1 Time Series Analysis (時系列分析)

1.1 Introduction

代表的テキスト：

・J.D. Hamilton (1994) *Econometric Analysis*

沖本・井上訳 (2006)『時系列解析 (上・下)』

・A.C. Harvey (1981) *Time Series Models*

国友・山本訳 (1985)『時系列モデル入門』

・沖本竜義 (2010)『経済・ファイナンスデータの計量時系列分析』

1. **Stationarity (定常性) :**

Let $y_1, y_2, \cdots, y_T$ be time series data.

(a) **Weak Stationarity (弱定常性) :**

$$\mathrm{E}(y_t) = \mu,$$
$$\mathrm{E}((y_t - \mu)(y_{t-\tau} - \mu)) = \gamma(\tau), \qquad \tau = 0, 1, 2, \cdots$$

The first and second moments depend on time difference, not time itself.

(b) **Strong Stationarity (強定常性) :**

Let $f(y_{t_1}, y_{t_2}, \cdots, y_{t_r})$ be the joint distribution of $y_{t_1}, y_{t_2}, \cdots, y_{t_r}$.

$$f(y_{t_1}, y_{t_2}, \cdots, y_{t_r}) = f(y_{t_1+\tau}, y_{t_2+\tau}, \cdots, y_{t_r+\tau})$$

All the moments are same for all τ.

2. **Ergodicity (エルゴード性) :**

   As time difference between two data is large, the two data become independent.

   $y_1, y_2, \cdots, y_T$ is said to be ergodic in mean when $\bar{y}$ converges in probability to $\mathrm{E}(y_t)$.

3. **Auto-covariance Function (自己共分散関数) :**

$$\mathrm{E}((y_t - \mu)(y_{t-\tau} - \mu)) = \gamma(\tau), \qquad \tau = 0, 1, 2, \cdots$$

$$\gamma(\tau) = \gamma(-\tau)$$

4. **Auto-correlation Function (自己相関関数) :**

$$\rho(\tau) = \frac{\mathrm{E}((y_t - \mu)(y_{t-\tau} - \mu))}{\sqrt{\mathrm{Var}(y_t)}\sqrt{\mathrm{Var}(y_{t-\tau})}} = \frac{\gamma(\tau)}{\gamma(0)}$$

Note that $\mathrm{Var}(y_t) = \mathrm{Var}(y_{t-\tau}) = \gamma(0)$.

5. **Sample Mean (標本平均) :**

$$\hat{\mu} = \frac{1}{T}\sum_{t=1}^{T} y_t$$

6. **Sample Auto-covariance (標本自己共分散) :**

$$\hat{\gamma}(\tau) = \frac{1}{T}\sum_{t=\tau+1}^{T} (y_t - \hat{\mu})(y_{t-\tau} - \hat{\mu})$$

7. **Correlogram (コレログラム, or 標本自己相関関数) :**

$$\hat{\rho}(\tau) = \frac{\hat{\gamma}(\tau)}{\hat{\gamma}(0)}$$

8. **Lag Operator (ラグ作要素) :**

$$L^{\tau} y_t = y_{t-\tau}, \qquad \tau = 1, 2, \cdots$$

9. **Likelihood Function (尤度関数)** — Innovation Form :

The joint distribution of $y_1, y_2, \cdots, y_T$ is written as:

$$
\begin{aligned}
f(y_1,, y_2, \cdots, y_T) &= f(y_T|y_{T-1}, \cdots, y_1) f(y_{T-1}, \cdots, y_1) \\
&= f(y_T|y_{T-1}, \cdots, y_1) f(y_{T-1}|y_{T-2}, \cdots, y_1) f(y_{T-2}, \cdots, y_1) \\
&\ \ \vdots \\
&= f(y_T|y_{T-1}, \cdots, y_1) f(y_{T-1}|y_{T-2}, \cdots, y_1) \ \cdots \ f(y_2|y_1) f(y_1) \\
&= f(y_1) \prod_{t=2}^{T} f(y_t|y_{t-1}, \cdots, y_1).
\end{aligned}
$$

Therefore, the log-likelihood function is given by:

$$\log f(y_1, y_2, \cdots, y_T) = \log f(y_1) + \sum_{t=2}^{T} \log f(y_t|y_{t-1}, \cdots, y_1).$$

Under the normality assumption, $f(y_t|y_{t-1}, \cdots, y_1)$ is given by the normal distribution with conditional mean $\mathrm{E}(y_t|y_{t-1}, \cdots, y_1)$ and conditional variance $\mathrm{Var}(y_t|y_{t-1}, \cdots, y_1)$.

1.2 Autoregressive Model (自己回帰モデル or AR モデル)

1. **AR(p) Model :**

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \epsilon_t,$$

which is rewritten as:

$$\phi(L) y_t = \epsilon_t,$$

where

$$\phi(L) = 1 - \phi_1 L - \phi_2 L^2 - \cdots - \phi_p L^p.$$

2. **Stationarity (定常性) :**

Suppose that all the p solutions of x from $\phi(x) = 0$ are real numbers

When the p solutions are greater than one, y_t is stationary.

Suppose that the p solutions include imaginary numbers.

When the p solutions are outside unit circle, y_t is stationary.

3. **Partial Autocorrelation Coefficient (偏自己相関係数), $\phi_{k,k}$:**

The partial autocorrelation coefficient between y_t and y_{t-k}, denoted by $\phi_{k,k}$, is a measure of strength of the relationship between y_t and y_{t-k}, after removing influence of $y_{t-1}, \cdots, y_{t-k+1}$.

$$\phi_{1,1} = \rho(1)$$

$$\begin{pmatrix} 1 & \rho(1) \\ \rho(1) & 1 \end{pmatrix} \begin{pmatrix} \phi_{2,1} \\ \phi_{2,2} \end{pmatrix} = \begin{pmatrix} \rho(1) \\ \rho(2) \end{pmatrix}$$

$$\begin{pmatrix} 1 & \rho(1) & \rho(2) \\ \rho(1) & 1 & \rho(1) \\ \rho(2) & \rho(1) & 1 \end{pmatrix} \begin{pmatrix} \phi_{3,1} \\ \phi_{3,2} \\ \phi_{3,3} \end{pmatrix} = \begin{pmatrix} \rho(1) \\ \rho(2) \\ \rho(3) \end{pmatrix}$$

$$\vdots$$

$$\begin{pmatrix} 1 & \rho(1) & \cdots & \rho(k-2) & \rho(k-1) \\ \rho(1) & 1 & & \rho(k-3) & \rho(k-2) \\ \vdots & \vdots & & \vdots & \vdots \\ \rho(k-1) & \rho(k-2) & \cdots & \rho(1) & 1 \end{pmatrix} \begin{pmatrix} \phi_{k,1} \\ \phi_{k,2} \\ \vdots \\ \phi_{k,k-1} \\ \phi_{k,k} \end{pmatrix} = \begin{pmatrix} \rho(1) \\ \rho(2) \\ \vdots \\ \rho(k) \end{pmatrix}$$

Use Cramer's rule (クラメールの公式) to obtain $\phi_{k,k}$.

$$\phi_{k,k} = \frac{\begin{vmatrix} 1 & \rho(1) & \cdots & \rho(k-2) & \rho(1) \\ \rho(1) & 1 & & \rho(k-3) & \rho(2) \\ \vdots & \vdots & & \vdots & \vdots \\ \rho(k-1) & \rho(k-2) & \cdots & \rho(1) & \rho(k) \end{vmatrix}}{\begin{vmatrix} 1 & \rho(1) & \cdots & \rho(k-2) & \rho(k-1) \\ \rho(1) & 1 & & \rho(k-3) & \rho(k-2) \\ \vdots & \vdots & & \vdots & \vdots \\ \rho(k-1) & \rho(k-2) & \cdots & \rho(1) & 1 \end{vmatrix}}$$

Example: AR(1) Model: $y_t = \phi_1 y_{t-1} + \epsilon_t$

1. The stationarity condition is: the solution of $\phi(x) = 1 - \phi_1 x = 0$, i.e., $x = 1/\phi_1$, is greater than one, or equivalently, $\phi_1 < 1$.

2. Rewriting the AR(1) model,

$$\begin{aligned} y_t &= \phi_1 y_{t-1} + \epsilon_t \\ &= \phi_1^2 y_{t-2} + \epsilon_t + \phi_1 \epsilon_{t-1} \\ &= \phi_1^3 y_{t-3} + \epsilon_t + \phi_1 \epsilon_{t-1} + \phi_1^2 \epsilon_{t-2} \\ &\vdots \end{aligned}$$

$$= \phi_1^s y_{t-s} + \epsilon_t + \phi_1 \epsilon_{t-1} + \cdots + \phi_1^{s-1} \epsilon_{t-s+1}.$$

As s is large, ϕ_1^s approaches zero. $\Longrightarrow$ Stationarity condition

3. For stationarity, $y_t = \phi_1 y_{t-1} + \epsilon_t$ is rewritten as:

$$y_t = \epsilon_t + \phi_1 \epsilon_{t-1} + \phi_1^2 \epsilon_{t-2} + \cdots$$

MA representation of AR model.

