circ}$	.9928$^{\circ\circ\circ}$	.9553$^{\circ\circ\circ}$	.9884$^{\circ\circ\circ}$	.9979
		0.9	.05	.9926	.9734$^{\circ\circ\circ}$	.9803$^{\circ\circ\circ}$	.9790$^{\circ\circ\circ}$	.9080$^{\circ\circ\circ}$	.9734$^{\circ\circ\circ}$	.9928
			.01	.9522$^{\circ\circ}$	.8674$^{\circ\circ\circ}$	.8908$^{\circ\circ\circ}$	.8840$^{\circ\circ\circ}$	.4647$^{\circ\circ\circ}$	.8810$^{\circ\circ\circ}$	.9590
			.10	.1012	.1018	.1039	.1042	.0991	.1018	.1046
		0.0	.05	.0500	.0546$^{\bullet\bullet}$	.0518	.0522	.0494	.0509	.0526
			.01	.0087	.0116	.0109	.0101	.0098	.0125$^{\bullet\bullet}$	.0115
			.10	.6643	.6734	.7019$^{\bullet\bullet\bullet}$	.7051$^{\bullet\bullet\bullet}$	.5612$^{\circ\circ\circ}$	.6734	.6682
	12	0.5	.05	.5086$^{\circ\circ}$	.5339	.5565$^{\bullet\bullet\bullet}$	.5567$^{\bullet\bullet\bullet}$	.4392$^{\circ\circ\circ}$	.5236	.5225
			.01	.2221$^{\circ\circ\circ}$	.2327°	.2583$^{\bullet\bullet}$	.2584$^{\bullet\bullet}$	.1706$^{\circ\circ\circ}$	.2435	.2431
			.10	.9991	.9967$^{\circ\circ\circ}$	.9973$^{\circ\circ\circ}$	.9975$^{\circ\circ\circ}$	.9735$^{\circ\circ\circ}$	.9967$^{\circ\circ\circ}$	.9991
		0.9	.05	.9971	.9913$^{\circ\circ\circ}$	.9945$^{\circ\circ\circ}$	.9945$^{\circ\circ\circ}$	.9330$^{\circ\circ\circ}$	.9906$^{\circ\circ\circ}$	.9971
			.01	.9860	.9441$^{\circ\circ\circ}$	.9615$^{\circ\circ\circ}$	.9596$^{\circ\circ\circ}$	.7745$^{\circ\circ\circ}$	.9481$^{\circ\circ\circ}$	.9868
			.10	.0975	.1094$^{\bullet\bullet\bullet}$	.1015	.1022	.1006	.1005	.0899$^{\circ\circ\circ}$
		0.0	.05	.0483	.0582$^{\bullet\bullet\bullet}$	.0507	.0512	.0507	.0499	.0553$^{\bullet\bullet}$
			.01	.0097	.0114	.0094	.0095	.0097	.0114	.0226$^{\bullet\bullet\bullet}$
			.10	.7421$^{\circ\circ\circ}$	.7514°	.7496$^{\circ\circ}$	.7502$^{\circ\circ}$	.6812$^{\circ\circ\circ}$	.7332$^{\circ\circ\circ}$	.7628
	8	0.5	.05	.6293°	.6269$^{\circ\circ}$	.6231$^{\circ\circ\circ}$	.6215$^{\circ\circ\circ}$	.5642$^{\circ\circ\circ}$	.5917$^{\circ\circ\circ}$	.6418
			.01	.3468$^{\circ\circ\circ}$	.3239$^{\circ\circ\circ}$	.3360$^{\circ\circ\circ}$	.3361$^{\circ\circ\circ}$	.2351$^{\circ\circ\circ}$	.3239$^{\circ\circ\circ}$	.4157
			.10	.9933	.9887$^{\circ\circ\circ}$	.9879$^{\circ\circ\circ}$	.9877$^{\circ\circ\circ}$	.9663$^{\circ\circ\circ}$	.9871$^{\circ\circ\circ}$	.9936
		0.9	.05	.9867	.9743$^{\circ\circ\circ}$	.9732$^{\circ\circ\circ}$	.9719$^{\circ\circ\circ}$	.9309$^{\circ\circ\circ}$	.9677$^{\circ\circ\circ}$	.9890
			.01	.9262$^{\circ\circ\circ}$	.8773$^{\circ\circ\circ}$	.8828$^{\circ\circ\circ}$	.8743$^{\circ\circ\circ}$	.6241$^{\circ\circ\circ}$	.8773$^{\circ\circ\circ}$	.9577
			.10	.0987	.1050$^{\bullet}$	.1035	.1026	.0984	.0983	.0886$^{\circ\circ\circ}$
		0.0	.05	.0506	.0543$^{\bullet\bullet}$	.0505	.0506	.0501	.0543$^{\bullet\bullet}$	.0556$^{\bullet\bullet}$
			.01	.0097	.0115	.0103	.0112	.0100	.0127$^{\bullet\bullet\bullet}$	.0265$^{\bullet\bullet\bullet}$
			.10	.8001$^{\circ\circ\circ}$	.8207°	.8314	.8342	.7465$^{\circ\circ\circ}$	.8085$^{\circ\circ\circ}$	.8311
(C)	10	0.5	.05	.6979$^{\circ\circ\circ}$	.7161	.7338$^{\bullet}$	.7343$^{\bullet}$	.6382$^{\circ\circ\circ}$	.7161	.7233
			.01	.4457$^{\circ\circ\circ}$	.4379$^{\circ\circ\circ}$	.4706$^{\circ\circ\circ}$	.4693$^{\circ\circ\circ}$	.2469$^{\circ\circ\circ}$	.4617$^{\circ\circ\circ}$	.4936
			.10	.9953	.9958	.9961	.9961	.9845$^{\circ\circ\circ}$	.9953	.9961
		0.9	.05	.9927	.9909$^{\circ\circ}$	.9913$^{\circ\circ}$	.9913$^{\circ\circ}$	.9606$^{\circ\circ\circ}$	.9909$^{\circ\circ}$	.9940
			.01	.9701$^{\circ\circ\circ}$	.9540$^{\circ\circ\circ}$	.9650$^{\circ\circ\circ}$	.9632$^{\circ\circ\circ}$	.6234$^{\circ\circ\circ}$	.9586$^{\circ\circ\circ}$	.9851
			.10	.1001	.1046	.1029	.1025	.0980	.1046	.0853$^{\circ\circ\circ}$
		0.0	.05	.0506	.0555$^{\bullet\bullet}$	.0532	.0522	.0499	.0535	.0560$^{\bullet\bullet\bullet}$
			.01	.0106	.0109	.0107	.0105	.0101	.0116	.0255$^{\bullet\bullet\bullet}$
			.10	.8812	.8753	.8846	.8842	.7781$^{\circ\circ\circ}$	.8753	.8795
	12	0.5	.05	.7457$^{\circ\circ\circ}$	.7952	.8078$^{\bullet\bullet\bullet}$	.8078$^{\bullet\bullet\bullet}$	.6765$^{\circ\circ\circ}$	.7887	.7900
			.01	.5383$^{\circ\circ\circ}$	.5392$^{\circ\circ\circ}$	.5840$^{\bullet\bullet\bullet}$	.5839$^{\bullet\bullet\bullet}$	.4030$^{\circ\circ\circ}$	.5517	.5631
			.10	.9971	.9993$^{\bullet\bullet\bullet}$	.9984$^{\bullet}$	.9984$^{\bullet}$	.9930$^{\circ\circ\circ}$	.9993$^{\bullet\bullet\bullet}$	.9973
		0.9	.05	.9947	.9975$^{\bullet}$	.9969	.9965	.9745$^{\circ\circ\circ}$	.9972	.9961
			.01	.9872$^{\circ\circ\circ}$	.9812$^{\circ\circ\circ}$	.9857$^{\circ\circ\circ}$	.9846$^{\circ\circ\circ}$	.8973$^{\circ\circ\circ}$	.9822$^{\circ\circ\circ}$	.9916

Table 8.1: Empirical Sizes and Sample Powers —< Continued >—

	n	ρ	α	f	w	ns	ls	cs	aw	t
(X)	8	0.0	.10	.0980	.1076••	.1020	.1006	.1003	.0979	.1273•••
			.05	.0481	.0571•••	.0507	.0507	.0518	.0481	.0808•••
			.01	.0089	.0116	.0091	.0092	.0104	.0116	.0307•••
		0.5	.10	.5527°°°	.7913•••	.7872•••	.7860•••	.6924•••	.7748•••	.5992
			.05	.4132°°°	.6480•••	.6214•••	.6189•••	.5492•••	.5975•••	.4607
			.01	.1663°°°	.2764•••	.2732•••	.2686•••	.1872°°°	.2764•••	.2463
		0.9	.10	.9985	.9958°°°	.9960°°°	.9957°°°	.9701°°°	.9953°°°	.9986
			.05	.9956	.9876°°°	.9869°°°	.9860°°°	.9354°°°	.9828°°°	.9963
			.01	.9147°°°	.9062°°°	.9002°°°	.8868°°°	.5890°°°	.9062°°°	.9530
	10	0.0	.10	.1002	.1073••	.1057•	.1059••	.1026	.1009	.1245•••
			.05	.0496	.0557•••	.0505	.0506	.0516	.0557•••	.0802•••
			.01	.0088	.0094	.0100	.0101	.0112	.0118•	.0311•••
		0.5	.10	.6070°°°	.8688•••	.8818•••	.8832•••	.7450•••	.8622•••	.6765
			.05	.4616°°°	.7608•••	.7712•••	.7691•••	.6333•••	.7608•••	.5285
			.01	.2000°°°	.4146•••	.4383•••	.4258•••	.1748°°°	.4433•••	.2888
		0.9	.10	1.000	.9996°°	.9997°	.9997°	.9876°°°	.9994°°	1.000
			.05	.9995	.9979°°°	.9983°°°	.9982°°°	.9611°°°	.9979°°°	.9996
			.01	.9860°°°	.9779°°°	.9826°°°	.9792°°°	.5675°°°	.9814°°°	.9936
	12	0.0	.10	.1019	.1039	.1047	.1048	.0977	.1039	.1208•••
			.05	.0541•	.0560•••	.0524	.0543••	.0534	.0535	.0789•••
			.01	.0098	.0110	.0110	.0104	.0111	.0116	.0321•••
		0.5	.10	.7070°°°	.9233•••	.9426•••	.9449•••	.7701••	.9233•••	.7551
			.05	.5086°°°	.8485•••	.8688•••	.8689•••	.6719•••	.8419•••	.5959
			.01	.2349°°°	.5587•••	.5824•••	.5737•••	.3015°°°	.5753•••	.3370
		0.9	.10	1.000	.9998	.9999	.9999	.9938°°°	.9998	1.000
			.05	.9999	.9995°°	.9997°	.9997°	.9740°°°	.9995°°	1.000
			.01	.9988	.9956°°°	.9970°°°	.9965°°°	.8901°°°	.9959°°°	.9993
(G)	8	0.0	.10	.0987	.1094•••	.1015	.1022	.1006	.1005	.1033
			.05	.0494	.0582•••	.0507	.0512	.0507	.0499	.0549••
			.01	.0097	.0114	.0094	.0095	.0097	.0114	.0125••
		0.5	.10	.5350	.5325°	.5143°°°	.5132°°°	.4470°°°	.5090°°°	.5447
			.05	.3725°°	.3764	.3549°°°	.3512°°°	.3106°°°	.3395°°°	.3875
			.01	.1309°°°	.1240°°°	.1162°°°	.1145°°°	.0895°°°	.1240°°°	.1498
		0.9	.10	.9912	.9798°°°	.9788°°°	.9780°°°	.9188°°°	.9767°°°	.9923
			.05	.9771	.9493°°°	.9474°°°	.9440°°°	.8565°°°	.9353°°°	.9791
			.01	.8716°°°	.7534°°°	.7494°°°	.7310°°°	.4499°°°	.7534°°°	.8980
	10	0.0	.10	.1005	.1050•	.1035	.1026	.0984	.0983	.1062••
			.05	.0490	.0543••	.0505	.0506	.0501	.0543••	.0553••
			.01	.0114	.0115	.0103	.0112	.0100	.0127•••	.0151•••
		0.5	.10	.6188°	.6055°°°	.6147°°	.6126°°°	.4878°°°	.5897°°°	.6304
			.05	.4579°°	.4519°°°	.4541°°	.4509°°°	.3674°°°	.4519°°°	.4721
			.01	.1841°°°	.1745°°°	.1785°°°	.1747°°°	.0913°°°	.1921°°°	.2099
		0.9	.10	.9988	.9942°°°	.9959°°°	.9958°°°	.9463°°°	.9935°°°	.9988
			.05	.9960	.9826°°°	.9857°°°	.9851°°°	.8939°°°	.9826°°°	.9956
			.01	.9626°°°	.8980°°°	.9059°°°	.8983°°°	.4273°°°	.9099°°°	.9718
	12	0.0	.10	.1017	.1046	.1029	.1025	.0980	.1046	.1074••
			.05	.0510	.0555••	.0532	.0522	.0499	.0535	.0558•••
			.01	.0104	.0109	.0107	.0105	.0101	.0116	.0132•••
		0.5	.10	.6843	.6713°°°	.6832	.6807	.5051°°°	.6713°°°	.6906
			.05	.5270°°°	.5252°°°	.5302°°	.5267°°°	.3871°°°	.5138°°°	.5461
			.01	.2362°°°	.2326°°°	.2373°°°	.2329°°°	.1304°°°	.2434°°°	.2657
		0.9	.10	.9997	.9986°°°	.9991°	.9992	.9624°°°	.9986°°°	.9997
			.05	.9992	.9954°°°	.9970°°°	.9973°°°	.9110°°°	.9951°°°	.9992
			.01	.9907	.9632°°°	.9726°°°	.9685°°°	.7499°°°	.9655°°°	.9926

Table 8.1: Empirical Sizes and Sample Powers —< Continued >—

	n	ρ	α	f	w	ns	ls	cs	aw	t
(D)	8	0.0	.10	.0987	.1094$^{\bullet\bullet\bullet}$	.1015	.1022	.1006	.1005	.0975
			.05	.0499	.0582$^{\bullet\bullet\bullet}$	.0507	.0512	.0507	.0499	.0507
			.01	.0093	.0114	.0094	.0095	.0097	.0114	.0114
		0.5	.10	.5700	.5587°	.5399$^{\circ\circ\circ}$	.5365$^{\circ\circ\circ}$	.4494$^{\circ\circ\circ}$	.5386$^{\circ\circ\circ}$	.5721
			.05	.4156	.4104	.3834$^{\circ\circ\circ}$	.3775$^{\circ\circ\circ}$	.3250$^{\circ\circ\circ}$	.3737$^{\circ\circ\circ}$	.4139
			.01	.1509	.1503	.1386$^{\circ\circ}$	.1347$^{\circ\circ\circ}$	.0921$^{\circ\circ\circ}$	.1503	.1494
		0.9	.10	.9896	.9790$^{\circ\circ\circ}$	.9783$^{\circ\circ\circ}$	.9780$^{\circ\circ\circ}$	.9184$^{\circ\circ\circ}$	.9762$^{\circ\circ\circ}$	.9903
			.05	.9748	.9476$^{\circ\circ\circ}$	.9423$^{\circ\circ\circ}$	.9402$^{\circ\circ\circ}$	.8593$^{\circ\circ\circ}$	.9336$^{\circ\circ\circ}$	.9770
			.01	.8772$^{\circ\circ\circ}$	.7675$^{\circ\circ\circ}$	.7617$^{\circ\circ\circ}$	.7432$^{\circ\circ\circ}$	.4513$^{\circ\circ\circ}$	.7675$^{\circ\circ\circ}$	.9022
	10	0.0	.10	.1022	.1050$^{\bullet}$	.1035	.1026	.0984	.0983	.1007
			.05	.0513	.0543$^{\bullet\bullet}$	.0505	.0506	.0501	.0543$^{\bullet\bullet}$	.0516
			.01	.0106	.0115	.0103	.0112	.0100	.0127$^{\bullet\bullet\bullet}$	.0128$^{\bullet\bullet\bullet}$
		0.5	.10	.6472	.6309$^{\circ\circ\circ}$	.6267$^{\circ\circ\circ}$	.6246$^{\circ\circ\circ}$	.4788$^{\circ\circ\circ}$	.6164$^{\circ\circ\circ}$	.6492
			.05	.5025	.4875°	.4814$^{\circ\circ}$	.4739$^{\circ\circ\circ}$	.3669$^{\circ\circ\circ}$	.4875°	.4992
			.01	.2196	.2126	.2048$^{\circ\circ\circ}$	.1983$^{\circ\circ\circ}$	.0864$^{\circ\circ\circ}$	.2316$^{\bullet}$	.2211
		0.9	.10	.9969	.9930$^{\circ\circ\circ}$	.9936$^{\circ\circ\circ}$	.9934$^{\circ\circ\circ}$	.9454$^{\circ\circ\circ}$	.9923$^{\circ\circ\circ}$	.9969
			.05	.9935	.9825$^{\circ\circ\circ}$	.9839$^{\circ\circ\circ}$	.9834$^{\circ\circ\circ}$	.8915$^{\circ\circ\circ}$	.9825$^{\circ\circ\circ}$	.9940
			.01	.9622$^{\circ\circ\circ}$	.9024$^{\circ\circ\circ}$	.9094$^{\circ\circ\circ}$	.8998$^{\circ\circ\circ}$	.4199$^{\circ\circ\circ}$	.9139$^{\circ\circ\circ}$	.9691
	12	0.0	.10	.1002	.1046	.1029	.1025	.0980	.1046	.0991
			.05	.0506	.0555$^{\bullet\bullet}$	.0532	.0522	.0499	.0535	.0519
			.01	.0113	.0109	.0107	.0105	.0101	.0116	.0129$^{\bullet\bullet\bullet}$
		0.5	.10	.7127	.6975$^{\circ\circ}$	.6974$^{\circ\circ}$	.6881$^{\circ\circ\circ}$	.4787$^{\circ\circ\circ}$	.6975$^{\circ\circ}$	.7135
			.05	.5672	.5594	.5541°	.5470$^{\circ\circ\circ}$	.3756$^{\circ\circ\circ}$	.5489$^{\circ\circ\circ}$	.5678
			.01	.2789	.2692°	.2687$^{\circ\circ}$	.2566$^{\circ\circ\circ}$	.1277$^{\circ\circ\circ}$	.2798	.2815
		0.9	.10	.9993	.9982$^{\circ\circ}$	.9983$^{\circ\circ}$	.9985$^{\circ\circ}$	.9624$^{\circ\circ\circ}$	.9982$^{\circ\circ}$	.9994
			.05	.9978	.9943$^{\circ\circ\circ}$	.9954$^{\circ\circ\circ}$	.9948$^{\circ\circ\circ}$	.9048$^{\circ\circ\circ}$	.9940$^{\circ\circ\circ}$	.9981
			.01	.9870°	.9618$^{\circ\circ\circ}$	.9658$^{\circ\circ\circ}$	.9620$^{\circ\circ\circ}$	.7409$^{\circ\circ\circ}$	.9642$^{\circ\circ\circ}$	.9897
(L)	8	0.0	.10	.0989	.1094$^{\bullet\bullet\bullet}$	.1015	.1022	.1006	.1005	.0984
			.05	.0492	.0582$^{\bullet\bullet\bullet}$	.0507	.0512	.0507	.0499	.0486
			.01	.0097	.0114	.0094	.0095	.0097	.0114	.0103
		0.5	.10	.5493	.5150$^{\circ\circ\circ}$	.4989$^{\circ\circ\circ}$	.4962$^{\circ\circ\circ}$	.4265$^{\circ\circ\circ}$	.4917$^{\circ\circ\circ}$	.5505
			.05	.3895	.3642$^{\circ\circ\circ}$	.3414$^{\circ\circ\circ}$	.3369$^{\circ\circ\circ}$	.2963$^{\circ\circ\circ}$	.3294$^{\circ\circ\circ}$	.3896
			.01	.1335	.1194$^{\circ\circ\circ}$	.1122$^{\circ\circ\circ}$	.1101$^{\circ\circ\circ}$	.0856$^{\circ\circ\circ}$	.1194$^{\circ\circ\circ}$	.1377
		0.9	.10	.9881	.9731$^{\circ\circ\circ}$	.9736$^{\circ\circ\circ}$	.9729$^{\circ\circ\circ}$	.9082$^{\circ\circ\circ}$	.9699$^{\circ\circ\circ}$	.9888
			.05	.9713	.9402$^{\circ\circ\circ}$	.9334$^{\circ\circ\circ}$	.9292$^{\circ\circ\circ}$	.8446$^{\circ\circ\circ}$	.9239$^{\circ\circ\circ}$	.9738
			.01	.8614$^{\circ\circ\circ}$	.7296$^{\circ\circ\circ}$	.7262$^{\circ\circ\circ}$	.7095$^{\circ\circ\circ}$	.4359$^{\circ\circ\circ}$	.7296$^{\circ\circ\circ}$	.8883
	10	0.0	.10	.1016	.1050$^{\bullet}$	.1035	.1026	.0984	.0983	.1019
			.05	.0499	.0543$^{\bullet\bullet}$	.0505	.0506	.0501	.0543$^{\bullet\bullet}$	.0515
			.01	.0112	.0115	.0103	.0112	.0100	.0127$^{\bullet\bullet\bullet}$	.0117$^{\bullet}$
		0.5	.10	.6266	.5880$^{\circ\circ\circ}$	.5904$^{\circ\circ\circ}$	.5861$^{\circ\circ\circ}$	.4637$^{\circ\circ\circ}$	.5748$^{\circ\circ\circ}$	.6269
			.05	.4732	.4308$^{\circ\circ\circ}$	.4283$^{\circ\circ\circ}$	.4252$^{\circ\circ\circ}$	.3447$^{\circ\circ\circ}$	.4308$^{\circ\circ\circ}$	.4748
			.01	.2010	.1660$^{\circ\circ\circ}$	.1690$^{\circ\circ\circ}$	.1646$^{\circ\circ\circ}$	.0848$^{\circ\circ\circ}$	.1830$^{\circ\circ\circ}$	.2083
		0.9	.10	.9971	.9913$^{\circ\circ\circ}$	.9932$^{\circ\circ\circ}$	.9933$^{\circ\circ\circ}$	.9372$^{\circ\circ\circ}$	.9903$^{\circ\circ\circ}$	.9971
			.05	.9932	.9788$^{\circ\circ\circ}$	.9823$^{\circ\circ\circ}$	.9801$^{\circ\circ\circ}$	.8798$^{\circ\circ\circ}$	.9788$^{\circ\circ\circ}$	.9936
			.01	.9577$^{\circ\circ}$	.8782$^{\circ\circ\circ}$	.8894$^{\circ\circ\circ}$	.8786$^{\circ\circ\circ}$	.4102$^{\circ\circ\circ}$	.8918$^{\circ\circ\circ}$	.9647
	12	0.0	.10	.1020	.1046	.1029	.1025	.0980	.1046	.1016
			.05	.0516	.0555$^{\bullet\bullet}$	.0532	.0522	.0499	.0535	.0526
			.01	.0118$^{\bullet}$	.0109	.0107	.0105	.0101	.0116	.0119$^{\bullet}$
		0.5	.10	.6937	.6486$^{\circ\circ\circ}$	.6524$^{\circ\circ\circ}$	.6489$^{\circ\circ\circ}$	.4733$^{\circ\circ\circ}$	.6486$^{\circ\circ\circ}$	.6942
			.05	.5477	.5059$^{\circ\circ\circ}$	.5036$^{\circ\circ\circ}$	.4985$^{\circ\circ\circ}$	.3578$^{\circ\circ\circ}$	.4939$^{\circ\circ\circ}$	.5483
			.01	.2618	.2203$^{\circ\circ\circ}$	.2254$^{\circ\circ\circ}$	.2198$^{\circ\circ\circ}$	.1199$^{\circ\circ\circ}$	.2294$^{\circ\circ\circ}$	.2647
		0.9	.10	.9991	.9979$^{\circ\circ}$	.9986	.9984	.9578$^{\circ\circ\circ}$	.9979$^{\circ\circ}$	.9991
			.05	.9978	.9925$^{\circ\circ\circ}$	.9947$^{\circ\circ\circ}$	.9942$^{\circ\circ\circ}$	.8984$^{\circ\circ\circ}$	.9924$^{\circ\circ\circ}$	.9982
			.01	.9872	.9499$^{\circ\circ\circ}$	.9594$^{\circ\circ\circ}$	.9541$^{\circ\circ\circ}$	.7194$^{\circ\circ\circ}$	.9534$^{\circ\circ\circ}$	.9895