(MA will be discussed later.)

4. Mean of AR(1) process, μ

$$\begin{aligned}\mu = \mathrm{E}(y_t) &= \mathrm{E}(\epsilon_t + \phi_1\epsilon_{t-1} + \phi_1^2\epsilon_{t-2} + \cdots) \\ &= \mathrm{E}(\epsilon_t) + \phi_1\mathrm{E}(\epsilon_{t-1}) + \phi_1^2\mathrm{E}(\epsilon_{t-2}) + \cdots = 0\end{aligned}$$

5. Autocovariance and autocorrelation functions of the AR(1) process:

   Rewriting the AR(1) process, we have:

$$y_t = \phi_1^\tau y_{t-\tau} + \epsilon_t + \phi_1\epsilon_{t-1} + \cdots + \phi_1^{\tau-1}\epsilon_{t-\tau+1}.$$

Therefore, the autocovariance function of AR(1) process is:

$$\begin{aligned}\gamma(\tau) &= \mathrm{E}((y_t - \mu)(y_{t-\tau} - \mu)) = \mathrm{E}(y_t y_{t-\tau}) \\ &= \mathrm{E}\Big((\phi_1^\tau y_{t-\tau} + \epsilon_t + \phi_1\epsilon_{t-1} + \cdots + \phi_1^{\tau-1}\epsilon_{t-\tau+1})y_{t-\tau}\Big) \\ &= \phi_1^\tau \mathrm{E}(y_{t-\tau}y_{t-\tau}) + \mathrm{E}(\epsilon_t y_{t-\tau}) + \phi_1 \mathrm{E}(\epsilon_{t-1}y_{t-\tau}) + \cdots + \phi_1^{\tau-1}\mathrm{E}(\epsilon_{t-\tau+1}y_{t-\tau}) \\ &= \phi_1^\tau \gamma(0).\end{aligned}$$

The autocorrelation function of AR(1) process is:

$$\rho(\tau) = \frac{\gamma(\tau)}{\gamma(0)} = \phi_1^\tau.$$

Multiply $y_{t-\tau}$ on both sides of the AR(1) process and take the expectation:

$$\mathrm{E}(y_t y_{t-\tau}) = \phi_1 \mathrm{E}(y_{t-1} y_{t-\tau}) + \mathrm{E}(\epsilon_t y_{t-\tau})$$

$$\gamma(\tau) = \begin{cases} \phi_1 \gamma(\tau - 1), & \text{for } \tau \neq 0, \\ \phi_1 \gamma(\tau - 1) + \sigma^2, & \text{for } \tau = 0. \end{cases}$$

Using $\gamma(\tau) = \gamma(-\tau)$, $\gamma(\tau)$ for $\tau = 0$ is given by:

$$\gamma(0) = \phi_1 \gamma(1) + \sigma^2 = \phi_1^2 \gamma(0) + \sigma^2.$$

Note that $\gamma(1) = \phi_1 \gamma(0)$.

Therefore, $\gamma(0)$ is given by:

$$\gamma(0) = \frac{\sigma^2}{1 - \phi_1^2}$$

6. Partial autocorrelation function of AR(1) process:

$$\phi_{1,1} = \rho(1) = \phi_1$$

$$\phi_{2,2} = \frac{\begin{vmatrix} 1 & \rho(1) \\ \rho(1) & \rho(2) \end{vmatrix}}{\begin{vmatrix} 1 & \rho(1) \\ \rho(1) & 1 \end{vmatrix}} = \frac{\rho(2) - \rho(1)^2}{1 - \rho(1)^2} = 0$$

7. Estimation of AR(1) model:

(a) Likelihood function

$$\begin{aligned}\log f(y_T, \cdots, y_1) &= \log f(y_1) + \sum_{t=1}^{T} \log f(y_t|y_{t-1}, \cdots, y_1) \\ &= -\frac{1}{2}\log(2\pi) - \frac{1}{2}\log\left(\frac{\sigma^2}{1-\phi_1^2}\right) - \frac{1}{\sigma^2/(1-\phi_1^2)}y_1^2 \\ &\quad -\frac{T-1}{2}\log(2\pi) - \frac{T-1}{2}\log(\sigma^2) - \frac{1}{\sigma^2}\sum_{t=2}^{T}(y_t - \phi_1 y_{t-1})^2\end{aligned}$$

$$= -\frac{T}{2}\log(2\pi) - \frac{T}{2}\log(\sigma^2) - \frac{1}{2}\log\left(\frac{1}{1-\phi_1^2}\right)$$
$$-\frac{1}{2\sigma^2/(1-\phi_1^2)}y_1^2 - \frac{1}{2\sigma^2}\sum_{t=2}^{T}(y_t - \phi_1 y_{t-1})^2$$

Note as follows:

$$f(y_1) = \frac{1}{\sqrt{2\pi\sigma^2/(1-\phi_1^2)}}\exp\left(-\frac{1}{2\sigma^2/(1-\phi_1^2)}y_1^2\right)$$
$$f(y_t|y_{t-1},\cdots,y_1) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{1}{2\sigma^2}(y_t - \phi_1 y_{t-1})^2\right)$$

$$\frac{\partial \log f(y_T,\cdots,y_1)}{\partial \sigma^2} = -\frac{T}{2}\frac{1}{\sigma^2} + \frac{1}{2\sigma^4/(1-\phi_1^2)}y_1^2 + \frac{1}{2\sigma^4}\sum_{t=2}^{T}(y_t-\phi_1 y_{t-1})^2 = 0$$

$$\frac{\partial \log f(y_T,\cdots,y_1)}{\partial \phi_1} = -\frac{\phi_1}{1-\phi_1^2} + \frac{\phi_1}{\sigma^2}y_1^2 + \frac{1}{\sigma^2}\sum_{t=2}^{T}(y_t-\phi_1 y_{t-1})y_{t-1} = 0$$

The MLE of ϕ_1 and σ^2 satisfies the above two equation.

$$\tilde{\sigma}^2 = \frac{1}{T}\left((1-\tilde{\phi}_1^2)y_1^2 + \sum_{t=2}^{T}(y_t-\tilde{\phi}_1 y_{t-1})^2\right)$$

$$\tilde{\phi}_1 = \frac{\sum_{t=2}^{T} y_t y_{t-1}}{\sum_{t=2}^{T} y_{t-1}^2} + \left(\tilde{\phi}_1 y_1^2 - \frac{\tilde{\sigma}^2\tilde{\phi}_1}{1-\tilde{\phi}_1^2}\right)\Big/\sum_{t=2}^{T} y_{t-1}^2$$

(b) Ordinary Least Squares (OLS) Method

$$S(\phi_1) = \sum_{t=2}^{T} (y_t - \phi_1 y_{t-1})^2$$

is minimized with respect to ϕ_1.

$$\hat{\phi}_1 = \frac{\sum_{t=2}^{T} y_{t-1} y_t}{\sum_{t=2}^{T} y_{t-1}^2} = \phi_1 + \frac{\sum_{t=2}^{T} y_{t-1} \epsilon_t}{\sum_{t=2}^{T} y_{t-1}^2} = \phi_1 + \frac{(1/T) \sum_{t=2}^{T} y_{t-1} \epsilon_t}{(1/T) \sum_{t=2}^{T} y_{t-1}^2}$$

$$\longrightarrow \phi_1 + \frac{\mathrm{E}(y_{t-1} \epsilon_t)}{\mathrm{E}(y_{t-1}^2)} = \phi_1$$

OLSE of ϕ_1 is a consistent estimator.

The following equations are utilized.

$$\mathrm{E}(y_{t-1}\epsilon_t) = 0$$

$$\mathrm{E}(y_{t-1}^2) = \mathrm{Var}(y_{t-1}) = \gamma(0)$$

8. Asymptotic distribution of OLSE $\hat{\phi}_1$:

$$\sqrt{T}(\hat{\phi}_1 - \phi_1) \longrightarrow N(0, 1 - \phi_1^2)$$

Proof:

$y_{t-1}\epsilon_t$, $t = 1, 2, \cdots, T$, are distributed with mean zero and variance $\dfrac{\sigma_\epsilon^4}{1 - \phi_1^2}$.