Table 8.1: Empirical Sizes and Sample Powers —< Continued >—

	n	ρ	α	f	w	ns	ls	cs	aw	t
(N)	8	0.0	.10	.0970	.1021	.0960	.0972	.0989	.0933°°	.0977
			.05	.0487	.0561•••	.0492	.0489	.0490	.0487	.0494
			.01	.0101	.0108	.0096	.0097	.0089	.0108	.0092
		0.5	.10	.5303	.4875°°°	.4709°°°	.4709°°°	.4099°°°	.4646°°°	.5305
			.05	.3675	.3420°°°	.3206°°°	.3174°°°	.2832°°°	.3066°°°	.3722
			.01	.1216°°	.1097°°°	.1046°°°	.1026°°°	.0786°°°	.1097°°°	.1322
		0.9	.10	.9896	.9688°°°	.9678°°°	.9672°°°	.9077°°°	.9643°°°	.9892
			.05	.9687	.9235°°°	.9193°°°	.9149°°°	.8387°°°	.9073°°°	.9712
			.01	.8402°°°	.6972°°°	.7064°°°	.6908°°°	.4349°°°	.6972°°°	.8727
	10	0.0	.10	.1019	.1009	.0973	.0963	.1029	.0942°	.1023
			.05	.0504	.0495	.0497	.0497	.0512	.0495	.0509
			.01	.0100	.0100	.0093	.0089	.0100	.0114	.0106
		0.5	.10	.6151	.5534°°°	.5602°°°	.5595°°°	.4498°°°	.5390°°°	.6156
			.05	.4540	.4022°°°	.4015°°°	.3979°°°	.3322°°°	.4022°°°	.4580
			.01	.1848	.1554°°°	.1549°°°	.1528°°°	.0815°°°	.1692°°°	.1908
		0.9	.10	.9970	.9910°°°	.9911°°°	.9910°°°	.9332°°°	.9900°°°	.9972
			.05	.9917	.9730°°°	.9758°°°	.9751°°°	.8767°°°	.9730°°°	.9920
			.01	.9517°°°	.8560°°°	.8746°°°	.8634°°°	.4160°°°	.8709°°°	.9599
	12	0.0	.10	.1033	.1014	.1019	.1019	.1015	.1014	.1040
			.05	.0493	.0514	.0503	.0508	.0530	.0485	.0493
			.01	.0096	.0097	.0101	.0100	.0103	.0104	.0096
		0.5	.10	.6820	.6269°°°	.6319°°°	.6309°°°	.4727°°°	.6269°°°	.6806
			.05	.5379	.4812°°°	.4819°°°	.4795°°°	.3596°°°	.4709°°°	.5383
			.01	.2542	.2081°°°	.2169°°°	.2133°°°	.1218°°°	.2175°°°	.2583
		0.9	.10	.9992	.9965°°°	.9975°°°	.9978°°	.9530°°°	.9965°°°	.9992
			.05	.9977	.9913°°°	.9932°°°	.9928°°°	.8953°°°	.9908°°°	.9977
			.01	.9844	.9398°°°	.9527°°°	.9476°°°	.7232°°°	.9453°°°	.9862
(T)	8	0.0	.10	.0997	.1053•	.0986	.0976	.1008	.0962	.0998
			.05	.0486	.0550••	.0468	.0473	.0524	.0452°°	.0496
			.01	.0110	.0108	.0120••	.0121••	.0117•	.0108	.0112
		0.5	.10	.5855	.5585°°°	.5452°°°	.5397°°°	.4697°°°	.5353°°°	.5884
			.05	.4278	.4121°°°	.3924°°°	.3907°°°	.3417°°°	.3756°°°	.4301
			.01	.1640	.1481°°°	.1426°°°	.1405°°°	.1003°°°	.1481°°°	.1716
		0.9	.10	.9910	.9764°°°	.9772°°°	.9772°°°	.9281°°°	.9732°°°	.9918
			.05	.9778°	.9476°°°	.9441°°°	.9416°°°	.8698°°°	.9348°°°	.9814
			.01	.8796°°°	.7680°°°	.7653°°°	.7520°°°	.4815°°°	.7680°°°	.9107
	10	0.0	.10	.0994	.0974	.0942°	.0953	.0974	.0900°°°	.0985
			.05	.0487	.0501	.0470	.0466	.0499	.0501	.0505
			.01	.0105	.0108	.0100	.0096	.0098	.0121••	.0115
		0.5	.10	.6665	.6356°°°	.6398°°°	.6366°°°	.5090°°°	.6186°°°	.6717
			.05	.5200	.4854°°°	.4846°°°	.4774°°°	.3840°°°	.4854°°°	.5221
			.01	.2298	.2029°°°	.2051°°°	.2020°°°	.0979°°°	.2199°°	.2350
		0.9	.10	.9973	.9926°°°	.9928°°°	.9928°°°	.9516°°°	.9920°°°	.9973
			.05	.9929	.9818°°°	.9837°°°	.9828°°°	.9025°°°	.9818°°°	.9933
			.01	.9628°°	.9003°°°	.9142°°°	.9050°°°	.4544°°°	.9119°°°	.9679
	12	0.0	.10	.1003	.0966	.0961	.0956	.0983	.0966	.0995
			.05	.0499	.0509	.0478	.0481	.0473	.0481	.0502
			.01	.0092	.0084	.0094	.0102	.0090	.0093	.0119•
		0.5	.10	.7359	.7048°°°	.7145°°°	.7126°°°	.5236°°°	.7048°°°	.7349
			.05	.5917	.5655°°°	.5640°°°	.5636°°°	.4134°°°	.5540°°°	.5956
			.01	.3001	.2701°°°	.2774°°°	.2695°°°	.1558°°°	.2820°°°	.3040
		0.9	.10	.9989	.9984	.9988	.9988	.9690°°°	.9984	.9989
			.05	.9977	.9948°°°	.9959°°°	.9955°°°	.9249°°°	.9944°°°	.9980
			.01	.9894	.9653°°°	.9728°°°	.9684°°°	.7739°°°	.9674°°°	.9911

Table 8.1: Empirical Sizes and Sample Powers —< Continued >—

	n	ρ	α	f	w	ns	ls	cs	aw	t
(U)	8	0.0	.10	.1015	.1094$^{\bullet\bullet\bullet}$	.1015	.1022	.1006	.1005	.1008
			.05	.0524	.0582$^{\bullet\bullet\bullet}$	.0507	.0512	.0507	.0499	.0520
			.01	.0092	.0114	.0094	.0095	.0097	.0114	.0100
		0.5	.10	.5145	.4944$^{\circ\circ\circ}$	.4990$^{\circ\circ}$	.5047	.4827$^{\circ\circ\circ}$	.4702$^{\circ\circ\circ}$	.5153
			.05	.3528	.3433$^{\circ\circ}$	.3426$^{\circ\circ}$	.3458°	.3392$^{\circ\circ\circ}$	.3053$^{\circ\circ\circ}$	.3583
			.01	.1198	.1086$^{\circ\circ\circ}$	.1185°	.1223	.1156$^{\circ\circ}$	.1086$^{\circ\circ\circ}$	.1268
		0.9	.10	.9829	.9592$^{\circ\circ\circ}$	.9627$^{\circ\circ\circ}$	.9640$^{\circ\circ\circ}$	.9254$^{\circ\circ\circ}$	.9529$^{\circ\circ\circ}$	.9838
			.05	.9598°	.9134$^{\circ\circ\circ}$	.9196$^{\circ\circ\circ}$	.9196$^{\circ\circ\circ}$	.8671$^{\circ\circ\circ}$	.8946$^{\circ\circ\circ}$	.9646
			.01	.8234$^{\circ\circ\circ}$	.6801$^{\circ\circ\circ}$	.7121$^{\circ\circ\circ}$	.7078$^{\circ\circ\circ}$	.5120$^{\circ\circ\circ}$	.6801$^{\circ\circ\circ}$	.8584
	10	0.0	.10	.1054$^{\bullet}$	.1050$^{\bullet}$	.1035	.1026	.0984	.0983	.1048
			.05	.0509	.0543$^{\bullet\bullet}$	.0505	.0506	.0501	.0543$^{\bullet\bullet}$	.0516
			.01	.0104	.0115	.0103	.0112	.0100	.0127$^{\bullet\bullet\bullet}$	.0117$^{\bullet}$
		0.5	.10	.6044	.5621$^{\circ\circ\circ}$	.5910°	.5990	.5544$^{\circ\circ\circ}$	.5448$^{\circ\circ\circ}$	.6040
			.05	.4394	.4053$^{\circ\circ\circ}$	.4399	.4460	.4207$^{\circ\circ\circ}$	.4053$^{\circ\circ\circ}$	.4420
			.01	.1751$^{\circ\circ}$	.1506$^{\circ\circ\circ}$	.1788°	.1853	.1384$^{\circ\circ\circ}$	.1655$^{\circ\circ\circ}$	.1889
		0.9	.10	.9961	.9858$^{\circ\circ\circ}$	.9898$^{\circ\circ\circ}$	.9907$^{\circ\circ\circ}$	.9594$^{\circ\circ\circ}$	.9838$^{\circ\circ\circ}$	.9962
			.05	.9894	.9625$^{\circ\circ\circ}$	.9726$^{\circ\circ\circ}$	.9735$^{\circ\circ\circ}$	.9209$^{\circ\circ\circ}$	.9625$^{\circ\circ\circ}$	.9900
			.01	.9392$^{\circ\circ\circ}$	.8381$^{\circ\circ\circ}$	.8755$^{\circ\circ\circ}$	.8754$^{\circ\circ\circ}$	.5253$^{\circ\circ\circ}$	.8495$^{\circ\circ\circ}$	.9505
	12	0.0	.10	.1046	.1046	.1029	.1025	.0980	.1046	.1044
			.05	.0536$^{\bullet}$	.0555$^{\bullet\bullet}$	.0532	.0522	.0499	.0535	.0537$^{\bullet}$
			.01	.0110	.0109	.0107	.0105	.0101	.0116	.0121$^{\bullet\bullet}$
		0.5	.10	.6702	.6226$^{\circ\circ\circ}$	.6704	.6785	.5980$^{\circ\circ\circ}$	.6226$^{\circ\circ\circ}$	.6693
			.05	.5155	.4773$^{\circ\circ\circ}$	.5243	.5314$^{\bullet}$	.4775$^{\circ\circ\circ}$	.4673$^{\circ\circ\circ}$	.5181
			.01	.2285°	.1957$^{\circ\circ\circ}$	.2423	.2507$^{\bullet}$	.2033$^{\circ\circ\circ}$	.2059$^{\circ\circ\circ}$	.2399
		0.9	.10	.9984	.9942$^{\circ\circ\circ}$	.9972$^{\circ\circ}$	.9974°	.9753$^{\circ\circ\circ}$	.9942$^{\circ\circ\circ}$	.9985
			.05	.9967	.9861$^{\circ\circ\circ}$	.9921$^{\circ\circ\circ}$	.9925$^{\circ\circ\circ}$	.9441$^{\circ\circ\circ}$	.9856$^{\circ\circ\circ}$	.9968
			.01	.9804	.9220$^{\circ\circ\circ}$	.9523$^{\circ\circ\circ}$	.9526$^{\circ\circ\circ}$	.8107$^{\circ\circ\circ}$	.9270$^{\circ\circ\circ}$	.9833
(Q)	8	0.0	.10	.0976	.1066$^{\bullet\bullet}$	.0984	.0990	.0982	.0956	.0988
			.05	.0504	.0534	.0486	.0499	.0496	.0446$^{\circ\circ}$	.0503
			.01	.0103	.0127$^{\bullet\bullet\bullet}$	.0103	.0102	.0113	.0127$^{\bullet\bullet\bullet}$	.0116
		0.5	.10	.5210	.4779$^{\circ\circ\circ}$	.4735$^{\circ\circ\circ}$	.4766$^{\circ\circ\circ}$	.4349$^{\circ\circ\circ}$	.4576$^{\circ\circ\circ}$	.5218
			.05	.3534	.3308$^{\circ\circ\circ}$	.3195$^{\circ\circ\circ}$	.3186$^{\circ\circ\circ}$	.3000$^{\circ\circ\circ}$	.2954$^{\circ\circ\circ}$	.3597
			.01	.1205°	.1015$^{\circ\circ\circ}$	.1017$^{\circ\circ\circ}$	.1019$^{\circ\circ\circ}$	.0889$^{\circ\circ\circ}$	.1015$^{\circ\circ\circ}$	.1294
		0.9	.10	.9853	.9632$^{\circ\circ\circ}$	.9665$^{\circ\circ\circ}$	.9655$^{\circ\circ\circ}$	.9083$^{\circ\circ\circ}$	.9595$^{\circ\circ\circ}$	.9855
			.05	.9659	.9168$^{\circ\circ\circ}$	.9178$^{\circ\circ\circ}$	.9146$^{\circ\circ\circ}$	.8473$^{\circ\circ\circ}$	.8990$^{\circ\circ\circ}$	.9697
			.01	.8310$^{\circ\circ\circ}$	.6780$^{\circ\circ\circ}$	.6954$^{\circ\circ\circ}$	.6832$^{\circ\circ\circ}$	.4608$^{\circ\circ\circ}$	.6780$^{\circ\circ\circ}$	.8685
	10	0.0	.10	.1022	.1055$^{\bullet}$	.1039	.1022	.1034	.0978	.1029
			.05	.0516	.0528	.0511	.0530	.0510	.0528	.0519
			.01	.0091	.0091	.0099	.0092	.0122$^{\bullet\bullet}$	.0108	.0091
		0.5	.10	.6077	.5452$^{\circ\circ\circ}$	.5628$^{\circ\circ\circ}$	.5635$^{\circ\circ\circ}$	.4895$^{\circ\circ\circ}$	.5289$^{\circ\circ\circ}$	.6078
			.05	.4431	.3907$^{\circ\circ\circ}$	.4031$^{\circ\circ\circ}$	.4035$^{\circ\circ\circ}$	.3584$^{\circ\circ\circ}$	.3907$^{\circ\circ\circ}$	.4469
			.01	.1790	.1417$^{\circ\circ\circ}$	.1603$^{\circ\circ\circ}$	.1615$^{\circ\circ\circ}$	.1036$^{\circ\circ\circ}$	.1580$^{\circ\circ\circ}$	.1879
		0.9	.10	.9964	.9864$^{\circ\circ\circ}$	.9894$^{\circ\circ\circ}$	.9898$^{\circ\circ\circ}$	.9449$^{\circ\circ\circ}$	.9853$^{\circ\circ\circ}$	.9964
			.05	.9906	.9687$^{\circ\circ\circ}$	.9762$^{\circ\circ\circ}$	.9747$^{\circ\circ\circ}$	.8972$^{\circ\circ\circ}$	.9687$^{\circ\circ\circ}$	.9911
			.01	.9432$^{\circ\circ}$	.8492$^{\circ\circ\circ}$	.8726$^{\circ\circ\circ}$	.8689$^{\circ\circ\circ}$	.4641$^{\circ\circ\circ}$	.8633$^{\circ\circ\circ}$	.9512
	12	0.0	.10	.0970	.0970	.0965	.0959	.0973	.0970	.0968
			.05	.0494	.0514	.0488	.0477	.0472	.0485	.0495
			.01	.0105	.0095	.0102	.0110	.0092	.0104	.0112
		0.5	.10	.6810	.6117$^{\circ\circ\circ}$	.6389$^{\circ\circ\circ}$	.6418$^{\circ\circ\circ}$	.5216$^{\circ\circ\circ}$	.6117$^{\circ\circ\circ}$	.6815
			.05	.5325	.4655$^{\circ\circ\circ}$	.4880$^{\circ\circ\circ}$	.4909$^{\circ\circ\circ}$	.4014$^{\circ\circ\circ}$	.4536$^{\circ\circ\circ}$	.5316
			.01	.2371	.1867$^{\circ\circ\circ}$	.2097$^{\circ\circ\circ}$	.2108$^{\circ\circ\circ}$	.1485$^{\circ\circ\circ}$	.1969$^{\circ\circ\circ}$	.2434
		0.9	.10	.9993	.9958$^{\circ\circ\circ}$	.9973$^{\circ\circ\circ}$	.9973$^{\circ\circ\circ}$	.9626$^{\circ\circ\circ}$	.9958$^{\circ\circ\circ}$	.9993
			.05	.9975	.9886$^{\circ\circ\circ}$	.9921$^{\circ\circ\circ}$	.9927$^{\circ\circ\circ}$	.9232$^{\circ\circ\circ}$	.9878$^{\circ\circ\circ}$	.9977
			.01	.9818	.9309$^{\circ\circ\circ}$	.9516$^{\circ\circ\circ}$	.9492$^{\circ\circ\circ}$	.7666$^{\circ\circ\circ}$	.9349$^{\circ\circ\circ}$	.9847

the Cauchy distribution $(\pi(1+x^2))^{-1}$, the chi-square distribution with one degree of freedom $\chi^2(1)-1$, the Gumbel (extreme-value) distribution $\exp(-x+\alpha)\exp(-e^{-x+\alpha})$ for $\alpha = -.577216$, the double exponential (LaPlace) distribution $-0.5\exp(-|x|)$, the logistic distribution $e^{-x}(1+e^{-x})^{-2}$, the standard normal distribution $N(0,1)$, the t distribution with three degrees of freedom $t(3)$, the uniform distribution $U(-2,2)$, and the quadratic distribution $3\sqrt{5}(5-x^2)/100$, respectively. We consider the same distribution functions taken in Section 7.4. For all the distributions, however, mean is assumed to be zero in this section. The standard error of the empirical power, denoted by $\hat{p}$, is obtained by $\sqrt{\hat{p}(1-\hat{p})/G}$. For example, when $\hat{p} = 0.5$, the standard error takes the maximum value, which is 0.005.

In this chapter, the random draws of (X_i, Y_i) are obtained as follows. Let u_i and v_i be the random variables which are mutually independently distributed. Both of u_i and v_i are generated as (B), (C), (X), (G), (D), (L), (N), (T), (U) or (Q). Denote the correlation coefficient between X and Y by ρ. Given the random draws of (u_i, v_i) and ρ, (X_i, Y_i) is transformed into:

$$\begin{pmatrix} X_i \\ Y_i \end{pmatrix} = \begin{pmatrix} 1 & \frac{1-\sqrt{1-\rho^2}}{\rho} \\ \frac{1-\sqrt{1-\rho^2}}{\rho} & 1 \end{pmatrix} \begin{pmatrix} u_i \\ v_i \end{pmatrix}.$$

This transformation gives us the following features: (i) variance of X is equal to that of Y and (ii) the correlation coefficient between X and Y is ρ. In the case of the Cauchy distribution the correlation coefficient does not exist, because the Cauchy random variable has neither mean nor variance. Even in the case of the Cauchy distribution, however, we can obtain the random draws of (X_i, Y_i) given (u_i, v_i) and ρ, utilizing the above formula.

Using the artificially generated data given the true correlation coefficient $\rho = 0.0$, 0.5, 0.9, we test the null hypothesis $H_0 : \rho = 0$ against the alternative hypothesis $H_1 : \rho > 0$. Taking the significance level $\alpha = 0.10,\ 0.05,\ 0.01$ and the one-sided test, the rejection rates out of $G = 10^4$ experiments are shown in Table 8.1. In addition, taking the sample size $n = 8,\ 10,\ 12$, both the nonparametric tests and the parametric t test are reported in the table.

As discussed in Table 7.2, each value in the case of $\rho = 0$ represents the empirical size, which should be theoretically equal to the significance level α. That is, in the case of $\rho = 0$ of Table 8.1, $^{\bullet}$, $^{\bullet\bullet}$, $^{\bullet\bullet\bullet}$, $^{\circ}$, $^{\circ\circ}$ and $^{\circ\circ\circ}$ represent comparison with the significance level α. Let $\hat{p}_k$ be the sample power of k, where k takes f, w, ns, ls, cs, aw or t. We put the superscript $^{\bullet}$ when $(\hat{p}_k - \alpha)/\sqrt{\mathrm{V}(\hat{p}_k)}$ is greater than 1.6449, the superscript $^{\bullet\bullet}$ when it is greater than 1.9600, and the superscript $^{\bullet\bullet\bullet}$ when it is greater than 2.5758, where $\mathrm{V}(\hat{p}_k)$ is given by $\mathrm{V}(\hat{p}_k) = \alpha(1-\alpha)/G$ under the null hypothesis $H_0 : \rho = \alpha$, where $G = 10^4$. 1.6449, 1.9600 and 2.5758 correspond to 95%, 97.5% and 99.5% points of the standard normal distribution, respectively. We put the superscript $^{\circ}$ if $(\hat{p}_k - \alpha)/\sqrt{\mathrm{V}(\hat{p}_k)}$ is less than -1.6449, the superscript $^{\circ\circ}$ if it is less than -1.9600, and the superscript $^{\circ\circ\circ}$ if it is less than -2.5758. Therefore, the values with the superscript $^{\bullet}$ indicate over-estimation of the empirical power and the values with the superscript

$^{\circ}$ represent under-estimation of the empirical power.

Moreover, in the case of $\rho = 0.5,\ 0.9$ of Table 8.1, $^{\bullet}$, $^{\bullet\bullet}$, $^{\bullet\bullet\bullet}$, $^{\circ}$, $^{\circ\circ}$ and $^{\circ\circ\circ}$ indicate comparison with the t test. k takes f, w, ns, ls, cs or aw in this case. We put the superscript $^{\bullet}$ when $(\hat{p}_k - \hat{p}_t)/\sqrt{\mathrm{V}(\hat{p}_k) + \mathrm{V}(\hat{p}_t)}$ is greater than 1.6449, the superscript $^{\bullet\bullet}$ when it is greater than 1.9600, and the superscript $^{\bullet\bullet\bullet}$ when it is greater than 2.5758. The two variances are approximated as: $\mathrm{V}(\hat{p}_k) \approx \hat{p}_k(1 - \hat{p}_k)/G$ and $\mathrm{V}(\hat{p}_t) \approx \hat{p}_t(1 - \hat{p}_t)/G$, where $G = 10^4$. We put the superscript $^{\circ}$ if $(\hat{p}_k - \hat{p}_t)/\sqrt{\mathrm{V}(\hat{p}_k) + \mathrm{V}(\hat{p}_t)}$ is less than -1.6449, the superscript $^{\circ\circ}$ if it is less than -1.9600, and the superscript $^{\circ\circ\circ}$ if it is less than -2.5758. Note that in large sample we have the following: $(\hat{p}_k - \hat{p}_t)/\sqrt{\mathrm{V}(\hat{p}_k) + \mathrm{V}(\hat{p}_t)} \sim N(0, 1)$ under the null hypothesis $H_0 : \rho = 0$ and the alternative one $H_1 : \rho \neq 0$. Therefore, the values with the superscript $^{\bullet}$ indicate more powerful test than the t test. In addition, the number of the superscript $^{\bullet}$ shows degree of the sample power. Contrarily, the values with the superscript $^{\circ}$ represent less powerful test than the t test.

As it is easily expected, for the normal sample (N), the t test performs better than the nonparametric tests, but for the other samples such as (B), (C), (X), (G), (D), (L), (T), (U) and (Q), the nonparametric tests are superior to the t test. We discuss the cases of $\rho = 0$ and those of $\rho \neq 0$ separately.

Empirical Size ($\rho = 0$): First, we focus only on the empirical sizes, which correspond to the cases of $\rho = 0$ in Table 8.1. The results of the t test are in the last column of Table 8.1. For the cases of $\rho = 0$ in (C), (X), (G), some cases of (B), and those of (D), we can see that the t test does not work, because the empirical sizes are statistically different from the significance levels. Especially, for t in (C), as α is large, the empirical size increases more than proportionally. We can see that the t statistic of (C) is distributed with much fatter tails than the t distribution with $n - 2$ degrees of freedom. In the case where the population distribution is (N), t shows a good performance, because t has the $t(n - 2)$ distribution even in small sample.

We discuss aw in the case of $\rho = 0$. aw performs better than t with respect to the empirical sizes, because the empirical sizes of aw are close to the significance levels α, compared with the sizes of t.

However, it is easily shown from Table 8.1 that f, ns, ls and cs are much better than w, aw and t in the sense of the empirical size. The cases of $\rho = 0$ which have neither $^{\bullet}$ nor $^{\circ}$ in the superscript are 85 out of 90 (i.e., 3 sample sizes $\times$ 3 significance levels $\times$ 10 population distributions = 90 cases) for f, 70 for w, 87 for ns, 87 for ls, 89 for cs (only in the case of $n = 10$, $\alpha = 0.01$ and (Q), the empirical size of cs is significantly over-estimated), 68 for aw, and 53 for t, respectively. Thus, in the size criterion, cs shows the best performance, but it is not too different from f, ns and ls.

Sample Power ($\rho \neq 0$): Now, we discuss the cases of $\rho = 0.5,\ 0.9$, which represent the sample powers. As it is well known, under normality assumption, t gives us the uniformly most powerful test. Therefore, for (N), t should be the best test. As a result,

Figure 8.1: Sample Powers: $n = 10$ and $\alpha = 0.10$

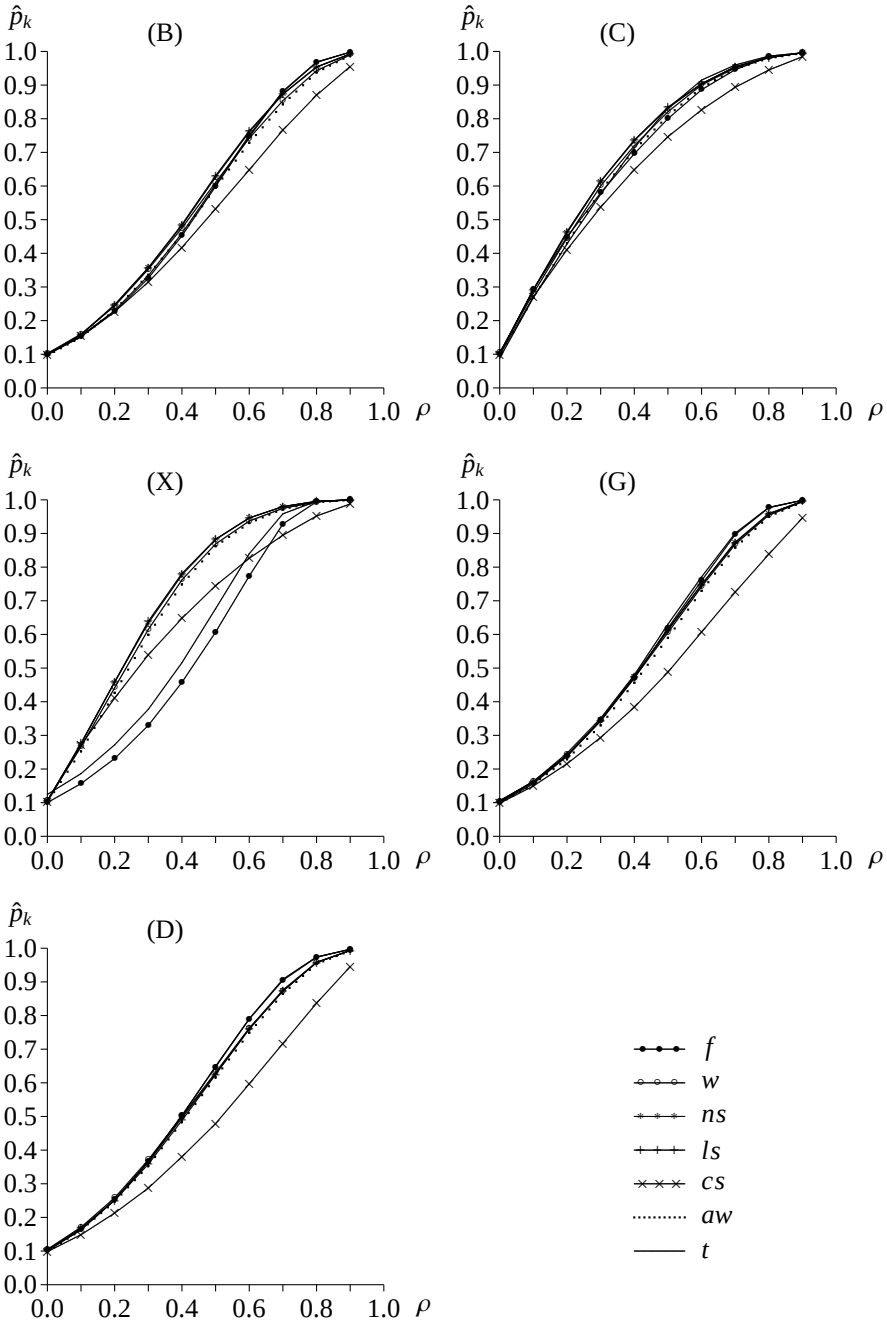
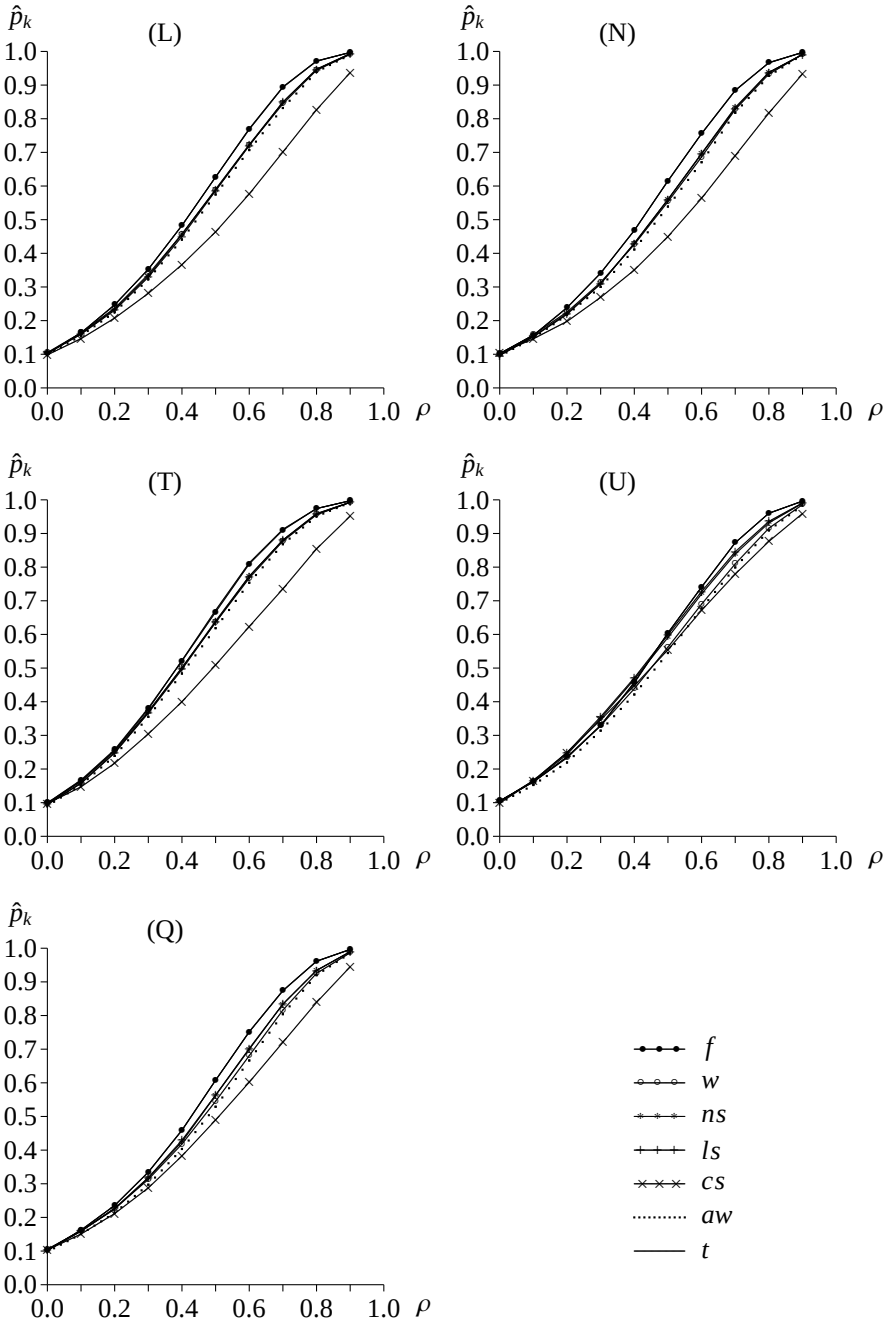


Figure 8.1: Sample Powers: $n = 10$ and $\alpha = 0.10$ —< Continued >—



in the case of (N), t is more powerful than any other tests for all $n = 8, 10, 12$, $\rho = 0.5$, 0.9 and $\alpha = 0.10, 0.05, 0.01$, i.e., all the values in t are larger than any other tests. Moreover, we obtain the result that t is the best in the sample power criterion when t works well in the empirical size criterion even if the population distribution is not normal. That is, as mentioned above, for $\rho \neq 0$, $^\bullet$ or $^\circ$ in f, w, ns, ls, cs and aw indicates comparison with the t test. In the cases where t works well for $\rho = 0$ and $\alpha = 0.1,\ 0.05,\ 0.01$ (i.e., $n = 12$ of (B), $n = 8$ of (D), $n = 8$ of (L), $n = 8, 10, 12$ of (N), $n = 8, 10$ of (T), $n = 8$ of (U) and $n = 8, 10, 12$ of (Q)), t shows the most more powerful test from Table 8.1, i.e., in this case, there are a lot of values which have $^\circ$ in the superscript. When t works well in the empirical size criterion, t is more powerful than f but t is not too different from f, i.e., f is also quite powerful.

Even in the case where t does not have an appropriate empirical size, we see the result that f, ns, ls and cs show quite good performances. When we compare f, ns, ls and cs in this situation, we can find that cs is much less powerful than f, ns and ls, although cs shows the best performance in the size criterion. f, ns and ls are more powerful than the other tests, and f is slightly less powerful than ns and ls. Thus, f, ns and ls are relatively better than the other tests in both empirical size and sample power criteria.

Even when t works well in the empirical size criterion, f is better than w, ns, ls, cs and aw (see $n = 12$ and $\rho = 0.9$ of (B), $n = 8$ of (D), $n = 8$ of (L), (N), $n = 8, 10$ of (T), $n = 8$ of (U) and (Q) in Table8.1).

Taking the case of $n = 10$ and $\alpha = 0.10$, Figure 8.1 represents the sample powers for $\rho = 0.0, 0.1, \cdots, 0.9$. In the case of $n = 10$, $\alpha = 0.10$ and $\rho = 0.0,\ 0.5,\ 0.9$, the values in Table 8.1 are utilized to draw Figure 8.1. For each population distribution, the features of the nonparametric and parametric tests are summarized as follows:

(B) It is difficult to distinguish all the lines except for cs. cs is not as powerful as the other tests. For $\rho = 0.7,\ 0.8$, t is the most powerful test and w and aw are less powerful than f, ns, ls and t, but f, w, ns, ls, aw and t are very similar to each other.

(C) We obtain the same results as (B). That is, we cannot distinguish all the lines except for cs. cs is less powerful than any other tests.

(X) f is parallel to t, but both are not as powerful as w, ns, ls and aw. cs intersects f and t. cs is more powerful than f and t for large ρ, but it is less powerful for small ρ. In any case, cs is not powerful, either.

(G) The sample power of cs is less than any other tests for all $\rho = 0.0, 0.1, \cdots, 0.9$. f and t is slightly better than w, ns, ls and aw.