From the central limit theorem,

$$\frac{(1/T)\sum_{t=1}^{T} y_{t-1}\epsilon_t}{\sqrt{\sigma_\epsilon^4/(1-\phi_1^2)}/\sqrt{T}} \longrightarrow N(0,1)$$

Rewriting,

$$\frac{1}{\sqrt{T}}\sum_{t=1}^{T} y_{t-1}\epsilon_t \longrightarrow N(0, \frac{\sigma_\epsilon^4}{1-\phi_1^2}).$$

Next,

$$\frac{1}{T}\sum_{t=1}^{T} y_{t-1}^2 \longrightarrow \mathrm{E}(y_{t-1}^2) = \gamma(0) = \frac{\sigma_\epsilon^2}{1-\phi_1^2}$$

yields:

$$\sqrt{T}(\hat{\phi}_1 - \phi_1) = \frac{(1/\sqrt{T})\sum_{t=1}^{T} y_{t-1}\epsilon_t}{(1/T)\sum_{t=1}^{T} y_{t-1}^2} \longrightarrow N(0, 1-\phi_1^2)$$

9. Some formulas:

   (a) Central Limit Theorem

   Random variables $x_1, x_2, \cdots, x_T$ are mutually independently distributed with mean μ and variance σ^2.

   Define $\bar{x} = (1/T)\sum_{t=1}^{T} x_t$.

Then,

$$\frac{\overline{x} - \mathrm{E}(\overline{x})}{\sqrt{\mathrm{V}(\overline{x})}} = \frac{\overline{x} - \mu}{\sigma / \sqrt{T}} \longrightarrow N(0, 1)$$

(b) Central Limit Theorem II

Random variables $x_1, x_2, \cdots, x_T$ are distributed with mean μ and variance σ^2.

Define $\overline{x} = (1/T) \sum_{t=1}^{T} x_t$.

Then,

$$\frac{\overline{x} - \mathrm{E}(\overline{x})}{\sqrt{\mathrm{V}(\overline{x})}} \longrightarrow N(0, 1)$$

(c) Let x and y be random variables.

y converges in distribution to a distribution, and x converges in probability to a fixed value.

Then, xy converges in distribution.

For example, consider:

$$y \longrightarrow N(\mu, \sigma^2), \qquad x \longrightarrow c.$$

Then, we obtain:

$$xy \longrightarrow N(c\mu, c^2\sigma^2)$$

10. **AR(1) +drift:** $y_t = \mu + \phi_1 y_{t-1} + \epsilon_t$

Mean:

Using the lag operator,

$$\phi(L) y_t = \mu + \epsilon_t$$

where $\phi(L) = 1 - \phi_1 L$.

Multiply $\phi(L)^{-1}$ on both sides. Then, when $|\phi_1| < 1$, we have:

$$y_t = \phi(L)^{-1}\mu + \phi(L)^{-1}\epsilon_t.$$

Taking the expectation on both sides,

$$\begin{aligned} \mathrm{E}(y_t) &= \phi(L)^{-1}\mu + \phi(L)^{-1}\mathrm{E}(\epsilon_t) \\ &= \phi(1)^{-1}\mu = \frac{\mu}{1-\phi_1} \end{aligned}$$

Example: AR(2) Model: Consider $y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \epsilon_t$.

1. The stationarity condition is: two solutions of x from $\phi(x) = 1 - \phi_1 x - \phi_2 x^2 = 0$ are outside the unit circle.

2. Rewriting the AR(2) model,

$$(1 - \phi_1 L - \phi_2 L^2) y_t = \epsilon_t.$$

Let $1/\alpha_1$ and $1/\alpha_2$ be the solutions of $\phi(x) = 0$.

Then, the AR(2) model is written as:

$$(1 - \alpha_1 L)(1 - \alpha_2 L) y_t = \epsilon_t,$$

which is rewritten as:

$$
\begin{aligned}
y_t &= \frac{1}{(1-\alpha_1 L)(1-\alpha_2 L)}\epsilon_t \\
&= \left(\frac{\alpha_1/(\alpha_1-\alpha_2)}{1-\alpha_1 L} + \frac{-\alpha_2/(\alpha_1-\alpha_2)}{1-\alpha_2 L}\right)\epsilon_t
\end{aligned}
$$

3. **Mean of AR(2) Model:**

When y_t is stationary, i.e., α_1 and α_2 are within the unit circle,

$$\mu = \mathrm{E}(y_t) = \mathrm{E}(\phi(L)\epsilon_t) = 0$$

4. **Autocovariance Function of AR(2) Model:**

$$\begin{aligned}\gamma(\tau) &= \mathrm{E}((y_t - \mu)(y_{t-\tau} - \mu)) = \mathrm{E}(y_t y_{t-\tau}) \\ &= \mathrm{E}\Big((\phi_1 y_{t-1} + \phi_2 y_{t-2} + \epsilon_t) y_{t-\tau}\Big) \\ &= \phi_1 \mathrm{E}(y_{t-1} y_{t-\tau}) + \phi_2 \mathrm{E}(y_{t-2} y_{t-\tau}) + \mathrm{E}(\epsilon_t y_{t-\tau}) \\ &= \begin{cases} \phi_1 \gamma(\tau - 1) + \phi_2 \gamma(\tau - 2), & \text{for } \tau \neq 0, \\ \phi_1 \gamma(\tau - 1) + \phi_2 \gamma(\tau - 2) + \sigma_\epsilon^2, & \text{for } \tau = 0. \end{cases}\end{aligned}$$

The initial condition is obtained by solving the following three equations:

$$\gamma(0) = \phi_1 \gamma(1) + \phi_2 \gamma(2) + \sigma_\epsilon^2,$$

$$\gamma(1) = \phi_1\gamma(0) + \phi_2\gamma(1),$$

$$\gamma(2) = \phi_1\gamma(1) + \phi_2\gamma(0).$$

Therefore, the initial conditions are given by:

$$\gamma(0) = \left(\frac{1-\phi_2}{1+\phi_2}\right)\frac{\sigma_\epsilon^2}{(1-\phi_2)^2 - \phi_1^2},$$

$$\gamma(1) = \frac{\phi_1}{1-\phi_2}\gamma(0) = \left(\frac{\phi_1}{1-\phi_2}\right)\left(\frac{1-\phi_2}{1+\phi_2}\right)\frac{\sigma_\epsilon^2}{(1-\phi_2)^2 - \phi_1^2}.$$

Given $\gamma(0)$ and $\gamma(1)$, we obtain $\gamma(\tau)$ as follows:

$$\gamma(\tau) = \phi_1\gamma(\tau-1) + \phi_2\gamma(\tau-2), \qquad \text{for } \tau = 2, 3, \cdots.$$

5. **Another solution for $\gamma(0)$:**

From $\gamma(0) = \phi_1\gamma(1) + \phi_2\gamma(2) + \sigma_\epsilon^2$,

$$\gamma(0) = \frac{\sigma_\epsilon^2}{1 - \phi_1\rho(1) - \phi_2\rho(2)}$$

where

$$\rho(1) = \frac{\phi_1}{1 - \phi_2}, \qquad \rho(2) = \phi_1\rho(1) + \phi_2 = \frac{\phi_1^2 + (1 - \phi_2)\phi_2}{1 - \phi_2}.$$

6. **Autocorrelation Function of AR(2) Model:**

Given $\rho(1)$ and $\rho(2)$,

$$\rho(\tau) = \phi_1\rho(\tau - 1) + \phi_2\rho(\tau - 2), \qquad \text{for } \tau = 3, 4, \cdots,$$

7. $\boldsymbol{\phi_{k,k}}$ **= Partial Autocorrelation Coefficient of AR(2) Process:**

$$\begin{pmatrix} 1 & \rho(1) & \cdots & \rho(k-2) & \rho(k-1) \\ \rho(1) & 1 & & \rho(k-3) & \rho(k-2) \\ \vdots & \vdots & & \vdots & \vdots \\ \rho(k-1) & \rho(k-2) & \cdots & \rho(1) & 1 \end{pmatrix} \begin{pmatrix} \phi_{k,1} \\ \phi_{k,2} \\ \vdots \\ \phi_{k,k-1} \\ \phi_{k,k} \end{pmatrix} = \begin{pmatrix} \rho(1) \\ \rho(2) \\ \vdots \\ \rho(k) \end{pmatrix},$$

for $k = 1, 2, \cdots$.

$$\phi_{k,k} = \frac{\begin{vmatrix} 1 & \rho(1) & \cdots \rho(k-2) & \rho(1) \\ \rho(1) & 1 & \rho(k-3) & \rho(2) \\ \vdots & \vdots & \vdots & \vdots \\ \rho(k-1) & \rho(k-2) \cdots & \rho(1) & \rho(k) \end{vmatrix}}{\begin{vmatrix} 1 & \rho(1) & \cdots \rho(k-2) & \rho(k-1) \\ \rho(1) & 1 & \rho(k-3) & \rho(k-2) \\ \vdots & \vdots & \vdots & \vdots \\ \rho(k-1) & \rho(k-2) \cdots & \rho(1) & 1 \end{vmatrix}}$$

Autocovariance Functions:

$$\gamma(1) = \phi_1\gamma(0) + \phi_2\gamma(1),$$

$$\gamma(2) = \phi_1\gamma(1) + \phi_2\gamma(0),$$

$$\gamma(\tau) = \phi_1\gamma(\tau-1) + \phi_2\gamma(\tau-2), \qquad \text{for } \tau = 3, 4, \cdots.$$

Autocorrelation Functions:

$$\rho(1) = \phi_1 + \phi_2\rho(1) = \frac{\phi_1}{1-\phi_2},$$

$$\rho(2) = \phi_1\rho(1) + \phi_2 = \frac{\phi_1^2}{1-\phi_2} + \phi_2,$$

$$\rho(\tau) = \phi_1\rho(\tau-1) + \phi_2\rho(\tau-2), \qquad \text{for } \tau = 3, 4, \cdots.$$

$$\phi_{1,1} = \rho(1) = \frac{\phi_1}{1 - \phi_2}$$

$$\phi_{2,2} = \frac{\begin{vmatrix} 1 & \rho(1) \\ \rho(1) & \rho(2) \end{vmatrix}}{\begin{vmatrix} 1 & \rho(1) \\ \rho(1) & 1 \end{vmatrix}} = \frac{\rho(2) - \rho(1)^2}{1 - \rho(1)^2} = \phi_2$$

$$\phi_{3,3} = \frac{\begin{vmatrix} 1 & \rho(1) & \rho(1) \\ \rho(1) & 1 & \rho(2) \\ \rho(2) & \rho(1) & \rho(3) \end{vmatrix}}{\begin{vmatrix} 1 & \rho(1) & \rho(2) \\ \rho(1) & 1 & \rho(1) \\ \rho(2) & \rho(1) & 1 \end{vmatrix}}$$

$$= \frac{(\rho(3) - \rho(1)\rho(2)) - \rho(1)^2(\rho(3) - \rho(1)) + \rho(2)\rho(1)(\rho(2) - 1)}{(1 - \rho(1)^2) - \rho(1)^2(1 - \rho(2)) + \rho(2)(\rho(1)^2 - \rho(2))} = 0.$$

8. **Log-Likelihood Function — Innovation Form:**

$$\log f(y_T, \cdots, y_1) = \log f(y_2, y_1) + \sum_{t=3}^{T} \log f(y_t | y_{t-1}, \cdots, y_1)$$

where

$$f(y_2, y_1) = \frac{1}{2\pi} \begin{vmatrix} \gamma(0) & \gamma(1) \\ \gamma(1) & \gamma(0) \end{vmatrix}^{-1/2} \exp\left(-\frac{1}{2} (y_1 \; y_2) \begin{pmatrix} \gamma(0) & \gamma(1) \\ \gamma(1) & \gamma(0) \end{pmatrix}^{-1} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \right),$$

$$f(y_t | y_{t-1}, \cdots, y_1) = \frac{1}{\sqrt{2\pi\sigma_\epsilon^2}} \exp\left(-\frac{1}{2\sigma_\epsilon^2} (y_t - \phi_1 y_{t-1} - \phi_2 y_{t-2})^2 \right).$$

Note as follows:

$$\begin{pmatrix} \gamma(0) & \gamma(1) \\ \gamma(1) & \gamma(0) \end{pmatrix} = \gamma(0) \begin{pmatrix} 1 & \rho(1) \\ \rho(1) & 1 \end{pmatrix} = \gamma(0) \begin{pmatrix} 1 & \phi_1/(1-\phi_2) \\ \phi_1/(1-\phi_2) & 1 \end{pmatrix}.$$

9. **AR(2) +drift:** $y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \epsilon_t$

   Mean:

   Rewriting the AR(2)+drift model,

$$\phi(L) y_t = \mu + \epsilon_t$$

   where $\phi(L) = 1 - \phi_1 L - \phi_2 L^2$.

   Under the stationarity assumption, we can rewrite the AR(2)+drift model as follows:

$$y_t = \phi(L)^{-1}\mu + \phi(L)^{-1}\epsilon_t.$$

Therefore,

$$\mathrm{E}(y_t) = \phi(L)^{-1}\mu + \phi(L)^{-1}\mathrm{E}(\epsilon_t) = \phi(1)^{-1}\mu = \frac{\mu}{1-\phi_1-\phi_2}$$

Example: AR(p) model: Consider $y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \epsilon_t$.

1. **Variance of AR(p) Process:**

   Under the stationarity condition (i.e., the p solutions of x from $\phi(x) = 0$ are outside the unit circle),

$$\gamma(0) = \frac{\sigma_\epsilon^2}{1-\phi_1\rho(1) - \cdots - \phi_p\rho(p)}.$$

Note that $\gamma(\tau) = \rho(\tau)\gamma(0)$.

Solve the following simultaneous equations for $\tau = 0, 1, \cdots, p$:

$$\begin{aligned}\gamma(\tau) &= \mathrm{E}((y_t - \mu)(y_{t-\tau} - \mu)) = \mathrm{E}(y_t y_{t-\tau}) \\ &= \begin{cases} \phi_1\gamma(\tau-1) + \phi_2\gamma(\tau-2) + \cdots + \phi_p\gamma(\tau-p), & \text{for } \tau \neq 0, \\ \phi_1\gamma(\tau-1) + \phi_2\gamma(\tau-2) + \cdots + \phi_p\gamma(\tau-p) + \sigma_\epsilon^2, & \text{for } \tau = 0. \end{cases}\end{aligned}$$

2. **Estimation of AR(p) Model:**

   1. **OLS:**

$$\min_{\phi_1,\cdots,\phi_p} \sum_{t=p+1}^{T} (y_t - \phi_1 y_{t-1} - \phi_2 y_{t-2} - \cdots - \phi_p y_{t-p})^2$$

   2. **MLE:**

$$\max_{\phi_1,\cdots,\phi_p} \log f(y_T,\cdots,y_1)$$

where

$$\log f(y_T,\cdots,y_1) = \log f(y_p,\cdots,y_2,y_1) + \sum_{t=p+1}^{T} \log f(y_t|y_{t-1},\cdots,y_1),$$

$$f(y_p,\cdots,y_2,y_1) = (2\pi)^{-p/2}|V|^{-1/2}\exp\left(-\frac{1}{2}(y_1\ y_2\ \cdots\ y_p)V^{-1}\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{pmatrix}\right)$$

$$V = \gamma(0)\begin{pmatrix} 1 & \rho(1) & \cdots & \rho(p-2) & \rho(p-1) \\ \rho(1) & 1 & & \rho(p-3) & \rho(p-2) \\ \vdots & \vdots & & \vdots & \vdots \\ \rho(p-1) & \rho(p-2) & \cdots & \rho(1) & 1 \end{pmatrix}$$

$$f(y_t|y_{t-1},\cdots,y_1) = \frac{1}{\sqrt{2\pi\sigma_\epsilon^2}}\exp\Big(-\frac{1}{2\sigma_\epsilon^2}(y_t - \phi_1 y_{t-1} - \phi_2 y_{t-2} - \cdots - \phi_p y_{t-p})^2\Big)$$

3. Yule=Walker (ユール・ウォーカー) Equation:

Multiply y_{t-1}, y_{t-2}, $\cdots$, y_{t-p} on both sides of $y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \epsilon_t = y_t$, take expectations for each case, and divide by the sample variance $\hat{\gamma}(0)$.

$$\begin{pmatrix} 1 & \hat{\rho}(1) & \cdots & \hat{\rho}(p-2) & \hat{\rho}(p-1) \\ \hat{\rho}(1) & 1 & & \hat{\rho}(p-3) & \hat{\rho}(p-2) \\ \vdots & \vdots & & \vdots & \vdots \\ \hat{\rho}(p-1) & \hat{\rho}(p-2) & \cdots & \hat{\rho}(1) & 1 \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_{p-1} \\ \phi_p \end{pmatrix} = \begin{pmatrix} \hat{\rho}(1) \\ \hat{\rho}(2) \\ \vdots \\ \hat{\rho}(p) \end{pmatrix}$$

where

$$\hat{\gamma}(\tau) = \frac{1}{T}\sum_{t=\tau+1}^{T}(y_t - \hat{\mu})(y_{t-\tau} - \hat{\mu}), \qquad \hat{\rho}(\tau) = \frac{\hat{\gamma}(\tau)}{\hat{\gamma}(0)}.$$

3. **AR(p) +drift:** $y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots \phi_p y_{t-p} + \epsilon_t$

   Mean:

$$\phi(L)y_t = \mu + \epsilon_t$$

   where $\phi(L) = 1 - \phi_1 L - \phi_2 L^2 - \cdots - \phi_p L^p$.

$$y_t = \phi(L)^{-1}\mu + \phi(L)^{-1}\epsilon_t$$

Taking the expectation on both sides,

$$\mathrm{E}(y_t) = \phi(L)^{-1}\mu + \phi(L)^{-1}\mathrm{E}(\epsilon_t) = \phi(1)^{-1}\mu$$
$$= \frac{\mu}{1 - \phi_1 - \phi_2 - \cdots - \phi_p}$$

4. **Partial Autocorrelation of AR(p) Process:**

$\phi_{k,k} = 0$ for $k = p + 1, p + 2, \cdots$.