(D) (D) has the exactly same features as (G). That is, cs is inferior to other tests for all $\rho = 0.0, 0.1, \cdots, 0.9$. f and t is slightly better than w, ns, ls and aw.

(L) There are three lines to be distinguished. f and t are the upper lines, cs is the lower line, and w, ns, ls and aw are in the middle.

(N) We have the same results as (L). *cs* shows a poor performance, while f and t are powerful.

(T) This case also has the same features as (L) and (N).

(U) *aw* is lower than *cs* for small ρ, but not for large ρ. For $\rho = 0.6,\ 0.7,\ 0.8$, f and t are more powerful than the other tests.

(Q) *cs* is the lowest, while f and t are the highest.

Thus, *cs* shows a poor performance for almost all the population distributions. f is as powerful as t for all the distributions, and it is the most powerful.

Summarizing above, it might be concluded that the permutation-based nonparametric test, f, is useful, because it gives us the correct empirical sizes and is quite powerful even though it does not need to assume the distribution function.

8.3.2 On Testing the Regression Coefficient

In Table 8.2, the testing procedure taken in Section 8.3.1 is applied to the regression analysis. The error term ϵ_i is assumed to have the bimodal distribution which consists of two normal distributions $0.5N(1,1) + 0.5N(-1, 0.5^2)$, the Cauchy distribution $(\pi(1+x^2))^{-1}$, the chi-square distribution with one degree of freedom $\chi^2(1) - 1$, the Gumbel (extreme-value) distribution $\exp(-x+\alpha)\exp(-e^{-x+\alpha})$ for $\alpha = -.577216$, the double exponential (LaPlace) distribution $-0.5\exp(-|x|)$, the logistic distribution $e^{-x}(1+e^{-x})^{-2}$, the standard normal distribution $N(0,1)$, the t distribution with three degrees of freedom $t(3)$, the uniform distribution $U(-2,2)$, and the quadratic distribution $3\sqrt{5}(5-x^2)/100$, which are denoted by (B), (C), (X), (G), (D), (L), (N), (T), (U) and (Q), respectively. Let $X_i = (X_{1,i}, X_{2,i}, \cdots, X_{k,i})$, where $X_{1,i} = 1$ and $(\beta_1, \beta_2, \beta_3, \beta_4) = (0.0,\ 0.0,\ 0.5,\ 0.9)$ are set. $X_{j,i}$ for $j = 2, 3, 4$ are generated from be the same distribution as the error term ϵ_i, where $X_{2,i}$, $X_{3,i}$ and $X_{4,i}$ are assumed to be mutually independent. Under the setup above, we obtain a series of data $\{Y_i\}$ from $Y_i = X_i\beta + \epsilon_i$. The regression coefficient estimate is given by $\hat{\beta} = \sum_{i=1}^{n}(X'X)^{-1}X_iY_i$, which indicates the sample covariance between $(X'X)^{-1}X_i$ and Y_i. Therefore, the significance test on the regression coefficient is equivalent to testing whether the correlation coefficient between $(X'X)^{-1}X_i$ and Y_i is zero.

The sample size is $n = 9,\ 10,\ 11$ and the number of the regression coefficient to be estimated is $k = 4$. The nonparametric tests are compared with the t test in both empirical size and sample power criteria. Each value in Table 8.2 represents the rejection rate out of G simulation runs, where $G = 10^4$. As in Table 8.1, we generate data (Y_i, X_i) for $i = 1, 2, \cdots, n$ for each distribution, compute the probability of $P(\hat{\beta}_j < \hat{\beta}_j^{(0)})$ given the generated data, repeat the experiment G times for $G = 10^4$, and obtain the number of the cases which satisfy $P(\hat{\beta}_j < \hat{\beta}_j^{(0)}) > 1 - \alpha$ out of the G experiments, where $\hat{\beta}_j^{(0)}$ denotes the jth regression coefficient computed from the original data. The number of the cases divided by 10^4 correspond to the empirical sizes or the sample powers. Thus, the ratio of $P(\hat{\beta}_j < \hat{\beta}_j^{(0)}) > 1 - \alpha$ is shown in Table

Table 8.2: Empirical Sizes and Sample Powers ($H_0: \beta_i = 0$ for $i = 2, 3, 4$)

	n	β	α	f	w	ns	ls	cs	aw	t
(B)	9	β_2	.10	.0921$^{\circ\circ\circ}$	.1041	.0944°	.0950°	.0988	.1018	.0973
			.05	.0457$^{\circ\circ}$	.0509	.0469	.0467	.0477	.0525	.0511
			.01	.0095	.0098	.0083°	.0087	.0087	.0112	.0125$^{\bullet\bullet}$
		β_3	.10	.3578$^{\circ\circ\circ}$	.3460$^{\circ\circ\circ}$	.3348$^{\circ\circ\circ}$	.3328$^{\circ\circ\circ}$	.2938$^{\circ\circ\circ}$	.3578$^{\circ\circ\circ}$	.4120
			.05	.2221$^{\circ\circ\circ}$	.2183$^{\circ\circ\circ}$	.2040$^{\circ\circ\circ}$	.2032$^{\circ\circ\circ}$	.1846$^{\circ\circ\circ}$	.2241$^{\circ\circ\circ}$	.2675
			.01	.0612$^{\circ\circ\circ}$	.0579$^{\circ\circ\circ}$	.0589$^{\circ\circ\circ}$	.0575$^{\circ\circ\circ}$	.0454$^{\circ\circ\circ}$	.0662$^{\circ\circ\circ}$	.0842
		β_4	.10	.7568$^{\bullet\bullet\bullet}$	.6912$^{\circ\circ\circ}$	.6819$^{\circ\circ\circ}$	.6785$^{\circ\circ\circ}$	.5540$^{\circ\circ\circ}$	.6226$^{\circ\circ\circ}$	.7135
			.05	.5995$^{\bullet\bullet\bullet}$	.5365$^{\circ\circ}$	.5193$^{\circ\circ\circ}$	.5112$^{\circ\circ\circ}$	.4214$^{\circ\circ\circ}$	.4608$^{\circ\circ\circ}$	.5521
			.01	.2773$^{\bullet\bullet\bullet}$	.2329	.2235$^{\circ\circ\circ}$	.2166$^{\circ\circ\circ}$	.1210$^{\circ\circ\circ}$	.1880$^{\circ\circ\circ}$	.2423
	10	β_2	.10	.0974	.0957	.0937$^{\circ\circ}$	.0942°	.0971	.0980	.1010
			.05	.0489	.0500	.0468	.0469	.0475	.0487	.0513
			.01	.0087	.0095	.0088	.0087	.0092	.0100	.0101
		β_3	.10	.3868$^{\circ\circ\circ}$	.3670$^{\circ\circ\circ}$	.3681$^{\circ\circ\circ}$	.3663$^{\circ\circ\circ}$	.3034$^{\circ\circ\circ}$	.3916$^{\circ\circ\circ}$	.4513
			.05	.2407$^{\circ\circ\circ}$	.2298$^{\circ\circ\circ}$	.2272$^{\circ\circ\circ}$	.2253$^{\circ\circ\circ}$	.1981$^{\circ\circ\circ}$	.2490$^{\circ\circ\circ}$	.2966
			.01	.0712$^{\circ\circ\circ}$	.0601$^{\circ\circ\circ}$	.0629$^{\circ\circ\circ}$	.0634$^{\circ\circ\circ}$	.0450$^{\circ\circ\circ}$	.0729$^{\circ\circ\circ}$	.0988
		β_4	.10	.8007$^{\bullet\bullet\bullet}$	.7323$^{\circ\circ\circ}$	.7350$^{\circ\circ\circ}$	.7323$^{\circ\circ\circ}$	.5686$^{\circ\circ\circ}$	.6892$^{\circ\circ\circ}$	.7711
			.05	.6643$^{\bullet\bullet\bullet}$	.5921$^{\circ\circ\circ}$	.5845$^{\circ\circ\circ}$	.5774$^{\circ\circ\circ}$	.4474$^{\circ\circ\circ}$	.5353$^{\circ\circ\circ}$	.6304
			.01	.3372$^{\bullet\bullet\bullet}$	.2719$^{\circ\circ\circ}$	.2725$^{\circ\circ\circ}$	.2632$^{\circ\circ\circ}$	.1194$^{\circ\circ\circ}$	.2352$^{\circ\circ\circ}$	.3042
	11	β_2	.10	.0954	.0992	.0972	.0987	.1038	.1043	.1033
			.05	.0459°	.0494	.0504	.0503	.0504	.0523	.0556$^{\bullet\bullet}$
			.01	.0094	.0097	.0096	.0095	.0108	.0100	.0110
		β_3	.10	.4061$^{\circ\circ\circ}$	.3770$^{\circ\circ\circ}$	.3799$^{\circ\circ\circ}$	.3788$^{\circ\circ\circ}$	.3117$^{\circ\circ\circ}$	.4183$^{\circ\circ\circ}$	.4900
			.05	.2686$^{\circ\circ\circ}$	.2411$^{\circ\circ\circ}$	.2453$^{\circ\circ\circ}$	.2428$^{\circ\circ\circ}$	.2053$^{\circ\circ\circ}$	.2774$^{\circ\circ\circ}$	.3344
			.01	.0883$^{\circ\circ\circ}$	.0757$^{\circ\circ\circ}$	.0783$^{\circ\circ\circ}$	.0751$^{\circ\circ\circ}$	.0481$^{\circ\circ\circ}$	.0926$^{\circ\circ\circ}$	.1205
		β_4	.10	.8380$^{\bullet\bullet\bullet}$	.7702$^{\circ\circ\circ}$	.7771$^{\circ\circ\circ}$	.7743$^{\circ\circ\circ}$	.5742$^{\circ\circ\circ}$	.7412$^{\circ\circ\circ}$	.8240
			.05	.7086$^{\bullet\bullet}$	.6305$^{\circ\circ\circ}$	.6338$^{\circ\circ\circ}$	.6273$^{\circ\circ\circ}$	.4648$^{\circ\circ\circ}$	.5975$^{\circ\circ\circ}$	.6939
			.01	.3899$^{\bullet\bullet\bullet}$	.3222$^{\circ\circ\circ}$	.3240$^{\circ\circ\circ}$	.3129$^{\circ\circ\circ}$	.1476$^{\circ\circ\circ}$	.2954$^{\circ\circ\circ}$	.3703
(C)	9	β_2	.10	.0988	.1059$^{\bullet\bullet}$	.0984	.0989	.0986	.0981	.0965
			.05	.0486	.0524	.0478	.0483	.0485	.0473	.0567$^{\bullet\bullet\bullet}$
			.01	.0098	.0098	.0087	.0091	.0094	.0094	.0208$^{\bullet\bullet\bullet}$
		β_3	.10	.4055$^{\circ\circ\circ}$	.3940$^{\circ\circ\circ}$	.3958$^{\circ\circ\circ}$	.3956$^{\circ\circ\circ}$	.3733$^{\circ\circ\circ}$	.4034$^{\circ\circ\circ}$	.5008
			.05	.2914$^{\circ\circ\circ}$	.2643$^{\circ\circ\circ}$	.2668$^{\circ\circ\circ}$	.2674$^{\circ\circ\circ}$	.2544$^{\circ\circ\circ}$	.2705$^{\circ\circ\circ}$	.4115
			.01	.1135$^{\circ\circ\circ}$	.0967$^{\circ\circ\circ}$	.1018$^{\circ\circ\circ}$	.0998$^{\circ\circ\circ}$	.0778$^{\circ\circ\circ}$	.0896$^{\circ\circ\circ}$	.2675
		β_4	.10	.5835$^{\circ\circ\circ}$	.5995$^{\circ\circ\circ}$	.5983$^{\circ\circ\circ}$	.6007$^{\circ\circ\circ}$	.5719$^{\circ\circ\circ}$	.5636$^{\circ\circ\circ}$	.6414
			.05	.4812$^{\circ\circ\circ}$	.4611$^{\circ\circ\circ}$	.4717$^{\circ\circ\circ}$	.4740$^{\circ\circ\circ}$	.4411$^{\circ\circ\circ}$	.4234$^{\circ\circ\circ}$	.5604
			.01	.2656$^{\circ\circ\circ}$	.2203$^{\circ\circ\circ}$	.2372$^{\circ\circ\circ}$	.2360$^{\circ\circ\circ}$	.1735$^{\circ\circ\circ}$	.1898$^{\circ\circ\circ}$	.4076
	10	β_2	.10	.0990	.1066$^{\bullet\bullet}$	.1032	.1025	.0985	.1001	.0952
			.05	.0491	.0541$^{\bullet}$	.0522	.0512	.0488	.0478	.0589$^{\bullet\bullet\bullet}$
			.01	.0104	.0090	.0090	.0090	.0116	.0098	.0216$^{\bullet\bullet\bullet}$
		β_3	.10	.4212$^{\circ\circ\circ}$	.4035$^{\circ\circ\circ}$	.4171$^{\circ\circ\circ}$	.4214$^{\circ\circ\circ}$	.3945$^{\circ\circ\circ}$	.4309$^{\circ\circ\circ}$	.5195
			.05	.3092$^{\circ\circ\circ}$	.2815$^{\circ\circ\circ}$	.2930$^{\circ\circ\circ}$	.2954$^{\circ\circ\circ}$	.2759$^{\circ\circ\circ}$	.2964$^{\circ\circ\circ}$	.4399
			.01	.1314$^{\circ\circ\circ}$	.1012$^{\circ\circ\circ}$	.1134$^{\circ\circ\circ}$	.1158$^{\circ\circ\circ}$	.0801$^{\circ\circ\circ}$	.1090$^{\circ\circ\circ}$	.3066
		β_4	.10	.6109$^{\circ\circ\circ}$	.6286$^{\circ\circ\circ}$	.6367$^{\circ\circ\circ}$	.6401$^{\circ\circ\circ}$	.5951$^{\circ\circ\circ}$	.6055$^{\circ\circ\circ}$	.6682
			.05	.5048$^{\circ\circ\circ}$	.4953$^{\circ\circ\circ}$	.5113$^{\circ\circ\circ}$	.5140$^{\circ\circ\circ}$	.4709$^{\circ\circ\circ}$	.4732$^{\circ\circ\circ}$	.5887
			.01	.2962$^{\circ\circ\circ}$	.2397$^{\circ\circ\circ}$	.2705$^{\circ\circ\circ}$	.2703$^{\circ\circ\circ}$	.1725$^{\circ\circ\circ}$	.2221$^{\circ\circ\circ}$	.4483
	11	β_2	.10	.0973	.1024	.1005	.1012	.0976	.0979	.0946°
			.05	.0467	.0463°	.0472	.0470	.0479	.0494	.0564$^{\bullet\bullet\bullet}$
			.01	.0092	.0091	.0096	.0086	.0088	.0086	.0213$^{\bullet\bullet\bullet}$
		β_3	.10	.4442$^{\circ\circ\circ}$	.4323$^{\circ\circ\circ}$	.4443$^{\circ\circ\circ}$	.4473$^{\circ\circ\circ}$	.4033$^{\circ\circ\circ}$	.4753$^{\circ\circ\circ}$	.5433
			.05	.3222$^{\circ\circ\circ}$	.2986$^{\circ\circ\circ}$	.3173$^{\circ\circ\circ}$	.3172$^{\circ\circ\circ}$	.2924$^{\circ\circ\circ}$	.3368$^{\circ\circ\circ}$	.4614
			.01	.1449$^{\circ\circ\circ}$	.1145$^{\circ\circ\circ}$	.1339$^{\circ\circ\circ}$	.1363$^{\circ\circ\circ}$	.0965$^{\circ\circ\circ}$	.1332$^{\circ\circ\circ}$	.3261
		β_4	.10	.6394$^{\circ\circ\circ}$	.6586$^{\circ\circ\circ}$	.6705$^{\circ\circ}$	.6718$^{\circ\circ}$	.6188$^{\circ\circ\circ}$	.6468$^{\circ\circ\circ}$	.6853
			.05	.5295$^{\circ\circ\circ}$	.5211$^{\circ\circ\circ}$	.5464$^{\circ\circ\circ}$	.5517$^{\circ\circ\circ}$	.5031$^{\circ\circ\circ}$	.5129$^{\circ\circ\circ}$	.6135
			.01	.3265$^{\circ\circ\circ}$	.2744$^{\circ\circ\circ}$	.3080$^{\circ\circ\circ}$	.3097$^{\circ\circ\circ}$	.2120$^{\circ\circ\circ}$	.2653$^{\circ\circ\circ}$	.4809