1.3 MA Model

MA (Moving Average，移動平均) Model:

1. MA(q)

$$y_t = \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots + \theta_q \epsilon_{t-q},$$

which is rewritten as:

$$y_t = \theta(L)\epsilon_t,$$

where

$$\theta(L) = 1 + \theta_1 L + \theta_2 L^2 + \cdots + \theta_q L^q.$$

2. **Invertibility (反転可能性):**

   The q solutions of x from $\theta(x) = 1 + \theta_1 x + \theta_2 x^2 + \cdots + \theta_q x^q = 0$ の q are outside the unit circle.

   $\Longrightarrow$ MA(q) model is rewritten as AR(∞) model.

Example: MA(1) Model: $y_t = \epsilon_t + \theta_1 \epsilon_{t-1}$

1. **Mean of MA(1) Process:**

$$\mathrm{E}(y_t) = \mathrm{E}(\epsilon_t + \theta_1 \epsilon_{t-1}) = \mathrm{E}(\epsilon_t) + \theta_1 \mathrm{E}(\epsilon_{t-1}) = 0$$

2. **Autocovariance Function of MA(1) Process:**

$$\gamma(0) = \mathrm{E}(y_t^2) = \mathrm{E}(\epsilon_t + \theta_1\epsilon_{t-1})^2 = \mathrm{E}(\epsilon_t^2 + 2\theta_1\epsilon_t\epsilon_{t-1} + \theta_1^2\epsilon_{t-1}^2)$$
$$= \mathrm{E}(\epsilon_t^2) + 2\theta_1\mathrm{E}(\epsilon_t\epsilon_{t-1}) + \theta_1^2\mathrm{E}(\epsilon_{t-1}^2) = (1 + \theta_1^2)\sigma_\epsilon^2$$

$$\gamma(1) = \mathrm{E}(y_t y_{t-1}) = \mathrm{E}((\epsilon_t + \theta_1\epsilon_{t-1})(\epsilon_{t-1} + \theta_1\epsilon_{t-2})) = \theta_1\sigma_\epsilon^2$$

$$\gamma(2) = \mathrm{E}(y_t y_{t-2}) = \mathrm{E}((\epsilon_t + \theta_1\epsilon_{t-1})(\epsilon_{t-2} + \theta_1\epsilon_{t-3})) = 0$$

3. **Autocorrelation Function of MA(1) Process:**

$$\rho(\tau) = \frac{\gamma(\tau)}{\gamma(0)} = \begin{cases} \dfrac{\theta_1}{1+\theta_1^2}, & \text{for } \tau = 1, \\ 0, & \text{for } \tau = 2, 3, \cdots. \end{cases}$$

Let x be $\rho(1)$.

$$\frac{\theta_1}{1+\theta_1^2} = x, \qquad \text{i.e.,} \qquad x\theta_1^2 - \theta + x = 0.$$

θ_1 should be a real number.

$$1 - 4x^2 > 0, \qquad \text{i.e.,} \qquad -\frac{1}{2} \le \rho(1) \le \frac{1}{2}.$$

4. **Invertibility Condition of MA(1) Process:**

$$\begin{aligned}
\epsilon_t &= -\theta_1\epsilon_{t-1} + y_t \\
&= (-\theta_1)^2\epsilon_{t-2} + y_t + (-\theta_1)y_{t-1} \\
&= (-\theta_1)^3\epsilon_{t-3} + y_t + (-\theta_1)y_{t-1} + (-\theta_1)^2y_{t-2} \\
&\quad\vdots \\
&= (-\theta_1)^s\epsilon_{t-s} + y_t + (-\theta_1)y_{t-1} + (-\theta_1)^2y_{t-2} + \cdots + (-\theta_1)^{t-s+1}y_{t-s+1}
\end{aligned}$$

When $(-\theta_1)^s \epsilon_{t-s} \longrightarrow 0$, the MA(1) model is written as the AR(∞) model, i.e.,

$$y_t = -(-\theta_1)y_{t-1} - (-\theta_1)^2 y_{t-2} - \cdots - (-\theta_1)^{t-s+1} y_{t-s+1} - \cdots + \epsilon_t$$

5. **Likelihood Function of MA(1) Process:**

The autocovariance functions are: $\gamma(0) = (1 + \theta_1^2)\sigma_\epsilon^2$, $\gamma(1) = \theta_1 \sigma_\epsilon^2$, and $\gamma(\tau) = 0$ for $\tau = 2, 3, \cdots$.

The joint distribution of $y_1, y_2, \cdots, y_T$ is:

$$f(y_1, y_2, \cdots, y_T) = \frac{1}{(2\pi)^{T/2}} |V|^{-1/2} \exp\left(-\frac{1}{2} Y' V^{-1} Y\right)$$

where

$$Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_T \end{pmatrix}, \qquad V = \sigma_\epsilon^2 \begin{pmatrix} 1+\theta_1^2 & \theta_1 & 0 & \cdots & 0 \\ \theta_1 & 1+\theta_1^2 & \theta_1 & \ddots & \vdots \\ 0 & \theta_1 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & 1+\theta_1^2 & \theta_1 \\ 0 & \cdots & 0 & \theta_1 & 1+\theta_1^2 \end{pmatrix}.$$

6. **MA(1) +drift:** $y_t = \mu + \epsilon_t + \theta_1 \epsilon_{t-1}$

Mean of MA(1) Process:

$$y_t = \mu + \theta(L)\epsilon_t,$$

where $\theta(L) = 1 + \theta_1 L$.

Taking the expectation,

$$\mathrm{E}(y_t) = \mu + \theta(L)\mathrm{E}(\epsilon_t) = \mu.$$

Example: MA(2) Model: $y_t = \epsilon_t + \theta_1\epsilon_{t-1} + \theta_2\epsilon_{t-2}$

1. Autocovariance Function of MA(2) Process:

$$\gamma(\tau) = \begin{cases} (1 + \theta_1^2 + \theta_2^2)\sigma_\epsilon^2, & \text{for } \tau = 0, \\ (\theta_1 + \theta_1\theta_2)\sigma_\epsilon^2, & \text{for } \tau = 1, \\ \theta_2\sigma_\epsilon^2, & \text{for } \tau = 2, \\ 0, & \text{otherwise.} \end{cases}$$

2. let $-1/\beta_1$ and $-1/\beta_2$ be two solutions of x from $\theta(x) = 0$.

   For invertibility condition, both β_1 and β_2 should be less than one in absolute value.

Then, the MA(2) model is represented as:

$$\begin{aligned} y_t &= \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} \\ &= (1 + \theta_1 L + \theta_2 L^2)\epsilon_t \\ &= (1 + \beta_1 L)(1 + \beta_2 L)\epsilon_t \end{aligned}$$

AR(∞) representation of the MA(2) model is given by:

$$\begin{aligned} \epsilon_t &= \frac{1}{(1 + \beta_1 L)(1 + \beta_2 L)} y_t \\ &= \left(\frac{\beta_1/(\beta_1 - \beta_2)}{1 + \beta_1 L} + \frac{-\beta_2/(\beta_1 - \beta_2)}{1 + \beta_2 L} \right) y_t \end{aligned}$$

3. **Likelihood Function:**

$$f(y_1, y_2, \cdots, y_T) = \frac{1}{(2\pi)^{T/2}} |V|^{-1/2} \exp\left(-\frac{1}{2} Y' V^{-1} Y\right)$$

where

$$Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_T \end{pmatrix}, \quad V = \sigma_\epsilon^2 \begin{pmatrix} 1+\theta_1^2+\theta_2^2 & \theta_1+\theta_1\theta_2 & \theta_2 & & 0 \\ \theta_1+\theta_1\theta_2 & 1+\theta_1^2+\theta_2^2 & \theta_1+\theta_1\theta_2 & \ddots & \\ \theta_2 & \theta_1+\theta_1\theta_2 & \ddots & \ddots & \theta_2 \\ & \ddots & \ddots & 1+\theta_1^2+\theta_2^2 & \theta_1+\theta_1\theta_2 \\ 0 & & \theta_2 & \theta_1+\theta_1\theta_2 & 1+\theta_1^2+\theta_2^2 \end{pmatrix}$$

4. **MA(2) +drift:** $y_t = \mu + \epsilon_t + \theta_1\epsilon_{t-1} + \theta_2\epsilon_{t-2}$

   Mean:

$$y_t = \mu + \theta(L)\epsilon_t,$$

   where $\theta(L) = 1 + \theta_1 L + \theta_2 L^2$.