Table 8.2: Empirical Sizes and Sample Powers —< Continued >—

	n	β	α	f	w	ns	ls	cs	aw	t
(X)	9	β_2	.10	.0981	.1026	.1017	.1013	.1009	.0984	.1198$^{\bullet\bullet\bullet}$
			.05	.0504	.0521	.0488	.0485	.0526	.0487	.0744$^{\bullet\bullet\bullet}$
			.01	.0102	.0109	.0093	.0093	.0089	.0113	.0267$^{\bullet\bullet\bullet}$
		β_3	.10	.3415$^{\circ\circ\circ}$	.3652$^{\circ\circ\circ}$	.3558$^{\circ\circ\circ}$	.3545$^{\circ\circ\circ}$	.3072$^{\circ\circ\circ}$	.3770$^{\circ\circ\circ}$	.4679
			.05	.2220$^{\circ\circ\circ}$	.2275$^{\circ\circ\circ}$	.2239$^{\circ\circ\circ}$	.2214$^{\circ\circ\circ}$	.1968$^{\circ\circ\circ}$	.2401$^{\circ\circ\circ}$	.3514
			.01	.0677$^{\circ\circ\circ}$	.0689$^{\circ\circ\circ}$	.0650$^{\circ\circ\circ}$	.0636$^{\circ\circ\circ}$	.0479$^{\circ\circ\circ}$	.0716$^{\circ\circ\circ}$	.1777
		β_4	.10	.6428$^{\circ\circ\circ}$	.6296$^{\circ\circ\circ}$	.6239$^{\circ\circ\circ}$	.6215$^{\circ\circ\circ}$	.5329$^{\circ\circ\circ}$	.5755$^{\circ\circ\circ}$	.6678
			.05	.5114$^{\circ\circ\circ}$	.4804$^{\circ\circ\circ}$	.4748$^{\circ\circ\circ}$	.4714$^{\circ\circ\circ}$	.4036$^{\circ\circ\circ}$	.4286$^{\circ\circ\circ}$	.5601
			.01	.2489$^{\circ\circ\circ}$	.2118$^{\circ\circ\circ}$	.2117$^{\circ\circ\circ}$	.2101$^{\circ\circ\circ}$	.1244$^{\circ\circ\circ}$	.1849$^{\circ\circ\circ}$	.3493
	10	β_2	.10	.1025	.1035	.1031	.1038	.1031	.1000	.1191$^{\bullet\bullet\bullet}$
			.05	.0512	.0528	.0499	.0493	.0478	.0505	.0730$^{\bullet\bullet\bullet}$
			.01	.0111	.0093	.0088	.0087	.0095	.0097	.0263$^{\bullet\bullet\bullet}$
		β_3	.10	.3599$^{\circ\circ\circ}$	.3754$^{\circ\circ\circ}$	.3745$^{\circ\circ\circ}$	.3761$^{\circ\circ\circ}$	.3119$^{\circ\circ\circ}$	.4026$^{\circ\circ\circ}$	.4950
			.05	.2431$^{\circ\circ\circ}$	.2424$^{\circ\circ\circ}$	.2391$^{\circ\circ\circ}$	.2364$^{\circ\circ\circ}$	.2007$^{\circ\circ\circ}$	.2680$^{\circ\circ\circ}$	.3747
			.01	.0730$^{\circ\circ\circ}$	.0755$^{\circ\circ\circ}$	.0766$^{\circ\circ\circ}$	.0752$^{\circ\circ\circ}$	.0510$^{\circ\circ\circ}$	.0917$^{\circ\circ\circ}$	.1964
		β_4	.10	.6770$^{\circ\circ\circ}$	.6562$^{\circ\circ\circ}$	.6605$^{\circ\circ\circ}$	.6611$^{\circ\circ\circ}$	.5487$^{\circ\circ\circ}$	.6218$^{\circ\circ\circ}$	.6984
			.05	.5532$^{\circ\circ\circ}$	.5177$^{\circ\circ\circ}$	.5173$^{\circ\circ\circ}$	.5148$^{\circ\circ\circ}$	.4193$^{\circ\circ\circ}$	.4809$^{\circ\circ\circ}$	.5962
			.01	.2846$^{\circ\circ\circ}$	.2406$^{\circ\circ\circ}$	.2523$^{\circ\circ\circ}$	.2459$^{\circ\circ\circ}$	.1152$^{\circ\circ\circ}$	.2298$^{\circ\circ\circ}$	.3941
	11	β_2	.10	.0990	.0998	.0972	.0976	.0974	.0984	.1171$^{\bullet\bullet\bullet}$
			.05	.0477	.0477	.0468	.0469	.0480	.0492	.0714$^{\bullet\bullet\bullet}$
			.01	.0104	.0089	.0097	.0093	.0111	.0091	.0264$^{\bullet\bullet\bullet}$
		β_3	.10	.3797$^{\circ\circ\circ}$	.3980$^{\circ\circ\circ}$	.4013$^{\circ\circ\circ}$	.3970$^{\circ\circ\circ}$	.3173$^{\circ\circ\circ}$	.4351$^{\circ\circ\circ}$	.5174
			.05	.2502$^{\circ\circ\circ}$	.2540$^{\circ\circ\circ}$	.2647$^{\circ\circ\circ}$	.2631$^{\circ\circ\circ}$	.2153$^{\circ\circ\circ}$	.2968$^{\circ\circ\circ}$	.3987
			.01	.0791$^{\circ\circ\circ}$	.0873$^{\circ\circ\circ}$	.0880$^{\circ\circ\circ}$	.0877$^{\circ\circ\circ}$	.0566$^{\circ\circ\circ}$	.1090$^{\circ\circ\circ}$	.2247
		β_4	.10	.7053$^{\circ\circ\circ}$	.6876$^{\circ\circ\circ}$	.6935$^{\circ\circ\circ}$	.6925$^{\circ\circ\circ}$	.5604$^{\circ\circ\circ}$	.6665$^{\circ\circ\circ}$	.7342
			.05	.5871$^{\circ\circ\circ}$	.5463$^{\circ\circ\circ}$	.5590$^{\circ\circ\circ}$	.5569$^{\circ\circ\circ}$	.4421$^{\circ\circ\circ}$	.5278$^{\circ\circ\circ}$	.6374
			.01	.3237$^{\circ\circ\circ}$	.2772$^{\circ\circ\circ}$	.2828$^{\circ\circ\circ}$	.2782$^{\circ\circ\circ}$	.1424$^{\circ\circ\circ}$	.2653$^{\circ\circ\circ}$	.4406
(G)	9	β_2	.10	.0974	.1061$^{\bullet\bullet}$	.1000	.1000	.1035	.1026	.1009
			.05	.0479	.0545$^{\bullet\bullet}$	.0514	.0514	.0521	.0520	.0509
			.01	.0086	.0104	.0106	.0098	.0096	.0112	.0105
		β_3	.10	.3645$^{\circ\circ\circ}$	.3518$^{\circ\circ\circ}$	.3458$^{\circ\circ\circ}$	.3462$^{\circ\circ\circ}$	.3005$^{\circ\circ\circ}$	.3635$^{\circ\circ\circ}$	.4320
			.05	.2302$^{\circ\circ\circ}$	.2278$^{\circ\circ\circ}$	.2197$^{\circ\circ\circ}$	.2174$^{\circ\circ\circ}$	.1909$^{\circ\circ\circ}$	.2328$^{\circ\circ\circ}$	.2921
			.01	.0714$^{\circ\circ\circ}$	.0668$^{\circ\circ\circ}$	.0639$^{\circ\circ\circ}$	.0621$^{\circ\circ\circ}$	.0484$^{\circ\circ\circ}$	.0707$^{\circ\circ\circ}$	.0966
		β_4	.10	.7364$^{\bullet\bullet\bullet}$	.6829$^{\circ\circ\circ}$	.6741$^{\circ\circ\circ}$	.6747$^{\circ\circ\circ}$	.5780$^{\circ\circ\circ}$	.6203$^{\circ\circ\circ}$	.7036
			.05	.5900$^{\bullet\bullet\bullet}$	.5275$^{\circ\circ\circ}$	.5248$^{\circ\circ\circ}$	.5231$^{\circ\circ\circ}$	.4426$^{\circ\circ\circ}$	.4653$^{\circ\circ\circ}$	.5629
			.01	.2924$^{\bullet\bullet\bullet}$	.2345$^{\circ\circ\circ}$	.2334$^{\circ\circ\circ}$	.2314$^{\circ\circ\circ}$	.1379$^{\circ\circ\circ}$	.1923$^{\circ\circ\circ}$	.2724
	10	β_2	.10	.1005	.1052$^{\bullet}$	.1022	.1028	.0963	.1000	.1050$^{\bullet}$
			.05	.0516	.0552$^{\bullet\bullet}$	.0508	.0519	.0497	.0482	.0551$^{\bullet\bullet}$
			.01	.0097	.0098	.0093	.0090	.0105	.0114	.0126$^{\bullet\bullet\bullet}$
		β_3	.10	.3910$^{\circ\circ\circ}$	.3668$^{\circ\circ\circ}$	.3698$^{\circ\circ\circ}$	.3691$^{\circ\circ\circ}$	.3140$^{\circ\circ\circ}$	.3952$^{\circ\circ\circ}$	.4639
			.05	.2551$^{\circ\circ\circ}$	.2401$^{\circ\circ\circ}$	.2400$^{\circ\circ\circ}$	.2381$^{\circ\circ\circ}$	.2073$^{\circ\circ\circ}$	.2591$^{\circ\circ\circ}$	.3267
			.01	.0823$^{\circ\circ\circ}$	.0744$^{\circ\circ\circ}$	.0765$^{\circ\circ\circ}$	.0738$^{\circ\circ\circ}$	.0486$^{\circ\circ\circ}$	.0802$^{\circ\circ\circ}$	.1215
		β_4	.10	.7758$^{\bullet\bullet\bullet}$	.7169$^{\circ\circ\circ}$	.7222$^{\circ\circ\circ}$	.7209$^{\circ\circ\circ}$	.5933$^{\circ\circ\circ}$	.6727$^{\circ\circ\circ}$	.7588
			.05	.6375$^{\bullet}$	.5746$^{\circ\circ\circ}$	.5795$^{\circ\circ\circ}$	.5763$^{\circ\circ\circ}$	.4721$^{\circ\circ\circ}$	.5284$^{\circ\circ\circ}$	.6259
			.01	.3472	.2760$^{\circ\circ\circ}$	.2870$^{\circ\circ\circ}$	.2823$^{\circ\circ\circ}$	.1306$^{\circ\circ\circ}$	.2358$^{\circ\circ\circ}$	.3388
	11	β_2	.10	.1033	.1015	.0989	.0991	.1003	.0996	.1039
			.05	.0501	.0488	.0497	.0490	.0475	.0498	.0522
			.01	.0090	.0092	.0081°	.0080$^{\circ\circ}$	.0076$^{\circ\circ}$	.0119$^{\bullet}$	.0120$^{\bullet\bullet}$
		β_3	.10	.4135$^{\circ\circ\circ}$	.3911$^{\circ\circ\circ}$	.3920$^{\circ\circ\circ}$	.3952$^{\circ\circ\circ}$	.3231$^{\circ\circ\circ}$	.4396$^{\circ\circ\circ}$	.5091
			.05	.2751$^{\circ\circ\circ}$	.2556$^{\circ\circ\circ}$	.2603$^{\circ\circ\circ}$	.2591$^{\circ\circ\circ}$	.2155$^{\circ\circ\circ}$	.2970$^{\circ\circ\circ}$	.3611
			.01	.0939$^{\circ\circ\circ}$	.0880$^{\circ\circ\circ}$	.0901$^{\circ\circ\circ}$	.0883$^{\circ\circ\circ}$	.0549$^{\circ\circ\circ}$	.0996$^{\circ\circ\circ}$	.1425
		β_4	.10	.8063	.7569$^{\circ\circ\circ}$	.7676$^{\circ\circ\circ}$	.7653$^{\circ\circ\circ}$	.6109$^{\circ\circ\circ}$	.7286$^{\circ\circ\circ}$	.8006
			.05	.6865	.6219$^{\circ\circ\circ}$	.6375$^{\circ\circ\circ}$	.6316$^{\circ\circ\circ}$	.4989$^{\circ\circ\circ}$	.5855$^{\circ\circ\circ}$	.6821
			.01	.4016	.3239$^{\circ\circ\circ}$	.3353$^{\circ\circ\circ}$	.3293$^{\circ\circ\circ}$	.1724$^{\circ\circ\circ}$	.2956$^{\circ\circ\circ}$	.4055

Table 8.2: Empirical Sizes and Sample Powers —< Continued >—

	n	β	α	f	w	ns	ls	cs	aw	t
(D)	9	β_2	.10	.0996	.1050$^{\bullet}$	.1018	.1006	.1018	.0996	.0972
			.05	.0484	.0548$^{\bullet\bullet}$	.0503	.0502	.0504	.0521	.0498
			.01	.0092	.0105	.0097	.0095	.0095	.0099	.0086
		β_3	.10	.3771$^{\circ\circ\circ}$	.3590$^{\circ\circ\circ}$	.3509$^{\circ\circ\circ}$	.3525$^{\circ\circ\circ}$	.3059$^{\circ\circ\circ}$	.3731$^{\circ\circ\circ}$	.4450
			.05	.2485$^{\circ\circ\circ}$	.2297$^{\circ\circ\circ}$	.2219$^{\circ\circ\circ}$	.2217$^{\circ\circ\circ}$	.1967$^{\circ\circ\circ}$	.2429$^{\circ\circ\circ}$	.3020
			.01	.0752$^{\circ\circ\circ}$	.0695$^{\circ\circ\circ}$	.0680$^{\circ\circ\circ}$	.0657$^{\circ\circ\circ}$	.0511$^{\circ\circ\circ}$	.0735$^{\circ\circ\circ}$	.1043
		β_4	.10	.7314$^{\bullet\bullet\bullet}$	.6709$^{\circ\circ\circ}$	.6666$^{\circ\circ\circ}$	.6681$^{\circ\circ\circ}$	.5929$^{\circ\circ\circ}$	.6121$^{\circ\circ\circ}$	.7001
			.05	.5957$^{\bullet\bullet\bullet}$	.5187$^{\circ\circ\circ}$	.5232$^{\circ\circ\circ}$	.5183$^{\circ\circ\circ}$	.4589$^{\circ\circ\circ}$	.4609$^{\circ\circ\circ}$	.5615
			.01	.3096$^{\bullet\bullet\bullet}$	.2394$^{\circ\circ\circ}$	.2473$^{\circ\circ\circ}$	.2486$^{\circ\circ\circ}$	.1569$^{\circ\circ\circ}$	.2043$^{\circ\circ\circ}$	.2855
	10	β_2	.10	.1012	.1031	.1028	.1023	.0980	.1004	.1019
			.05	.0508	.0537$^{\bullet}$	.0514	.0515	.0510	.0482	.0521
			.01	.0094	.0097	.0096	.0091	.0105	.0092	.0103
		β_3	.10	.4070$^{\circ\circ\circ}$	.3768$^{\circ\circ\circ}$	.3816$^{\circ\circ\circ}$	.3794$^{\circ\circ\circ}$	.3260$^{\circ\circ\circ}$	.4037$^{\circ\circ\circ}$	.4791
			.05	.2713$^{\circ\circ\circ}$	.2453$^{\circ\circ\circ}$	.2447$^{\circ\circ\circ}$	.2448$^{\circ\circ\circ}$	.2171$^{\circ\circ\circ}$	.2665$^{\circ\circ\circ}$	.3397
			.01	.0907$^{\circ\circ\circ}$	.0753$^{\circ\circ\circ}$	.0777$^{\circ\circ\circ}$	.0797$^{\circ\circ\circ}$	.0526$^{\circ\circ\circ}$	.0870$^{\circ\circ\circ}$	.1340
		β_4	.10	.7684$^{\bullet\bullet\bullet}$	.6997$^{\circ\circ\circ}$	.7109$^{\circ\circ\circ}$	.7120$^{\circ\circ\circ}$	.6140$^{\circ\circ\circ}$	.6615$^{\circ\circ\circ}$	.7511
			.05	.6437$^{\bullet}$	.5642$^{\circ\circ\circ}$	.5709$^{\circ\circ\circ}$	.5711$^{\circ\circ\circ}$	.4891$^{\circ\circ\circ}$	.5194$^{\circ\circ\circ}$	.6306
			.01	.3632$^{\bullet}$	.2733$^{\circ\circ\circ}$	.2943$^{\circ\circ\circ}$	.2887$^{\circ\circ\circ}$	.1541$^{\circ\circ\circ}$	.2448$^{\circ\circ\circ}$	.3504
	11	β_2	.10	.1026	.1002	.0995	.1007	.0994	.1006	.1023
			.05	.0506	.0493	.0490	.0480	.0481	.0484	.0480
			.01	.0096	.0085	.0080$^{\circ\circ}$	.0082°	.0083°	.0093	.0097
		β_3	.10	.4298$^{\circ\circ\circ}$	.3972$^{\circ\circ\circ}$	.4022$^{\circ\circ\circ}$	.4017$^{\circ\circ\circ}$	.3290$^{\circ\circ\circ}$	.4475$^{\circ\circ\circ}$	.5157
			.05	.2917$^{\circ\circ\circ}$	.2634$^{\circ\circ\circ}$	.2699$^{\circ\circ\circ}$	.2678$^{\circ\circ\circ}$	.2255$^{\circ\circ\circ}$	.3024$^{\circ\circ\circ}$	.3740
			.01	.1042$^{\circ\circ\circ}$	.0879$^{\circ\circ\circ}$	.0951$^{\circ\circ\circ}$	.0947$^{\circ\circ\circ}$	.0593$^{\circ\circ\circ}$	.1058$^{\circ\circ\circ}$	.1549
		β_4	.10	.8020$^{\bullet\bullet}$	.7417$^{\circ\circ\circ}$	.7543$^{\circ\circ\circ}$	.7552$^{\circ\circ\circ}$	.6379$^{\circ\circ\circ}$	.7132$^{\circ\circ\circ}$	.7901
			.05	.6878$^{\bullet}$	.6030$^{\circ\circ\circ}$	.6246$^{\circ\circ\circ}$	.6270$^{\circ\circ\circ}$	.5220$^{\circ\circ\circ}$	.5779$^{\circ\circ\circ}$	.6758
			.01	.4164	.3174$^{\circ\circ\circ}$	.3402$^{\circ\circ\circ}$	.3374$^{\circ\circ\circ}$	.1996$^{\circ\circ\circ}$	.2975$^{\circ\circ\circ}$	.4138
(L)	9	β_2	.10	.0988	.1051$^{\bullet}$	.1017	.1009	.1009	.0998	.0962
			.05	.0503	.0522	.0483	.0486	.0507	.0513	.0490
			.01	.0086	.0097	.0092	.0093	.0098	.0093	.0083°
		β_3	.10	.3805$^{\circ\circ\circ}$	.3581$^{\circ\circ\circ}$	.3470$^{\circ\circ\circ}$	.3482$^{\circ\circ\circ}$	.3020$^{\circ\circ\circ}$	.3696$^{\circ\circ\circ}$	.4325
			.05	.2414$^{\circ\circ\circ}$	.2232$^{\circ\circ\circ}$	.2191$^{\circ\circ\circ}$	.2187$^{\circ\circ\circ}$	.1932$^{\circ\circ\circ}$	.2405$^{\circ\circ\circ}$	.2857
			.01	.0721$^{\circ\circ\circ}$	.0692$^{\circ\circ\circ}$	.0661$^{\circ\circ\circ}$	.0635$^{\circ\circ\circ}$	.0488$^{\circ\circ\circ}$	.0697$^{\circ\circ\circ}$	.0893
		β_4	.10	.7467$^{\bullet\bullet\bullet}$	.6819$^{\circ\circ\circ}$	.6785$^{\circ\circ\circ}$	.6778$^{\circ\circ\circ}$	.5860$^{\circ\circ\circ}$	.6211$^{\circ\circ\circ}$	.7077
			.05	.6067$^{\bullet\bullet\bullet}$	.5321$^{\circ\circ\circ}$	.5299$^{\circ\circ\circ}$	.5256$^{\circ\circ\circ}$	.4518$^{\circ\circ\circ}$	.4622$^{\circ\circ\circ}$	.5583
			.01	.3017$^{\bullet\bullet\bullet}$	.2352$^{\circ\circ\circ}$	.2404$^{\circ\circ\circ}$	.2380$^{\circ\circ\circ}$	.1454$^{\circ\circ\circ}$	.1996$^{\circ\circ\circ}$	.2636
	10	β_2	.10	.1017	.1049	.1025	.1016	.1002	.1003	.1004
			.05	.0518	.0549$^{\bullet\bullet}$	.0512	.0514	.0515	.0499	.0511
			.01	.0084	.0096	.0090	.0088	.0101	.0089	.0099
		β_3	.10	.4035$^{\circ\circ\circ}$	.3731$^{\circ\circ\circ}$	.3753$^{\circ\circ\circ}$	.3747$^{\circ\circ\circ}$	.3241$^{\circ\circ\circ}$	.3979$^{\circ\circ\circ}$	.4690
			.05	.2665$^{\circ\circ\circ}$	.2436$^{\circ\circ\circ}$	.2429$^{\circ\circ\circ}$	.2423$^{\circ\circ\circ}$	.2125$^{\circ\circ\circ}$	.2616$^{\circ\circ\circ}$	.3242
			.01	.0865$^{\circ\circ\circ}$	.0735$^{\circ\circ\circ}$	.0776$^{\circ\circ\circ}$	.0772$^{\circ\circ\circ}$	.0484$^{\circ\circ\circ}$	.0835$^{\circ\circ\circ}$	.1139
		β_4	.10	.7869$^{\bullet\bullet\bullet}$	.7168$^{\circ\circ\circ}$	.7235$^{\circ\circ\circ}$	.7255$^{\circ\circ\circ}$	.6062$^{\circ\circ\circ}$	.6756$^{\circ\circ\circ}$	.7610
			.05	.6553$^{\bullet\bullet\bullet}$	.5797$^{\circ\circ\circ}$	.5827$^{\circ\circ\circ}$	.5822$^{\circ\circ\circ}$	.4826$^{\circ\circ\circ}$	.5286$^{\circ\circ\circ}$	.6326
			.01	.3600$^{\bullet\bullet\bullet}$	.2774$^{\circ\circ\circ}$	.2919$^{\circ\circ\circ}$	.2861$^{\circ\circ\circ}$	.1436$^{\circ\circ\circ}$	.2429$^{\circ\circ\circ}$	.3290
	11	β_2	.10	.1046	.1020	.0999	.1008	.1009	.1025	.1010
			.05	.0505	.0485	.0484	.0481	.0490	.0493	.0480
			.01	.0100	.0080$^{\circ\circ}$	.0083°	.0083°	.0083°	.0104	.0092
		β_3	.10	.4296$^{\circ\circ\circ}$	.3966$^{\circ\circ\circ}$	.3994$^{\circ\circ\circ}$	.3997$^{\circ\circ\circ}$	.3229$^{\circ\circ\circ}$	.4460$^{\circ\circ\circ}$	.5064
			.05	.2861$^{\circ\circ\circ}$	.2593$^{\circ\circ\circ}$	.2689$^{\circ\circ\circ}$	.2650$^{\circ\circ\circ}$	.2231$^{\circ\circ\circ}$	.2984$^{\circ\circ\circ}$	.3623
			.01	.1004$^{\circ\circ\circ}$	.0850$^{\circ\circ\circ}$	.0905$^{\circ\circ\circ}$	.0906$^{\circ\circ\circ}$	.0558$^{\circ\circ\circ}$	.1011$^{\circ\circ\circ}$	.1347
		β_4	.10	.8204$^{\bullet\bullet}$	.7607$^{\circ\circ\circ}$	.7711$^{\circ\circ\circ}$	.7725$^{\circ\circ\circ}$	.6262$^{\circ\circ\circ}$	.7311$^{\circ\circ\circ}$	.8070
			.05	.7029$^{\bullet\bullet\bullet}$	.6211$^{\circ\circ\circ}$	.6415$^{\circ\circ\circ}$	.6364$^{\circ\circ\circ}$	.5124$^{\circ\circ\circ}$	.5910$^{\circ\circ\circ}$	.6843
			.01	.4162$^{\bullet\bullet\bullet}$	.3206$^{\circ\circ\circ}$	.3405$^{\circ\circ\circ}$	.3360$^{\circ\circ\circ}$	.1802$^{\circ\circ\circ}$	.2985$^{\circ\circ\circ}$	.3961