   Therefore,

$$\mathrm{E}(y_t) = \mu + \theta(L)\mathrm{E}(\epsilon_t) = \mu$$

Example: MA(q) Model: $y_t = \epsilon_t + \theta_1\epsilon_{t-1} + \theta_2\epsilon_{t-2} + \cdots + \theta_q\epsilon_{t-q}$

1. **Mean of MA(q) Process:**

$$\mathrm{E}(y_t) = \mathrm{E}(\epsilon_t + \theta_1\epsilon_{t-1} + \theta_2\epsilon_{t-2} + \cdots + \theta_q\epsilon_{t-q}) = 0$$

2. **Autocovariance Function of MA(q) Process:**

$$\gamma(\tau) = \begin{cases} \sigma_\epsilon^2(\theta_0\theta_\tau + \theta_1\theta_{\tau+1} + \cdots + \theta_{q-\tau}\theta_q) = \sigma_\epsilon^2 \sum_{i=0}^{q-\tau} \theta_i\theta_{\tau+i}, & \tau = 1, 2, \cdots, q, \\ 0, & \tau = q+1, q+2, \cdots, \end{cases}$$

where $\theta_0 = 1$.

3. MA(q) process is stationary.

4. **MA(q) +drift:** $y_t = \mu + \epsilon_t + \theta_1\epsilon_{t-1} + \theta_2\epsilon_{t-2} + \cdots + \theta_q\epsilon_{t-q}$

   Mean:

$$y_t = \mu + \theta(L)\epsilon_t,$$

   where $\theta(L) = 1 + \theta_1 L + \theta_2 L^2 + \cdots + \theta_q L^q$.

   Therefore, we have:

$$\mathrm{E}(y_t) = \mu + \theta(L)\mathrm{E}(\epsilon_t) = \mu.$$

1.4 ARMA Model

ARMA (Autoregressive Moving Average，自己回帰移動平均) Process

1. ARMA(p, q)

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots + \theta_q \epsilon_{t-q},$$

which is rewritten as:

$$\phi(L) y_t = \theta(L) \epsilon_t,$$

where $\phi(L) = 1 - \phi_1 L - \phi_2 L^2 - \cdots - \phi_p L^p$ and $\theta(L) = 1 + \theta_1 L + \theta_2 L^2 + \cdots + \theta_q L^q$.

2. Likelihood Function:

   The variance-covariance matrix of Y, denoted by V, has to be computed.

Example: ARMA(1,1) Process: $y_t = \phi_1 y_{t-1} + \epsilon_t + \theta_1 \epsilon_{t-1}$

Obtain the autocorrelation coefficient.

The mean of y_t is to take the expectation on both sides.

$$\mathrm{E}(y_t) = \phi_1 \mathrm{E}(y_{t-1}) + \mathrm{E}(\epsilon_t) + \theta_1 \mathrm{E}(\epsilon_{t-1}),$$

where the second and third terms are zeros.

Therefore, we obtain:

$$\mathrm{E}(y_t) = 0.$$

The autocovariance of y_t is to take the expectation, multiplying $y_{t-\tau}$ on both sides.

$$\mathrm{E}(y_t y_{t-\tau}) = \phi_1 \mathrm{E}(y_{t-1} y_{t-\tau}) + \mathrm{E}(\epsilon_t y_{t-\tau}) + \theta_1 \mathrm{E}(\epsilon_{t-1} y_{t-\tau}).$$

Each term is given by:

$$\mathrm{E}(y_t y_{t-\tau}) = \gamma(\tau), \qquad \mathrm{E}(y_{t-1} y_{t-\tau}) = \gamma(\tau - 1),$$

$$\mathrm{E}(\epsilon_t y_{t-\tau}) = \begin{cases} \sigma_\epsilon^2, & \tau = 0, \\ 0, & \tau = 1, 2, \cdots, \end{cases} \qquad \mathrm{E}(\epsilon_{t-1} y_{t-\tau}) = \begin{cases} (\phi_1 + \theta_1)\sigma_\epsilon^2, & \tau = 0, \\ \sigma_\epsilon^2, & \tau = 1, \\ 0, & \tau = 2, 3, \cdots. \end{cases}$$

Therefore, we obtain;

$$\begin{aligned} \gamma(0) &= \phi_1 \gamma(1) + (1 + \phi_1\theta_1 + \theta_1^2)\sigma_\epsilon^2, \\ \gamma(1) &= \phi_1 \gamma(0) + \theta_1 \sigma_\epsilon^2, \\ \gamma(\tau) &= \phi_1 \gamma(\tau - 1), \qquad \tau = 2, 3, \cdots. \end{aligned}$$

From the first two equations, $\gamma(0)$ and $\gamma(1)$ are computed by:

$$\begin{pmatrix} 1 & -\phi_1 \\ -\phi_1 & 1 \end{pmatrix} \begin{pmatrix} \gamma(0) \\ \gamma(1) \end{pmatrix} = \sigma_\epsilon^2 \begin{pmatrix} 1 + \phi_1\theta_1 + \theta_1^2 \\ \theta_1 \end{pmatrix}$$

$$\begin{pmatrix} \gamma(0) \\ \gamma(1) \end{pmatrix} = \sigma_\epsilon^2 \begin{pmatrix} 1 & -\phi_1 \\ -\phi_1 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 1 + \phi_1\theta_1 + \theta_1^2 \\ \theta_1 \end{pmatrix}$$

$$= \frac{\sigma_\epsilon^2}{1 - \phi_1^2} \begin{pmatrix} 1 & \phi_1 \\ \phi_1 & 1 \end{pmatrix} \begin{pmatrix} 1 + \phi_1\theta_1 + \theta_1^2 \\ \theta_1 \end{pmatrix} = \frac{\sigma_\epsilon^2}{1 - \phi_1^2} \begin{pmatrix} 1 + 2\phi_1\theta_1 + \theta_1^2 \\ (1 + \phi_1\theta_1)(\phi_1 + \theta_1) \end{pmatrix}.$$

Thus, the initial value of the autocorrelation coefficient is given by:

$$\rho(1) = \frac{(1 + \phi_1\theta_1)(\phi_1 + \theta_1)}{1 + 2\phi_1\theta_1 + \theta_1^2}.$$

We have:

$$\rho(\tau) = \phi_1\rho(\tau - 1).$$

ARMA(p, q) +drift:

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots \phi_p y_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots + \theta_q \epsilon_{t-q}.$$

Mean of ARMA(p, q) Process: $\phi(L) y_t = \mu + \theta(L)\epsilon_t$,

where $\phi(L) = 1 - \phi_1 L - \phi_2 L^2 - \cdots - \phi_p L^p$ and $\theta(L) = 1 + \theta_1 L + \theta_2 L^2 + \cdots + \theta_q L^q$.

$$y_t = \phi(L)^{-1}\mu + \phi(L)^{-1}\theta(L)\epsilon_t.$$

Therefore,

$$\mathrm{E}(y_t) = \phi(L)^{-1}\mu + \phi(L)^{-1}\theta(L)\mathrm{E}(\epsilon_t) = \phi(1)^{-1}\mu = \frac{\mu}{1 - \phi_1 - \phi_2 - \cdots - \phi_p}.$$

1.5 ARIMA Model

Autoregressive Integrated Moving Average (ARIMA，自己回帰和分移動平均) Model

ARIMA(p, d, q) Process

$$\phi(L)\Delta^d y_t = \theta(L)\epsilon_t,$$

where $\Delta^d y_t = \Delta^{d-1}(1-L)y_t = \Delta^{d-1}y_t - \Delta^{d-1}y_{t-1} = (1-L)^d y_t$ for $d = 1, 2, \cdots$, and $\Delta^0 y_t = y_t$.

1.6 SARIMA Model

Seasonal ARIMA (SARIMA) Process:

1. SARIMA(p, d, q)

$$\phi(L)\Delta^d \Delta_s y_t = \theta(L)\epsilon_t,$$

where

$$\Delta_s y_t = (1 - L^s) y_t = y_t - y_{t-s}.$$

$s = 4$ when y_t denotes quarterly date and $s = 12$ when y_t represents monthly data.

1.7 Optimal Prediction

1. AR(p) Process: $y_t = \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + \epsilon_t$

   (a) Define:

   $$\mathrm{E}(y_{t+k}|Y_t) = y_{t+k|t},$$

   where Y_t denotes all the information available at time t.

   Taking the conditional expectation of $y_{t+k} = \phi_1 y_{t+k-1} + \cdots + \phi_p y_{t+k-p} + \epsilon_{t+k}$ on both sides,

   $$y_{t+k|t} = \phi_1 y_{t+k-1|t} + \cdots + \phi_p y_{t+k-p|t},$$

where $y_{s|t} = y_s$ for $s \leq t$.

(b) Optimal prediction is given by solving the above differential equation.

2. MA(q) Process: $y_t = \epsilon_t + \theta_1 \epsilon_{t-1} + \cdots + \theta_q \epsilon_{t-q}$

(a) Let $\hat{\epsilon}_T$, $\hat{\epsilon}_{T-1}$, $\cdots$, $\hat{\epsilon}_1$ be the estimated errors.

(b) $y_{t+k} = \epsilon_{t+k} + \theta_1 \epsilon_{t+k-1} + \cdots + \theta_q \epsilon_{t+k-q}$

(c) Therefore,

$$y_{t+k|t} = \epsilon_{t+k|t} + \theta_1 \epsilon_{t+k-1|t} + \cdots + \theta_q \epsilon_{t+k-q|t},$$

where $\epsilon_{s|t} = 0$ for $s > t$ and $\epsilon_{s|t} = \hat{\epsilon}_s$ for $s \leq t$.

3. ARMA(p, q) Process: $y_t = \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} + \cdots + \theta_q \epsilon_{t-q}$

(a) $y_{t+k} = \phi_1 y_{t+k-1} + \cdots + \phi_p y_{t+k-p} + \epsilon_{t+k} + \theta_1 \epsilon_{t+k-1} + \cdots + \theta_q \epsilon_{t+k-q}$

(b) Optimal prediction is:

$$y_{t+k|t} = \phi_1 y_{t+k-1|t} + \cdots + \phi_p y_{t+k-p|t} + \epsilon_{t+k|t} + \theta_1 \epsilon_{t+k-1|t} + \cdots + \theta_q \epsilon_{t+k-q|t},$$

where $y_{s|t} = y_s$ and $\epsilon_{s|t} = \hat{\epsilon}_s$ for $s \leq t$, and $\epsilon_{s|t} = 0$ for $s > t$.

1.8 Identification

1. Based on AIC or SBIC given d, s, we obtain p, q.

   (a) AIC (Akaike's Information Criterion)

   $$\text{AIC} = -2\log(\text{likelihood}) + 2k,$$

   where $k = p + q$, which is the number of parameters estimated.

   (b) SBIC (Shwarz's Bayesian Information Criterion)

   $$\text{SBIC} = -2\log(\text{likelihood}) + k\log T,$$

where T denotes the number of observations.

2. From the sample autocorrelation coefficient function $\hat{\rho}(k)$ and the partial autocorrelation coefficient function $\hat{\phi}_{k,k}$ for $k = 1, 2, \cdots$, we obtain p, d, q, s.

	AR(p) Process	MA(q) Process
Autocorrelation Function	Gradually decreasing	$\rho(k) = 0$, $k = q+1, q+2, \cdots$
Partial Autocorrelation Function	$\phi(k,k) = 0$, $k = p+1, p+2, \cdots$	Gradually decreasing

   (a) Compute $\Delta_s y_t$ to remove seasonality.

Compute the autocovariance functions of $\Delta_s y_t$.

If the autocovariance functions have period s, we take $(1 - L^s)$, again.

(b) Determine the order of difference.

Compute the partial autocovariance functions every time.

If the autocovariance functions decrease as τ is large, go to the next step.

(c) Determine the order of AR terms (i.e., p).

Compute the partial autocovariance functions every time.

The partial autocovariance functions are close to zero after some τ, go to the next step.

(d) Determine the order of MA terms (i.e., q).

Compute the autocovariance functions every time.

If the autocovariance functions are randomly around zero, end of the procedure.

1.9 Example of SARIMA using Consumption Data

Construct SARIMA model using monthly and seasonally unadjusted consumption expenditure data and STATA12.

Estimation Period: Jan., 1970 — Dec., 2012 ($T = 516$)

```
. gen time=_n

. tsset time
        time variable:  time, 1 to 516
                delta:  1 unit

. corrgram expend
```

```
                                                      -1       0       1 -1       0       1
 LAG       AC       PAC      Q      Prob>Q  [Autocorrelation]  [Partial Autocor]
-------------------------------------------------------------------------------------------
1        0.8488   0.8499   373.88  0.0000           |------            |------
2        0.8231   0.3858   726.18  0.0000           |------            |---
3        0.8716   0.5266     1122  0.0000           |------            |----
4        0.8706   0.4025   1517.6  0.0000           |------            |---
5        0.8498   0.3447   1895.3  0.0000           |------            |--
6        0.8085   0.0074   2237.9  0.0000           |------            |
7        0.8378   0.1528   2606.5  0.0000           |------            |-
8        0.8460   0.1467     2983  0.0000           |------            |-
9        0.8342   0.3006   3349.9  0.0000           |------            |--
10       0.7735  -0.1518     3666  0.0000           |------          -|
11       0.7852  -0.1185   3992.3  0.0000           |------            |
12       0.9234   0.9442   4444.5  0.0000           |-------           |-------
13       0.7754  -0.5486   4764.1  0.0000           |------        ----|
14       0.7482  -0.3248   5062.1  0.0000           |-----           --|
15       0.7963  -0.2392   5400.5  0.0000           |------          -|


. gen dexp=expend-l.expend
(1 missing value generated)
```

```
. corrgram dexp

                                                                -1       0       1 -1       0       1
 LAG       AC       PAC      Q      Prob>Q  [Autocorrelation]  [Partial Autocor]
-------------------------------------------------------------------------------------------------------
1       -0.4316  -0.4329   96.485  0.0000        ---|                ---|
2       -0.2546  -0.5441   130.13  0.0000         --|               ----|
3        0.1721  -0.4091   145.53  0.0000           |-               ---|
4        0.0667  -0.3459   147.85  0.0000           |                 --|
5        0.0715  -0.0036   150.52  0.0000           |                   |
6       -0.2428  -0.1489   181.36  0.0000          -|                  -|
7        0.0711  -0.1400   184.01  0.0000           |                  -|
8        0.0668  -0.2900   186.36  0.0000           |                 --|
9        0.1704   0.1681   201.64  0.0000           |-                  |-
10      -0.2485   0.1306   234.21  0.0000          -|                   |-
11      -0.4293  -0.9305   331.56  0.0000        ---|           -------|
12       0.9773   0.6768   837.12  0.0000           |-------            |-----
13      -0.4152   0.3778   928.56  0.0000        ---|                   |---
14      -0.2583   0.2688   964.03  0.0000         --|                   |--
15       0.1712   0.0406   979.63  0.0000           |-                  |

. gen sdex=dexp-l12.dexp
(13 missing values generated)
```

```
. corrgram sdex

                                                                -1       0       1 -1       0       1
 LAG       AC       PAC      Q     Prob>Q  [Autocorrelation]  [Partial Autocor]
-------------------------------------------------------------------------------------------------
1       -0.4752  -0.4753   114.28  0.0000        ---|                ---|
2       -0.0244  -0.3235   114.58  0.0000           |                 --|
3        0.1163  -0.0759   121.46  0.0000           |                   |
4       -0.1246  -0.1365   129.37  0.0000           |                  -|
5        0.0341  -0.1016   129.96  0.0000           |                   |
6       -0.0151  -0.1136   130.08  0.0000           |                   |
7       -0.0395  -0.1413   130.88  0.0000           |                  -|
8        0.1123   0.0092   137.35  0.0000           |                   |
9       -0.0664  -0.0100   139.62  0.0000           |                   |
10       0.0168   0.0069   139.76  0.0000           |                   |
11       0.1642   0.2422   153.68  0.0000           |-                  |-
12      -0.3888  -0.2469    231.9  0.0000        ---|                  -|
13       0.2242  -0.1205   257.96  0.0000           |-                  |
14      -0.0147  -0.0941   258.07  0.0000           |                   |
15      -0.0708  -0.0591   260.68  0.0000           |                   |
```

```
. arima sdex, ar(1,2) ma(1)

(setting optimization to BHHH)
Iteration 0:   log likelihood = -5107.4608
Iteration 1:   log likelihood =  -5102.391
Iteration 2:   log likelihood = -5099.9071
Iteration 3:   log likelihood = -5099.4216
Iteration 4:   log likelihood = -5099.2463
(switching optimization to BFGS)
Iteration 5:   log likelihood = -5099.2361
Iteration 6:   log likelihood = -5099.2346
Iteration 7:   log likelihood = -5099.2346
Iteration 8:   log likelihood = -5099.2346

ARIMA regression

Sample:  14 - 516                               Number of obs      =       503
                                                Wald chi2(3)       =    973.93
Log likelihood = -5099.235                      Prob > chi2        =    0.0000
```

sdex	Coef.	OPG Std. Err.	z	P>|z|	[95% Conf.	Interval]
sdex						
_cons	-15.64573	59.17574	-0.26	0.791	-131.628	100.3366
ARMA						
ar						
L1.	.1271774	.0581883	2.19	0.029	.0131304	.2412244
L2.	.1009983	.053626	1.88	0.060	-.0041068	.2061034
ma						
L1.	-.8343264	.0419364	-19.90	0.000	-.9165202	-.7521326
/sigma	6111.128	139.0105	43.96	0.000	5838.673	6383.584