Table 8.2: Empirical Sizes and Sample Powers —< Continued >—

	n	β	α	f	w	ns	ls	cs	aw	t
(N)	9	β_2	.10	.0983	.1049	.1009	.1005	.0973	.1014	.0964
			.05	.0507	.0549••	.0509	.0516	.0510	.0476	.0499
			.01	.0105	.0099	.0093	.0102	.0112	.0095	.0101
		β_3	.10	.3808°°°	.3554°°°	.3482°°°	.3466°°°	.3037°°°	.3583°°°	.4253
			.05	.2430°°°	.2261°°°	.2184°°°	.2157°°°	.1917°°°	.2211°°°	.2780
			.01	.0756°	.0637°°°	.0638°°°	.0635°°°	.0501°°°	.0636°°°	.0829
		β_4	.10	.7513•••	.6933°	.6873°°°	.6865°°°	.5785°°°	.6178°°°	.7044
			.05	.6088•••	.5344°°	.5287°°°	.5255°°°	.4443°°°	.4611°°°	.5503
			.01	.2974•••	.2329°°	.2358°°	.2307°°°	.1423°°°	.1848°°°	.2478
	10	β_2	.10	.0968	.0984	.0980	.0977	.0990	.1002	.0985
			.05	.0502	.0524	.0495	.0499	.0477	.0520	.0507
			.01	.0106	.0112	.0106	.0111	.0101	.0118•	.0107
		β_3	.10	.4071°°°	.3670°°°	.3708°°°	.3722°°°	.3131°°°	.3924°°°	.4604
			.05	.2671°°°	.2416°°°	.2397°°°	.2394°°°	.2076°°°	.2521°°°	.3081
			.01	.0828°°°	.0698°°°	.0724°°°	.0721°°°	.0476°°°	.0787°°°	.0999
		β_4	.10	.7863•••	.7260°°°	.7329°°°	.7298°°°	.5952°°°	.6815°°°	.7661
			.05	.6609•••	.5851°°°	.5888°°°	.5867°°°	.4724°°°	.5293°°°	.6238
			.01	.3547•••	.2738°°°	.2840°°°	.2832°°°	.1365°°°	.2371°°°	.3117
	11	β_2	.10	.0995	.1009	.1008	.1006	.0994	.1011	.0991
			.05	.0488	.0496	.0507	.0518	.0504	.0503	.0469
			.01	.0104	.0098	.0094	.0102	.0113	.0100	.0100
		β_3	.10	.4286°°°	.3921°°°	.3915°°°	.3885°°°	.3195°°°	.4389°°°	.5055
			.05	.2840°°°	.2503°°°	.2548°°°	.2553°°°	.2087°°°	.2850°°°	.3522
			.01	.0906°°°	.0805°°°	.0839°°°	.0816°°°	.0509°°°	.0936°°°	.1204
		β_4	.10	.8262•••	.7613°°°	.7711°°°	.7688°°°	.6053°°°	.7279°°°	.8119
			.05	.7097•••	.6197°°°	.6355°°°	.6328°°°	.4901°°°	.5926°°°	.6876
			.01	.4101•••	.3252°°°	.3339°°°	.3264°°°	.1694°°°	.2954°°°	.3800
(T)	9	β_2	.10	.0980	.1056•	.1014	.1017	.1019	.1031	.0991
			.05	.0507	.0532	.0482	.0484	.0490	.0499	.0483
			.01	.0099	.0106	.0105	.0100	.0095	.0103	.0103
		β_3	.10	.3808°°°	.3586°°°	.3530°°°	.3528°°°	.3173°°°	.3720°°°	.4480
			.05	.2518°°°	.2310°°°	.2229°°°	.2218°°°	.2032°°°	.2352°°°	.3119
			.01	.0803°°°	.0701°°°	.0667°°°	.0671°°°	.0523°°°	.0708°°°	.1174
		β_4	.10	.7102•••	.6691°°°	.6697°°°	.6686°°°	.5893°°°	.6142°°°	.6922
			.05	.5885•••	.5250°°°	.5237°°°	.5229°°°	.4596°°°	.4589°°°	.5662
			.01	.3087	.2356°°°	.2500°°°	.2494°°°	.1588°°°	.1964°°°	.3028
	10	β_2	.10	.0946°	.0966	.0931°°	.0927°°	.0972	.0977	.0975
			.05	.0452°°	.0499	.0471	.0466	.0472	.0495	.0491
			.01	.0098	.0091	.0085	.0087	.0090	.0097	.0094
		β_3	.10	.4105°°°	.3787°°°	.3830°°°	.3822°°°	.3381°°°	.4053°°°	.4849
			.05	.2707°°°	.2478°°°	.2479°°°	.2487°°°	.2258°°°	.2700°°°	.3504
			.01	.0948°°°	.0794°°°	.0831°°°	.0822°°°	.0561°°°	.0894°°°	.1456
		β_4	.10	.7470	.6986°°°	.7097°°°	.7085°°°	.6068°°°	.6659°°°	.7376
			.05	.6259	.5640°°°	.5738°°°	.5721°°°	.4870°°°	.5189°°°	.6220
			.01	.3561	.2735°°°	.2913°°°	.2913°°°	.1571°°°	.2436°°°	.3614
	11	β_2	.10	.0935°°	.1013	.0973	.0964	.0958	.0977	.0957
			.05	.0474	.0492	.0474	.0467	.0464°	.0488	.0488
			.01	.0096	.0089	.0084	.0086	.0080°°	.0096	.0110
		β_3	.10	.4360°°°	.4043°°°	.4061°°°	.4043°°°	.3407°°°	.4366°°°	.5212
			.05	.2958°°°	.2652°°°	.2681°°°	.2684°°°	.2311°°°	.3005°°°	.3845
			.01	.1074°°°	.0825°°°	.0890°°°	.0908°°°	.0643°°°	.1063°°°	.1708
		β_4	.10	.7731	.7346°°°	.7409°°°	.7409°°°	.6210°°°	.7102°°°	.7771
			.05	.6613	.5950°°°	.6154°°°	.6156°°°	.5122°°°	.5755°°°	.6708
			.01	.4021°°°	.3188°°°	.3436°°°	.3406°°°	.1959°°°	.2952°°°	.4216

Table 8.2: Empirical Sizes and Sample Powers —< Continued >—

	n	β	α	f	w	ns	ls	cs	aw	t
(U)	9	β_2	.10	.0998	.1026	.0964	.0957	.0997	.0970	.0953
			.05	.0495	.0527	.0492	.0488	.0501	.0482	.0477
			.01	.0081°	.0093	.0099	.0103	.0100	.0102	.0095
		β_3	.10	.3701°°°	.3477°°°	.3380°°°	.3363°°°	.2857°°°	.3504°°°	.4092
			.05	.2319°°°	.2202°°°	.2067°°°	.2050°°°	.1782°°°	.2210°°°	.2614
			.01	.0660	.0634°°	.0591°°°	.0574°°°	.0416°°°	.0629°°	.0717
		β_4	.10	.7704•••	.7127	.7004°°°	.6952°°°	.5525°°°	.6328°°°	.7199
			.05	.6184•••	.5462	.5277°°°	.5202°°°	.4177°°°	.4604°°°	.5490
			.01	.2749•••	.2232	.2090	.2034°°	.1146°°°	.1740°°°	.2153
	10	β_2	.10	.1027	.1043	.1039	.1042	.1015	.1000	.1025
			.05	.0520	.0560•••	.0529	.0527	.0532	.0508	.0513
			.01	.0082°	.0092	.0098	.0100	.0102	.0111	.0107
		β_3	.10	.3974°°°	.3686°°°	.3663°°°	.3646°°°	.2948°°°	.3837°°°	.4502
			.05	.2586°°°	.2386°°°	.2321°°°	.2276°°°	.1918°°°	.2431°°°	.2891
			.01	.0755°°°	.0681°°°	.0667°°°	.0647°°°	.0424°°°	.0760°°°	.0867
		β_4	.10	.8148•••	.7468°°°	.7491°°°	.7422°°°	.5624°°°	.6943°°°	.7827
			.05	.6737•••	.6000°°°	.5929°°°	.5859°°°	.4434°°°	.5348°°°	.6310
			.01	.3352•••	.2631°°°	.2575°°°	.2483°°°	.1010°°°	.2218°°°	.2839
	11	β_2	.10	.0994	.1009	.1018	.1012	.1021	.1021	.0988
			.05	.0504	.0510	.0491	.0480	.0487	.0488	.0500
			.01	.0096	.0084	.0081°	.0078°°	.0083°	.0121••	.0102
		β_3	.10	.4234°°°	.3925°°°	.3867°°°	.3834°°°	.3023°°°	.4287°°°	.4906
			.05	.2794°°°	.2518°°°	.2506°°°	.2467°°°	.2005°°°	.2830°°°	.3282
			.01	.0895°°°	.0789°°°	.0805°°°	.0789°°°	.0454°°°	.0886°°°	.1090
		β_4	.10	.8532•••	.7925°°°	.7986°°°	.7939°°°	.5636°°°	.7562°°°	.8341
			.05	.7353•••	.6490°°°	.6495°°°	.6412°°°	.4567°°°	.6057°°°	.7002
			.01	.3930•••	.3171°°°	.3115°°°	.2997°°°	.1332°°°	.2874°°°	.3592
(Q)	9	β_2	.10	.0993	.1030	.0983	.0980	.0998	.0991	.0954
			.05	.0486	.0522	.0498	.0496	.0506	.0484	.0476
			.01	.0076°°	.0106	.0104	.0104	.0097	.0102	.0086
		β_3	.10	.3715°°°	.3501°°°	.3428°°°	.3435°°°	.2959°°°	.3546°°°	.4123
			.05	.2348°°°	.2201°°°	.2099°°°	.2088°°°	.1829°°°	.2274°°°	.2632
			.01	.0660°	.0659°	.0611°°°	.0598°°°	.0432°°°	.0655°°	.0727
		β_4	.10	.7668•••	.7090°	.7010°°°	.6962°°°	.5683°°°	.6326°°°	.7199
			.05	.6180•••	.5440	.5310°°°	.5275°°°	.4324°°°	.4668°°°	.5509
			.01	.2808•••	.2309	.2233	.2196	.1232°°°	.1864°°°	.2252
	10	β_2	.10	.1022	.1034	.1022	.1027	.1006	.1002	.1019
			.05	.0513	.0553••	.0533	.0537•	.0505	.0501	.0515
			.01	.0089	.0097	.0089	.0093	.0103	.0100	.0088
		β_3	.10	.3978°°°	.3683°°°	.3723°°°	.3692°°°	.3049°°°	.3877°°°	.4513
			.05	.2610°°°	.2384°°°	.2361°°°	.2335°°°	.1987°°°	.2485°°°	.2979
			.01	.0774°°°	.0714°°°	.0711°°°	.0683°°°	.0447°°°	.0778°°°	.0931
		β_4	.10	.8088•••	.7423°°°	.7482°°°	.7449°°°	.5836°°°	.6941°°°	.7758
			.05	.6741•••	.5996°°°	.5950°°°	.5890°°°	.4603°°°	.5391°°°	.6316
			.01	.3457•••	.2698°°°	.2715°°°	.2632°°°	.1142°°°	.2302°°°	.2918
	11	β_2	.10	.1012	.1022	.0984	.0992	.1021	.1015	.1000
			.05	.0499	.0487	.0485	.0477	.0491	.0481	.0486
			.01	.0102	.0076°°	.0092	.0086	.0087	.0107	.0100
		β_3	.10	.4230°°°	.3942°°°	.3912°°°	.3889°°°	.3111°°°	.4347°°°	.4956
			.05	.2816°°°	.2543°°°	.2593°°°	.2562°°°	.2093°°°	.2896°°°	.3369
			.01	.0908°°°	.0841°°°	.0832°°°	.0804°°°	.0504°°°	.0919°°°	.1122
		β_4	.10	.8485•••	.7881°°°	.7951°°°	.7946°°°	.5887°°°	.7521°°°	.8281
			.05	.7286•••	.6470°°°	.6573°°°	.6509°°°	.4783°°°	.6086°°°	.6953
			.01	.4016•••	.3204°°°	.3271°°°	.3167°°°	.1502°°°	.2923°°°	.3698

8.2, where $\alpha = 0.10,\ 0.05,\ 0.01$ is examined. Theoretically each value in the table should be equivalent to the probability which rejects the null hypothesis $H_0 : \beta_j = 0$ against the alternative hypothesis $H_1 : \beta_j > 0$ for $j = 2, 3, 4$. In Table 8.2, the probability which rejects $H_0 : \beta_2 = 0$ corresponds to the significance level α and the probabilities which reject $H_0 : \beta_j = 0$ for $j = 3, 4$ indicates the sample powers, because the true parameter values are given by $(\beta_2, \beta_3, \beta_4) = (0.0,\ 0.5,\ 0.9)$

In Table 8.2, the superscripts $^{\bullet}$, $^{\bullet\bullet}$ and $^{\bullet\bullet\bullet}$ in β_2 imply that the empirical size is statistically larger than α by the one-sided test of the significance levels 10%, 5% and 1% (i.e., $\alpha = 0.10,\ 0.05,\ 0.01$), respectively. The superscripts $^{\circ}$, $^{\circ\circ}$ and $^{\circ\circ\circ}$ in β_2 indicate that the empirical size is statistically smaller than α by the one-sided test of the significance levels 10%, 5% and 1%, respectively. That is, each value without the superscript $^{\bullet}$ or $^{\circ}$ in β_2 gives us the correct size. $^{\bullet}$ and $^{\circ}$ in β_3 and β_4 represent a comparison with t. The value with $^{\bullet}$ implies that the sample power of the corresponding test is significantly greater than that of t (i.e., the corresponding test is more powerful than t), and the value with $^{\circ}$ indicates that the sample power of the corresponding test is significantly less than that of t (i.e., the corresponding test is less powerful than t).

As in Section 8.3.1, in the case of (N), the t test should be better than any other nonparametric tests. In other words, t in β_2 should be closer to the significance level α than any other nonparametric tests, and t in β_3 and β_4 has to be larger than any other tests, because the OLS estimator of β_j follows the t distribution with $n - k$ degrees of freedom when the error terms ϵ_i, $i = 1, 2, \cdots, n$, are mutually independently and normally distributed with mean zero.

The results are as follows. β_2 in Table 8.2 indicates the empirical size, because the true value of β_2 is given by zero. For the t tests of (C) and (X), the empirical sizes in β_2 are over-estimated for all $\alpha = 0.10,\ 0.05,\ 0.01$. However, for the nonparametric tests f, w, ns, ls, cs and aw, almost all the values in Table 8.2 are very close to the significance level α. Therefore, we can conclude that the nonparametric tests is superior to the t test in the sense of the size criterion. β_3 and β_4 in Table 8.2 represent the sample powers, because $\beta_3 = 0.5$ and $\beta_4 = 0.9$ are the true values. For β_3, in all the cases except for (U), $n = 9$, $\alpha = 0.01$ and f, t is more powerful than the nonparametric tests, because except for only one case all the values have $^{\circ}$ in the superscript. In the cases of (B), (L), (N), (U) and (Q) in β_4, clearly f is superior to t, although t is better than f in the cases of (C) and (X) in β_4. In the other cases such as (G), (D) and (T), f is better than t for small n and large α. Thus, we can observe through Table 8.2 that except for a few cases the permutation test f is more powerful than the t test. In addition, each difference between f and t becomes small as n is large. Therefore, the permutation test f is close to the t test as the sample size is large.

Next, graphically we compare the sample powers. Taking the case of $n = 10$, $k = 4$ and $\alpha = 0.10$, Figure 8.2 represents the sample powers for $\beta_i = 0$, $i = 1, 2, 3$, and $\beta_4 = 0.0, 0.1, \cdots, 0.9$. For each population distribution, the features of the nonparametric and parametric tests are summarized as follows:

(B) There are five lines to be distinguished. The first line from the top is f, the

Figure 8.2: Sample Powers: $n = 10$, $k = 4$, $\alpha = 0.10$ and $\beta_i = 0$ for $i = 1, 2, 3$

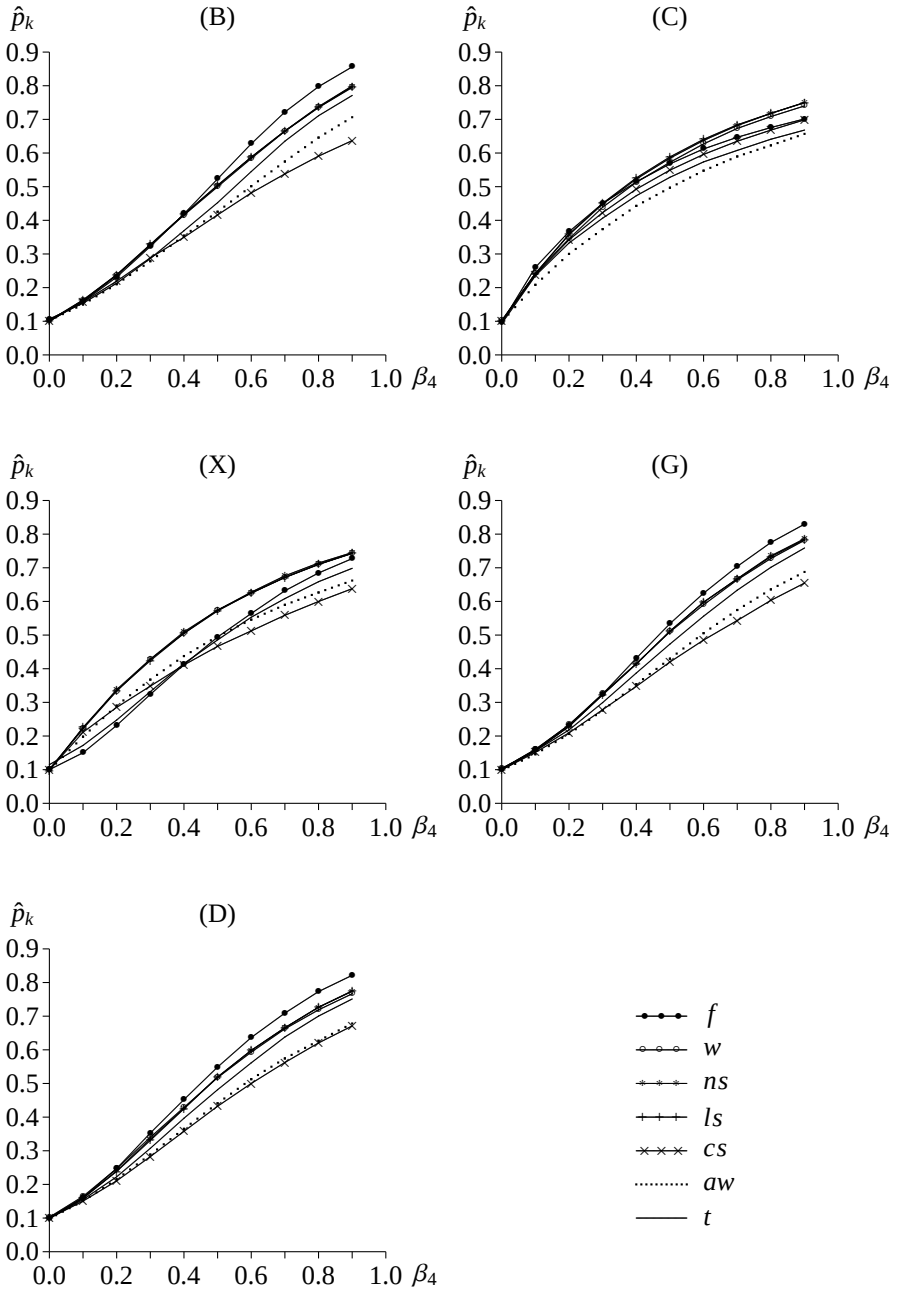
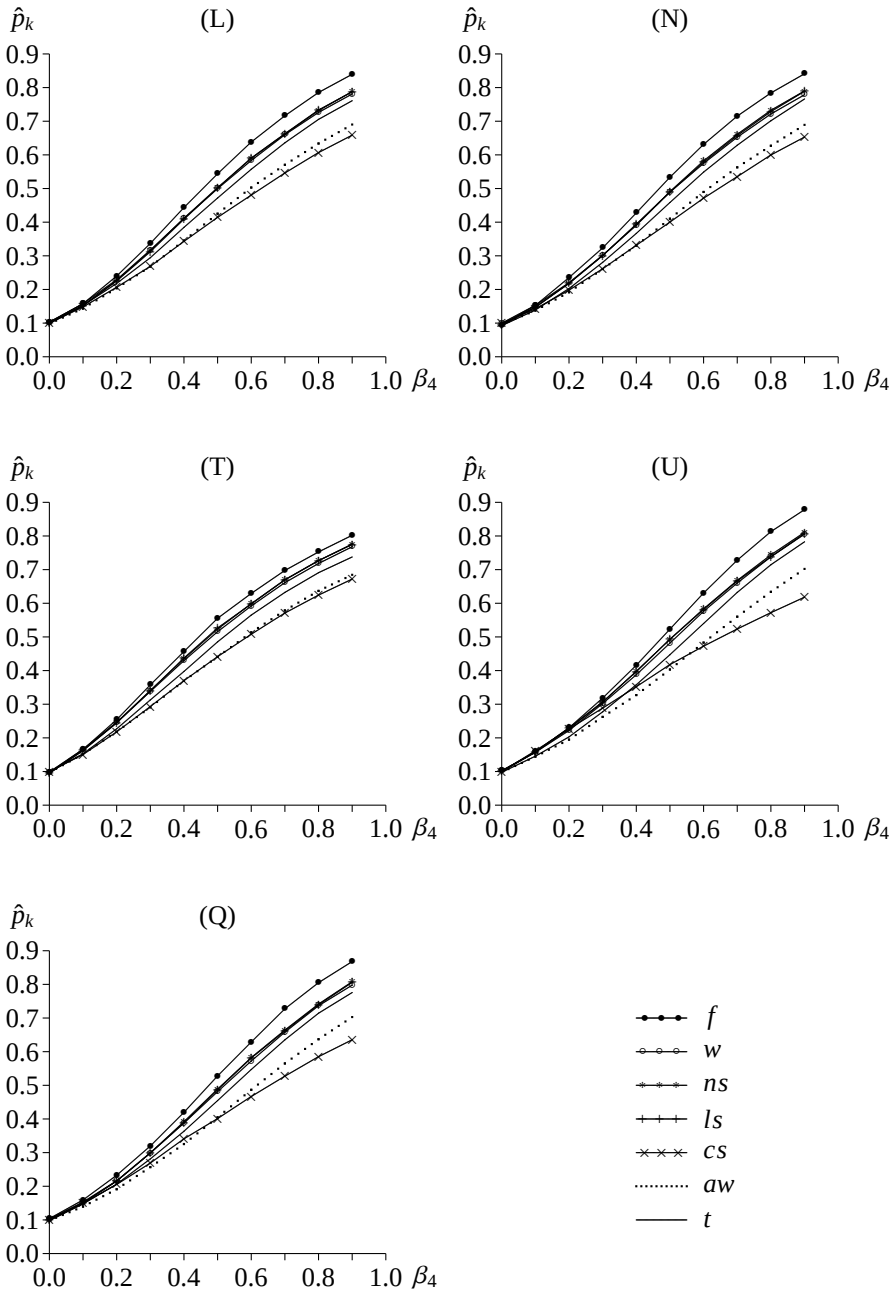


Figure 8.2: Sample Powers: $n = 10$, $k = 4$, $\alpha = 0.10$ and $\beta_i = 0$ for $i = 1, 2, 3$
—< Continued >—



second line consists of w, ns and ls, the third line is given by t, the next line corresponds to aw, and the bottom line is cs.

(C) We have six lines around $\beta_4 = 0.6, 0.7, 0.8$, which are (i) ns and ls, (ii) w, (iii) f, (iv) cs, (v) t, and (vi) aw from the top,

(X) The first line from the top is given by w, ns and ls, which are overlapped. f gives us the worst test for small β_4, but cs is the worst for β_4 greater than 0.4.

(G) We have the same results as (B). That is, we can see five lines. The first line from the top is f, the second line is given by w, ns and ls, the third line is t, the next line is aw, and the bottom line is cs.

(D) f indicates the best test. cs is the worst test. aw is slightly better than cs.

(L) This case is also similar to (B) and (G).

(N) This is also similar to (B), (G) and (L).

(T) aw is very close to cs. Both are inferior to the other tests. f is the best test.

(U) For small β_4, aw is the worst and t is the second worst. cs is not so bad for small β_4, but it is the worst for large β_4.

(Q) Except for small β_4, we obtain the same results as (N), (B), (G) and (L).

Thus, in all the cases except for (C) and (X), f shows the best performance. cs is the worst test at least when β_4 is large. w, ns and ls are between f and t in a lot of cases. aw is as poor as cs, because aw is approximated as a normal distribution, which is a large sample property and does not hold in small sample.

CPU Time: As mentioned above, in the case where we perform the significance test of the regression coefficient, we need to compute the $n!$ regression coefficients (for example, $n!$ is equal to about 3.6 million when $n = 10$). In Table 8.3, CPU time per simulation run is shown, where the arithmetic average from 10^3 simulation runs is computed. Table 8.3 indicates the CPU time required to obtain all the tests (i.e., f, w, ns, ls, cs, aw and t) in each case of $n = 10, 11, 12$ and $k = 2, 3, 4$, where $\beta_i = 0$ is taken for all $i = 1, 2, \cdots, k$. Pentium III 1GHz Dual CPU personal computer, Windows 2000 SP2 operating system and Open Watcom C/C++32 Optimizing Compiler (Version 1.0) and are utilized. The order of computation is about $n! \times (k-1)$, because the constant term is included in the regression model which we consider in this chapter. Note that the order of computation is $n! \times k$ if the constant term is not included in the regression model (see Remark 7). The case of sample size n is n times more computer-intensive than that of sample size $n-1$. For example, it might be expected that the case of $n = 15$ and $k = 4$ takes about 25.7 days (i.e., $15 \times 14 \times 13 \times 13.549$ minutes) to obtain the result, which is not feasible in practice. Thus, the permutation test discussed in this chapter is very computer-intensive.

Therefore, we need to consider less computational procedure. In order to reduce computational burden when $n!$ is large, it might be practical to choose some of the $n!$ permutations randomly and perform the same testing procedure discussed in this

Table 8.3: CPU Time (minutes)

$n \setminus k$	2	3	4
11	0.047	0.070	0.092
12	0.552	0.824	1.096
13	6.970	10.262	13.549

chapter. That is, as shown in Remark 9, taking M permutations out of the $n!$ permutations randomly, we compute the probabilities $P(\hat{\beta}_j < \hat{\beta}_j^{(0)})$ and $P(\hat{\beta}_j > \hat{\beta}_j^{(0)})$, where $\hat{\beta}_j^{(0)}$ denotes the estimate of the jth regression coefficient obtained from the original data. If either of them is smaller than $\alpha = 0.1$, the null hypothesis $H_0 : \beta_j = 0$ is rejected. In Table 7.3 on p.424, we have examined whether the empirical sizes depend on M, taking an example of testing difference between two-sample means, where the case of $M = 10^6$ is very close to the case of all the possible combinations. Since we consider the same nonparametric tests, clearly it can be expected to obtain the same results.

8.4 Empirical Example

In Remark 8 of Section 8.2.2, we have shown how to construct the confidence interval for each parameter. In Remark 9, we have discussed how to reduce computational burden when the sample size is large. In this section, we show an empirical example as an application, taking the same example as in Chapter 7 (Section 7.5). Annual data from *Annual Report on National Accounts* (Economic and Social Research Institute, Cabinet Office, Government of Japan) is used. Let GDP_t be Gross Domestic Product (1990 price, billions of Japanese yen), M_t be Imports of Goods and Services (1990 price, billions of Japanese yen), and P_t be Terms of Trade Index, which is given by Gross Domestic Product Implicit Price Deflator divided by Imports of Goods and Services Implicit Price Deflator. In Chapter 7, we have estimated two import functions (7.19) and (7.20), which are as follows:

$$\log M_t = \beta_1 + \beta_2 \log GDP_t + \beta_3 \log P_t, \tag{7.19}$$

$$\log M_t = \beta_1 + \beta_2 \log GDP_t + \beta_3 \log P_t + \beta_4 \log M_{t-1}, \tag{7.20}$$

where β_1, β_2, β_3 and β_4 are the unknown parameters to be estimated. Because we have $n = 46$ for equation (7.19) and $n = 45$ for equation (7.20), it is impossible to obtain $n!$ permutations for both of the cases. Therefore, we take $M = 10^6$ permutations, instead of $M = n!$ permutations.

As shown in Remark 8, by considering the permutation between $(X'X)^{-1}X_i'$ and e_i, $i = 1, 2, \cdots, n$, we can obtain M regression coefficients, compute the arithmetic average (AVE), standard error (SER), skewness (Skewness) and kurtosis (Kurtosis) based on the M regression coefficients and obtain the percent points (0.010, 0.025, 0.050, 0.100, 0.250, 0.500, 0.750, 0.900, 0.950, 0.975 and 0.990) to construct the confidence

Table 8.4: $t(n-k)$ versus Permutation

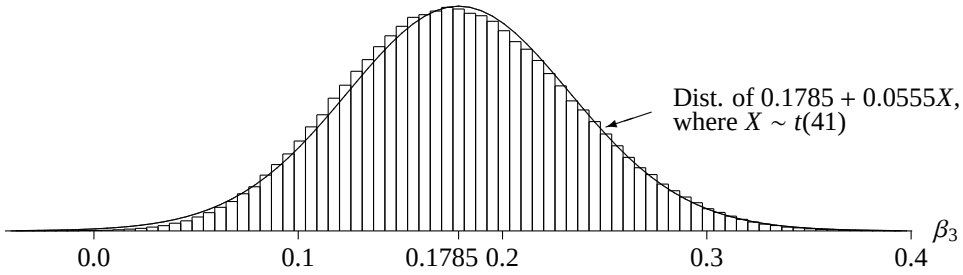
— Equation (7.19) —

	$t(n-k)$ for $n=46$ and $k=3$			Permutation ($N=10^6$)		
	β_1	β_2	β_3	β_1	β_2	β_3
OLSE	−6.0123	1.2841	0.1970			
AVE				−6.0127	1.2841	0.1969
SER	0.5137	0.0403	0.0779	0.5020	0.0394	0.0762
Skewness	0.0000	0.0000	0.0000	0.0141	−0.0136	0.0138
Kurtosis	3.1622	3.1622	3.1622	2.8773	2.8771	2.8754
0.010	−7.2559	1.1864	0.0083	−7.1627	1.1931	0.0228
0.025	−7.0498	1.2026	0.0396	−6.9897	1.2069	0.0487
0.050	−6.8768	1.2162	0.0658	−6.8369	1.2189	0.0716
0.100	−6.6814	1.2315	0.0955	−6.6605	1.2331	0.0987
0.250	−6.3619	1.2566	0.1439	−6.3565	1.2572	0.1447
0.500	−6.0123	1.2841	0.1970	−6.0141	1.2842	0.1967
0.750	−5.6626	1.3115	0.2500	−5.6701	1.3111	0.2489
0.900	−5.3431	1.3366	0.2985	−5.3631	1.3350	0.2955
0.950	−5.1477	1.3520	0.3281	−5.1830	1.3489	0.3228
0.975	−4.9747	1.3656	0.3544	−5.0306	1.3609	0.3461
0.990	−4.7686	1.3817	0.3857	−4.8536	1.3745	0.3729

— Equation (7.20) —

	$t(n-k)$ for $n=45$ and $k=4$				Permutation ($N=10^6$)			
	β_1	β_2	β_3	β_4	β_1	β_2	β_3	β_4
OLSE	−1.2387	0.4111	0.1785	0.6175				
AVE					−1.2390	0.4112	0.1785	0.6175
SER	0.8169	0.1376	0.0555	0.0958	0.7892	0.1330	0.0536	0.0926
Skewness	0.0000	0.0000	0.0000	0.0000	0.0495	−0.0164	0.1190	−0.0076
Kurtosis	3.1538	3.1538	3.1538	3.1538	2.8620	2.8614	2.8675	2.8632
0.010	−3.2126	0.0786	0.0444	0.3859	−3.0157	0.1051	0.0605	0.4047
0.025	−2.8862	0.1336	0.0665	0.4242	−2.7588	0.1510	0.0773	0.4367
0.050	−2.6120	0.1798	0.0852	0.4564	−2.5288	0.1914	0.0923	0.4647
0.100	−2.3020	0.2320	0.1062	0.4928	−2.2538	0.2389	0.1102	0.4977
0.250	−1.7944	0.3175	0.1407	0.5523	−1.7818	0.3206	0.1412	0.5541
0.500	−1.2387	0.4111	0.1785	0.6175	−1.2465	0.4116	0.1772	0.6177
0.750	−0.6829	0.5048	0.2163	0.6827	−0.7034	0.5024	0.2146	0.6809
0.900	−0.1754	0.5903	0.2507	0.7423	−0.2127	0.5827	0.2487	0.7370
0.950	0.1347	0.6425	0.2718	0.7786	0.0754	0.6297	0.2691	0.7699
0.975	0.4089	0.6887	0.2904	0.8108	0.3179	0.6693	0.2865	0.7974
0.990	0.7353	0.7437	0.3126	0.8491	0.6008	0.7143	0.3060	0.8288

Figure 8.3: Empirical Distribution for $\hat{\beta}_3$ in Equation (7.20)



intervals. The results are in Table 8.4. In Table 8.4, the right hand side corresponds to the results of the permutation test, denoted by f in the previous sections, while the left hand side represents the conventional OLS results, denoted by t in the previous sections. Therefore, the left hand side is based on the normality assumption on the error term. Thus, in the left hand side, Skewness, Kurtosis and the percent points are obtained from the $t(n-k)$ distribution. Remember that in the case of the $t(n-k)$ distribution Skewness and Kurtosis are given by 0.0 and $3+6/(n-k-4)$. In the left hand side, OLSE indicates the OLS estimate of each parameter, which is independent of the underlying distribution on the error term. Moreover, note that OLSE and SER in the left hand side are the exactly same results as equations (7.19) and (7.20) on p.426.

OLSE in the left hand size is almost equal to AVE in the right hand side, because the permutation test is based on OLSE and the permutation between $(X'X)^{-1}X_i'$ and e_i, $i = 1, 2, \cdots, n$. As for Skewness, Kurtosis and the percent points, the $t(n-k)$ test is different from the permutation test. Comparing both tests with respect to SER, the permutation test is smaller than $t(n-k)$ test for all the parameters, i.e., β_i for $i = 1, 2, 3$ in equation (7.19) and β_i for $i = 1, 2, 3, 4$ in equation (7.20). Furthermore, the confidence interval of the permutation test has smaller range than that of the $t(n-k)$ test, because for example we take 0.025 and 0.975 in β_2 of equation (7.19), and obtain $1.3656 - 1.2026 = 0.163$ for the $t(n-k)$ test and $1.3609 - 1.2069 = 0.154$ for the permutation test. In addition, we can know the true distribution on the error term by using the permutation test, i.e., Skewness is almost zero for both tests and all the β_i, $i = 1, 2, 3$, in equation (7.19), but the permutation test has smaller Kurtosis than the t test for all the parameters. That is, the empirical distribution based on the permutation test is symmetric around the OLS estimate, but it has thinner tails than the $t(n-k)$ distribution.

In Figure 8.3, the empirical distribution based on the permutation is displayed for β_3 in equation (7.20), which is compared with the $t(n-k)$ distribution (note that $n-k = 45-4 = 41$). The solid line indicates the distribution of $0.1785 + 0.0555X$, where 0.1785 and 0.0555 represent the OLSE of β_3 and its standard error, and X denotes $X \sim t(n-k)$ for $n-k = 41$. The bar graph represents the empirical distribution based on the permutation. Skewness of β_3 in equation (7.20) is 0.1190, which implies that the empirical distribution is slightly skewed to the right. Thus, using the permutation test,

we can obtain a non-symmetric empirical distribution, which might be more plausible in practice. Remember that the t distribution is always symmetric.

8.5 Summary

Only when the error term is normally distributed, we can utilize the t test for testing the regression coefficient. Since the distribution of the error term is not known, we need to check whether the normality assumption is plausible before testing the hypothesis. As a result of testing, in the case where the normality assumption is rejected, we cannot test the hypothesis on the regression coefficient using the t test. In order to improve this problem, in this chapter we have shown the significance test on the regression coefficient, which can be applicable to any distribution.

In Section 8.3.1, We have tested whether the correlation coefficient between two samples is zero and examined the sample powers of the two tests. For each of the cases where the underlying samples are normal, chi-squared, uniform, logistic and Cauchy, 10^4 simulation runs are performed and the nonparametric tests are compared with the parametric t test with respect to the empirical sizes and the sample powers. As it is easily expected, the t test is sometimes a biased test under the non-Gaussian assumption. That is, we have the cases where the empirical sizes are over-estimated. However, the nonparametric permutation test, denoted by f, gives us the correct empirical sizes without depending on the underlying distribution. Specifically, even when the sample is normal, the nonparametric permutation test is very close to t test (theoretically, the t test should be better than any other test when the sample is normal).

In Section 8.3.2, we have performed the Monte Carlo experiments on the significance test of the regression coefficients. It might be concluded that the nonparametric permutation test is closer to the true size than the t test for almost all the cases. Moreover, the sample powers are compared for both tests. As a result, we find that the permutation test is more powerful than the t test. See Figure 8.2. Thus, we can find through the Monte Carlo experiments that the nonparametric test discussed in this chapter is applied to any distribution of the underlying sample.

However, the problem is that the nonparametric test is too computer-intensive. We have shown that when $n!$ is too large it is practical to choose some of the $n!$ permutations randomly and perform the testing procedure. An empirical example of $(n, k) = (46, 3), (45, 4)$ is shown in Section 8.4, where the permutation test f is compared with the conventional t test. There, we have performed the testing procedure taking the 10^6 permutations out of all the possible permutations ($(n-k)!$ in this case) randomly. Thus, it has been shown in this chapter that we can reduce the computational burden. From the empirical example, we have obtained the results that the permutation test f has smaller length in the confidence intervals than the t test.

Appendix 8.1: Permutation

In this appendix, the source code on permutation is shown. The recursion is required for permutation. Therefore, the source code is written in C language, which is shown as follows.

permutation(n)

```
1:    int  idx[101];
2: /* ---------------------------------------------- */
3: void permutation(int n)
4: {
5:    int  i;
6:    void perm(int i,int n);
7:
8:    for(i=1;i<=n;i++) idx[i]=i;
9:    i=1;
10:   perm(i,n);
11: }
12: /* ---------------------------------------------- */
13: void perm(int i,int n)
14: {
15:   int j,m;
16:
17:   if( i<n ){
18:      for(j=i;j<=n;j++){
19:          m=idx[i]; idx[i]=idx[j]; idx[j]=m;
20:          perm(i+1,n);
21:          m=idx[i]; idx[i]=idx[j]; idx[j]=m;
22:      }
23:   }
24:   else{
25:      for(i=1;i<=n;i++) printf("%3d",idx[i]);
26:      printf("\n");
27:   }
28: }
```

`idx[101]` is defined as an external variable in Line 1. In Line 8, initially `idx[i]=i` is input for `i=1,2,...,n`. In Line 25, `idx[i]`, `i=1,2,...,n`, are permuted and printed out on screen. An example of `n=4` is shown in Table 8.5, where all the possible permutations are printed out in order.

As `n` increases, the number of all the possible permutations, `n!`, becomes extremely large. Therefore, we sometimes need the source code which obtains some of all the permutations randomly to reduce computational burden. The following Fortran 77 source code obtains one permutation randomly out of `n!` permutations.

random_permutation(ix,iy,n,index)

```
1:       subroutine random_permutation(ix,iy,n,index)
2:       dimension index(100)
```

Table 8.5: Output by `permutation(n)`: Case of `n=4`

Output	
1 2 3 4	3 2 1 4
1 2 4 3	3 2 4 1
1 3 2 4	3 1 2 4
1 3 4 2	3 1 4 2
1 4 3 2	3 4 1 2
1 4 2 3	3 4 2 1
2 1 3 4	4 2 3 1
2 1 4 3	4 2 1 3
2 3 1 4	4 3 2 1
2 3 4 1	4 3 1 2
2 4 3 1	4 1 3 2
2 4 1 3	4 1 2 3

```
3:          do 1 i=1,n
4:      1 index(i)=i
5:          do 2 i=1,n-1
6:        call urnd(ix,iy,rn)
7:        j=int(rn*(n-i+1))+i
8:        idx=index(j)
9:        index(j)=index(i)
10:     2 index(i)=idx
11:       return
12:       end
```

In Line 10, one of the integers from 1 to n is randomly input into `index(i)`. Repeating `random_permutation(ix,iy,n,index)`, we can obtain as many permutations as we want.

As another source code which obtains one of the permutations randomly, the case of `n2` $= 0$ in `random_combination(n1,n2,num)` on p.439 corresponds to the **random permutation**, where the number of permutation is given by `num`. Note that `n1` in the function `random_combination(n1,n2,num)` on p.439 is equivalent to `n` in the subroutine `random_permutation(ix,iy,n,index)` in the case of `n2` $= 0$.

Appendix 8.2: Distribution of $\hat{\rho}$

Distribution of $\hat{\rho}$ under $\rho = 0$: We derive the distribution function of the sample correlation under $\rho = 0$. Assume that (X_i, Y_i), $i = 1, 2, \cdots, n$, are normally distributed, i.e.,

$$\begin{pmatrix} X_i \\ Y_i \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_X \\ \mu_Y \end{pmatrix}, \begin{pmatrix} \sigma_X^2 & \rho\sigma_X\sigma_Y \\ \rho\sigma_X\sigma_Y & \sigma_Y^2 \end{pmatrix}\right).$$

Define $E_i = (Y_i - \mu_Y) - \rho\dfrac{\sigma_Y}{\sigma_X}(X_i - \mu_X)$ for $i = 1, 2, \cdots, n$. Then, it is easily shown that E_1, E_2, $\cdots$, E_n are mutually independently and normally distributed with mean zero and variance $(1 - \rho^2)\sigma_Y^2$ and that E_i are independent of X_i for all $i = 1, 2, \cdots, n$.

Moreover, we can rewrite the above equation as $Y_i = \alpha + \beta X_i + E_i$, where $\beta = \rho\sigma_Y/\sigma_X$ and $\alpha = \mu_Y - \beta\mu_X$. Because X_i is independent of E_i, the conditional distribution of Y_i given $X_1 = x_1$, $X_2 = x_2$, $\cdots$, $X_n = x_n$ is normally distributed with mean $\alpha + \beta x_i$ and variance $(1-\rho^2)\sigma_Y^2$.

Now, define $\hat{\beta}$ as:

$$\hat{\beta} = \frac{\sum_{i=1}^n (x_i - \overline{x})(Y_i - \overline{Y})}{\sum_{i=1}^n (x_i - \overline{x})^2},$$

which corresponds to the OLS estimator of β given $X_i = x_i$, $i = 1, 2, \cdots, n$. Furthermore, we can easily show that the conditional distribution of $\hat{\beta}$ given $X_i = x_i$, $i = 1, 2, \cdots, n$, is normal with mean β and variance $(1-\rho^2)\sigma_Y^2/\sum_{i=1}^n (x_i - \overline{x})^2$. Define S^2 as:

$$\begin{aligned} S^2 &= \frac{1}{n-2}\sum_{i=1}^n (Y_i - \hat{\alpha} - \hat{\beta}x_i)^2 = \frac{1}{n-2}\sum_{i=1}^n \Big((Y_i - \overline{Y}) - \hat{\beta}(x_i - \overline{x})\Big)^2 \\ &= \frac{1}{n-2}\Big(\sum_{i=1}^n (Y_i - \overline{Y})^2 - \hat{\beta}^2 \sum_{i=1}^n (x_i - \overline{x})^2\Big), \end{aligned}$$

where $\hat{\alpha} = \overline{Y} - \hat{\beta}\overline{x}$ is substituted in the second equality. $\dfrac{(n-2)S^2}{(1-\rho^2)\sigma_Y^2}$ is distributed as a chi-square random variable with $n-2$ degrees of freedom. In addition, given $X_i = x_i$, $i = 1, 2, \cdots, n$, $\hat{\beta}$ is independent of S^2. Define T as:

$$T = \frac{\dfrac{\hat{\beta} - \beta}{\sqrt{(1-\rho^2)\sigma_Y^2/\sum_{i=1}^n (x_i - \overline{x})^2}}}{\sqrt{\dfrac{(n-2)S^2}{(1-\rho^2)\sigma_Y^2}\Big/(n-2)}} = \frac{\hat{\beta} - \beta}{\sqrt{S^2/\sum_{i=1}^n (x_i - \overline{x})^2}}. \tag{8.3}$$

The conditional distribution of T given $X_i = x_i$, $i = 1, 2, \cdots, n$, is given by the t distribution with $n-2$ degrees of freedom. From this fact, we can obtain the distribution of the sample correlation coefficient $\hat{\rho}$ under $\rho = 0$.

$\rho = 0$ implies that $\beta = 0$. Therefore, under $\rho = 0$, equation (8.3) is rewritten as:

$$\begin{aligned} T &= \frac{\hat{\beta}}{\sqrt{S^2/\sum_{i=1}^n (x_i - \overline{x})^2}} = \frac{\dfrac{\sum_{i=1}^n (x_i - \overline{x})(Y_i - \overline{Y})}{\sum_{i=1}^n (x_i - \overline{x})^2}}{\sqrt{\dfrac{\frac{1}{n-2}\sum_{i=1}^n (Y_i - \overline{Y})^2 - \hat{\beta}^2 \sum_{i=1}^n (x_i - \overline{x})^2}{\sum_{i=1}^n (x_i - \overline{x})^2}}} \\ &= \frac{\left(\dfrac{\sum_{i=1}^n (x_i - \overline{x})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^n (x_i - \overline{x})^2}}\right)\sqrt{n-2}}{\sqrt{\sum_{i=1}^n (Y_i - \overline{Y})^2 - \left(\dfrac{\sum_{i=1}^n (x_i - \overline{x})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^n (x_i - \overline{x})^2}}\right)^2}} \end{aligned} \tag{8.4}$$

$$= \frac{\left(\dfrac{\sum_{i=1}^{n}(x_i-\overline{x})(Y_i-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2\sum_{i=1}^{n}(Y_i-\overline{Y})^2}}\right)\sqrt{n-2}}{\sqrt{1-\left(\dfrac{\sum_{i=1}^{n}(x_i-\overline{x})(Y_i-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2\sum_{i=1}^{n}(Y_i-\overline{Y})^2}}\right)^2}},$$

which has a t distribution with $n-2$ degrees of freedom. Although T depends on $\dfrac{\sum_{i=1}^{n}(x_i-\overline{x})(Y_i-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2\sum_{i=1}^{n}(Y_i-\overline{Y})^2}}$, T is independent of x_i, $i=1,2,\cdots,n$, except for it.
Therefore, replacing x_i by X_i, we have the following:

$$\frac{\left(\dfrac{\sum_{i=1}^{n}(X_i-\overline{X})(Y_i-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\overline{X})^2\sum_{i=1}^{n}(Y_i-\overline{Y})^2}}\right)\sqrt{n-2}}{\sqrt{1-\left(\dfrac{\sum_{i=1}^{n}(X_i-\overline{X})(Y_i-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\overline{X})^2\sum_{i=1}^{n}(Y_i-\overline{Y})^2}}\right)^2}} = \frac{\hat{\rho}\sqrt{n-2}}{\sqrt{1-\hat{\rho}^2}} \sim t(n-2),$$

where $\hat{\rho}$ is given by:

$$\hat{\rho} = \frac{\sum_{i=1}^{n}(X_i-\overline{X})(Y_i-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\overline{X})^2\sum_{i=1}^{n}(Y_i-\overline{Y})^2}}.$$

Thus, the conditional distribution of T given $X_i = x_i$, $i=1,2,\cdots,n$, depends on $\hat{\rho}$ and does not depend on $X_i = x_i$, $i=1,2,\cdots,n$. Therefore, the conditional distribution of T reduces to the unconditional distribution of T. That is, we can obtain the result that the function of $\hat{\rho}$ under $\rho = 0$ is distributed as the t distribution with $n-2$ degrees of freedom. Based on the distribution of T, transforming the variables, the density function of $\hat{\rho}$ under $\rho = 0$ is derived as follows:

$$f(\hat{\rho}) = \frac{\Gamma(\frac{n-1}{2})}{\sqrt{\pi}\Gamma(\frac{n-2}{2})}(1-\hat{\rho})^{\frac{n-4}{2}}. \tag{8.5}$$

Distribution of $\hat{\rho}$ under $\rho \neq 0$: Take equation (8.4) as the test statistic in this case. Now, we define R as:

$$R = \frac{\sum_{i=1}^{n}(x_i-\overline{x})(Y_i-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2}}.$$

The mean and variance of R given $x_1, x_2, \cdots, x_n$ are as follows:

$$\mathrm{E}\left(\frac{\sum_{i=1}^{n}(x_i-\overline{x})(Y_i-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2}}\right) = \beta\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2},$$

$$\mathrm{V}\left(\frac{\sum_{i=1}^{n}(x_i-\overline{x})(Y_i-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2}}\right) = \sigma_Y^2(1-\rho^2),$$

which correspond to the conditional expectations of R given x_i, $i = 1, 2, \cdots, n$. Thus, we obtain $R \sim N(\beta\sqrt{\sum_{i=1}^n (x_i - \overline{x})^2}, \sigma_Y^2(1-\rho^2))$, which implies:

$$\frac{R}{\sigma_Y\sqrt{1-\rho^2}} \sim N(\psi, 1),$$

where ψ is given by:

$$\psi \equiv \frac{\beta\sqrt{\sum_{i=1}^n (x_i - \overline{x})^2}}{\sigma_Y\sqrt{1-\rho^2}} = \frac{\rho\sqrt{\sum_{i=1}^n (x_i - \overline{x})^2}}{\sigma_X\sqrt{1-\rho^2}}.$$

Moreover, as mentioned above, we have:

$$\frac{(n-2)S^2}{\sigma_Y^2(1-\rho^2)} \sim \chi^2(n-2).$$

Then, we can verify that T in equation (8.4) has the noncentral t distribution with $n-2$ degrees of freedom and noncentrality parameter ψ, i.e.,

$$\begin{aligned} T &= \frac{\dfrac{R}{\sigma_Y\sqrt{1-\rho^2}}}{\sqrt{\dfrac{(n-2)S^2}{\sigma_Y^2(1-\rho^2)}\Big/(n-2)}} = \frac{\left(\dfrac{\sum_{i=1}^n (x_i - \overline{x})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^n (x_i - \overline{x})^2}}\right)\sqrt{n-2}}{\sqrt{\displaystyle\sum_{i=1}^n (Y_i - \overline{Y})^2 - \left(\frac{\sum_{i=1}^n (x_i - \overline{x})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^n (x_i - \overline{x})^2}}\right)^2}} \\ &\sim t(n-2; \psi), \end{aligned}$$

which is a conditional distribution of T given $X_i = x_i$, $i = 1, 2, \cdots, n$. See 2.2.14 for the noncentral t distribution. Note that noncentrality parameter ψ is a function of x_1, x_2, $\cdots$, x_n, i.e., $\psi = (\rho/\sqrt{1-\rho^2})\sqrt{\sum_{i=1}^n (x_i - \overline{x})^2}/\sigma_X$. Because ψ is taken as a function of $w = \sum_{i=1}^n (x_i - \overline{x})^2/\sigma_X^2$, the above noncentral t distribution is regarded as the conditional distribution of T given $W = w$, where $W = \sum_{i=1}^n (X_i - \overline{X})^2/\sigma_X^2$ and $\psi^2 = w\rho^2/(1-\rho^2)$.

$$\begin{aligned} f(t|w) &= \frac{e^{-\frac{1}{2}\psi^2}}{\sqrt{(n-2)\pi}\,\Gamma(\frac{n-2}{2})} \sum_{j=0}^{\infty} \Gamma\Big(\frac{n+j-1}{2}\Big)\frac{(\psi t)^j}{j!}\Big(\frac{2}{n-2}\Big)^{\frac{j}{2}}\Big(1 + \frac{t^2}{n-2}\Big)^{-\frac{n+j-1}{2}} \\ &= \frac{e^{-\frac{\rho^2}{2(1-\rho^2)}w}}{\sqrt{(n-2)\pi}\,\Gamma(\frac{n-2}{2})} \sum_{j=0}^{\infty} \Gamma\Big(\frac{n+j-1}{2}\Big)\frac{t^j}{j!}\Big(\frac{w\rho^2}{1-\rho^2}\Big)^{\frac{j}{2}}\Big(\frac{2}{n-2}\Big)^{\frac{j}{2}}\Big(1 + \frac{t^2}{n-2}\Big)^{-\frac{n+j-1}{2}}. \end{aligned}$$

As for W, we have $W \sim \chi^2(n-1)$, which density is denoted by $f(w)$, i.e.,

$$f(w) = \frac{1}{2^{\frac{n-1}{2}}\Gamma(\frac{n-1}{2})} w^{\frac{n-1}{2}-1} e^{-\frac{1}{2}w}.$$

The unconditional distribution of T is derived by integrating w from zero to infinity, i.e.,

$$
\begin{aligned}
f(t) &= \int_0^\infty f(t|w) f(w)\, \mathrm{d}w \\
&= \int_0^\infty \frac{e^{-\frac{\rho^2}{2(1-\rho^2)}w}}{\sqrt{(n-2)\pi}\Gamma(\frac{n-2}{2})} \\
&\qquad \times \sum_{j=0}^{\infty} \Gamma\Big(\frac{n+j-1}{2}\Big)\frac{t^j}{j!}\Big(\frac{w\rho^2}{1-\rho^2}\Big)^{\frac{j}{2}}\Big(\frac{2}{n-2}\Big)^{\frac{j}{2}}\Big(1+\frac{t^2}{n-2}\Big)^{-\frac{n+j-1}{2}} \\
&\qquad \times \frac{1}{2^{\frac{n-1}{2}}\Gamma(\frac{n-1}{2})} w^{\frac{n-1}{2}-1} e^{-\frac{1}{2}w}\, \mathrm{d}w \\
&= \sum_{j=0}^{\infty} \frac{1}{\sqrt{(n-2)\pi}\Gamma(\frac{n-2}{2})}\Gamma\Big(\frac{n+j-1}{2}\Big)\frac{t^j}{j!}\Big(\frac{\rho^2}{1-\rho^2}\Big)^{\frac{j}{2}}\Big(\frac{2}{n-2}\Big)^{\frac{j}{2}}\Big(1+\frac{t^2}{n-2}\Big)^{-\frac{n+j-1}{2}} \\
&\qquad \times \frac{1}{2^{\frac{n-1}{2}}\Gamma(\frac{n-1}{2})} \int_0^\infty w^{\frac{n+j-1}{2}-1} e^{-\frac{1}{2(1-\rho^2)}w}\, \mathrm{d}w \\
&= \sum_{j=0}^{\infty} \frac{1}{\sqrt{(n-2)\pi}\Gamma(\frac{n-2}{2})}\Gamma\Big(\frac{n+j-1}{2}\Big)\frac{t^j}{j!}\Big(\frac{\rho^2}{1-\rho^2}\Big)^{\frac{j}{2}}\Big(\frac{2}{n-2}\Big)^{\frac{j}{2}}\Big(1+\frac{t^2}{n-2}\Big)^{-\frac{n+j-1}{2}} \\
&\qquad \times \frac{1}{2^{\frac{n-1}{2}}\Gamma(\frac{n-1}{2})} (1-\rho^2)^{\frac{n+j-1}{2}} 2^{\frac{n+j-1}{2}}\Gamma\Big(\frac{n+j-1}{2}\Big) \\
&\qquad \times \int_0^\infty \frac{1}{2^{\frac{n+j-1}{2}}\Gamma(\frac{n+j-1}{2})} v^{\frac{n+j-1}{2}-1} e^{-\frac{1}{2}v}\, \mathrm{d}v \\
&= \frac{(1-\rho^2)^{\frac{n-1}{2}}}{\sqrt{(n-2)\pi}\Gamma(\frac{n-1}{2})\Gamma(\frac{n-2}{2})} \sum_{j=0}^{\infty} \Gamma^2\Big(\frac{n+j-1}{2}\Big)\frac{1}{j!}\frac{(2\rho t)^j}{(n-2)^{\frac{j}{2}}}\Big(1+\frac{t^2}{n-2}\Big)^{-\frac{n+j-1}{2}}.
\end{aligned}
\tag{8.6}
$$

In the fourth equality, the transformation $v = w/(1-\rho^2)$ is used. Thus, the unconditional distribution of T is derived as (8.6). Using the density function (8.6), we con construct the confidence interval of ρ and test the null hypothesis such as $H_0 : \rho = \rho_0$. Furthermore, utilizing the relationship between T and $\hat{\rho}$, i.e., $T = \hat{\rho}\sqrt{n-2}/\sqrt{1-\hat{\rho}^2}$, we have the probability density function of $\hat{\rho}$ under $-1 < \rho < 1$ as follows:

$$
f(\hat{\rho};\rho) = \frac{(1-\rho^2)^{\frac{n-1}{2}}(1-\hat{\rho}^2)^{\frac{n-4}{2}}}{\sqrt{\pi}\Gamma(\frac{n-1}{2})\Gamma(\frac{n-2}{2})} \sum_{j=0}^{\infty} \Gamma^2\Big(\frac{n+j-1}{2}\Big)\frac{(2\rho\hat{\rho})^j}{j!}, \tag{8.7}
$$

which density was derived by R.A. Fisher for the first time (see Takeuchi (1963)). Finally, note that the density function (8.5) is equivalent to the density function (8.7) in the case of $\rho = 0$ and therefore (8.7) is consistent with (8.5).

References

Conover, W.J., 1980, *Practical Nonparametric Statistics* (Second Edition), John Wiley & Sons.

Fisher, R.A., 1966, *The Design of Experiments* (eighth edition), New York: Hafner.

Gibbons, J.D. and Chakraborti, S., 1992, *Nonparametric Statistical Inference* (Third Edition, revised and Expanded), Marcel Dekker.

Hogg, R.V. and Craig, A.T., 1995, *Introduction to Mathematical Statistics* (Fifth Edition), Prentice Hall.

Hollander, M. and Wolfe, D.A., 1973, *Nonparametric Statistical Methods*, John Wiley & Sons.

Lehmann, E.L., 1986, *Testing Statistical Hypotheses* (Second Edition), John Wiley & Sons.

Randles, R.H. and Wolfe, D.A., 1979, *Introduction to the Theory of Nonparametric Statistics*, John Wiley & Sons.

Sprent, P., 1989, *Applied Nonparametric Statistical Methods*, Chapman and Hall.

Stuart, A. and Ord, J.K., 1991, *Kendall's Advanced Theory of Statistics, Vol.2* (Fifth Edition), Edward Arnold.

Stuart, A. and Ord, J.K., 1994, *Kendall's Advanced Theory of Statistics, Vol.1* (Sixth Edition), Edward Arnold.

Takeuchi, K., 1963, *Mathematical Statistics* (in Japanese), Toyo-Keizai.

Source Code Index

Index