Note: The test of the variance against zero is one sided, and the two-sided confidence interval is truncated at zero.

```
. estat ic

-----------------------------------------------------------------------------
       Model |    Obs    ll(null)   ll(model)     df          AIC         BIC
-------------+---------------------------------------------------------------
           . |    503           .   -5099.235      5     10208.47    10229.57
-----------------------------------------------------------------------------
               Note:  N=Obs used in calculating BIC; see [R] BIC note
```

1.10 ARCH and GARCH Models

Autoregressive Conditional Heteroskedasticity (ARCH)

Generalized Autoregressive Conditional Heteroskedasticity (GARCH)

1. ARCH (p) Model

$$\epsilon_t | \epsilon_{t-1}, \epsilon_{t-2}, \cdots, \epsilon_1 \sim N(0, h_t),$$

where,

$$h_t = \alpha_0 + \alpha_1 \epsilon_{t-1}^2 + \cdots + \alpha_p \epsilon_{t-p}^2.$$

The unconditional variance of ϵ_t is:

$$\sigma_\epsilon^2 = \frac{\alpha_0}{1 - \alpha_1 - \alpha_2 - \cdots - \alpha_p}$$

2. GARCH (p, q) Model

$$\epsilon_t | \epsilon_{t-1}, \epsilon_{t-2}, \cdots, \epsilon_1 \sim N(0, h_t),$$

where

$$h_t = \alpha_0 + \alpha_1 \epsilon_{t-1}^2 + \cdots + \alpha_p \epsilon_{t-p}^2 + \beta_1 h_{t-1} + \cdots + \beta_q h_{t-q}.$$

3. Application to OLS (Case of ARCH(1) Model):

$$y_t = x_t\beta + \epsilon_t, \qquad \epsilon_t|\epsilon_{t-1}, \epsilon_{t-2}, \cdots, \epsilon_1 \sim N(0, \alpha_0 + \alpha_1\epsilon_{t-1}^2).$$

The joint density of $\epsilon_1, \epsilon_2, \cdots, \epsilon_T$ is:

$$\begin{aligned} f(\epsilon_1, \cdots, \epsilon_T) &= f(\epsilon_1)\prod_{t=2}^{T} f(\epsilon_t|\epsilon_{t-1}, \cdots, \epsilon_1) \\ &= (2\pi)^{-1/2}\left(\frac{\alpha_0}{1-\alpha_1}\right)^{-1/2}\exp\left(-\frac{1}{2\alpha_0/(1-\alpha_1)}\epsilon_1^2\right) \\ &\times (2\pi)^{-(T-1)/2}\prod_{t=2}^{T}(\alpha_0 + \alpha_1\epsilon_{t-1}^2)^{-1/2}\exp\left(-\frac{1}{2}\sum_{t=2}^{T}\frac{\epsilon_t^2}{\alpha_0 + \alpha_1\epsilon_{t-1}^2}\right). \end{aligned}$$

The log-likelihood function is:

$$\begin{aligned}
&\log L(\beta, \alpha_0, \alpha_1; y_1, \cdots, y_T) \\
&\quad = -\frac{1}{2}\log(2\pi) - \frac{1}{2}\log\left(\frac{\alpha_0}{1-\alpha_1}\right) - \frac{1}{2\alpha_0/(1-\alpha_1)}(y_1 - x_1\beta)^2 \\
&\qquad -\frac{T-1}{2}\log(2\pi) - \frac{1}{2}\sum_{t=2}^{T}\log\left(\alpha_0 + \alpha_1(y_{t-1} - x_{t-1}\beta)^2\right) \\
&\qquad -\frac{1}{2}\sum_{t=2}^{T}\frac{(y_t - x_t\beta)^2}{\alpha_0 + \alpha_1(y_{t-1} - x_{t-1}\beta)^2}.
\end{aligned}$$

Obtain α_0, α_1 and β such that the log-likelihood function is maximized.

$\alpha_0 > 0$ and $\alpha_1 > 0$ have to be satisfied.

These two conditions are explicitly included, when the model is modified to: $\mathrm{E}(\epsilon_t^2|\epsilon_{t-1}, \epsilon_{t-2}, \cdots, \epsilon_1) = \alpha_0^2 + \alpha_1^2\epsilon_{t-1}^2$.

Testing the ARCH(1) Effect:

(a) Estimate $y_t = x_t\beta + u_t$ by OLS, and compute $\hat{\beta}$ and $\hat{u}_t = y_t - x_t\hat{\beta}$.

(b) Estimate $\hat{u}_t^2 = \alpha_0 + \alpha_1\hat{u}_{t-1}^2$ by OLS. If $\hat{\alpha}_1$ is significant, there is the ARCH(1) effect in the error term.

This test corresponds to LM test.

2 Vector Autoregressive (VAR) Model – Causality, Impulse Response Function and etc

Vector Autoregressive Process:

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \epsilon_t,$$

where

$$y_t : \ k \times 1, \qquad \mu : \ k \times 1, \qquad \epsilon_t : \ k \times 1, \qquad \phi_i : \ k \times k.$$

Rewriting the above equation,

$$\phi(L)y_t = \mu + \epsilon_t,$$

where $\phi(L) = I_k - \phi_1 L - \phi_2 L^2 - \cdots - \phi_p L^p$.

VAR(1) Model:

$$y_t = \phi_1 y_{t-1} + \epsilon_t, \qquad \text{i.e.,} \qquad (I_k - \phi_1 L) y_t = \epsilon_t.$$

When y_t is stationary, we obtain:

$$\begin{aligned} y_t &= (I_k - \phi_1 L)^{-1} \epsilon_t \\ &= (I_k + \phi_1 L + \phi_1^2 L^2 + \phi_1^3 L^3 + \cdots) \epsilon_t \\ &= \epsilon_t + \phi_1 \epsilon_{t-1} + \phi_1^2 \epsilon_{t-2} + \phi_1^3 \epsilon_{t-3} + \cdots \end{aligned}$$

VAR(1)=VMA(∞)

VAR(2) Model:

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \epsilon_t, \qquad \text{i.e.,} \qquad (I_k - \phi_1 L - \phi_2 L^2) y_{t-1} = \epsilon_t.$$

When y_t is stationary, we obtain:

$$\begin{aligned} y_{t-1} &= (I_k - \phi_1 L - \phi_2 L^2)^{-1} \epsilon_t \\ &= \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots \end{aligned}$$

VAR(2)=VMA(∞)

VAR(p) Model:

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \epsilon_t,$$

i.e.,

$$(I_k - \phi_1 L - \phi_2 L^2 - \cdots - \phi_p L^p) y_{t-1} = \epsilon_t.$$

When y_t is stationary, we obtain:

$$\begin{aligned} y_t &= (I_k - \phi_1 L - \phi_2 L^2 - \cdots - \phi_p L^p)^{-1} \epsilon_t \\ &= \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots \end{aligned}$$

VAR(p)=VMA(∞)

2.1 Autocovariance Matrix and Autocorrelation Matrix

Let y_t be a $k \times 1$ vector.

Autocovariance Function Matrix:

$$\Gamma(\tau) = \mathrm{E}((y_t - \mu)(y_{t-\tau} - \mu)'), \qquad \tau = 0, 1, 2, \cdots,$$

where $\mathrm{E}(y_t) = \mu$. $\Gamma(\tau)$ is a $k \times k$ matrix.

$$\Gamma(\tau) = \Gamma(-\tau)'$$

Autocorrelation Function Matrix:

$$\rho(\tau) = D^{-1/2}\Gamma(\tau)D^{-1/2},$$

where the (i, j)th element of D is given by $\gamma_{ii}(\tau) = \mathrm{V}(y_{it})$ for $i = j$ and zero otherwise.

$$\rho(\tau) = \rho(-\tau)'$$

2.2 Granger Cuasality Test (グレンジャー因果性テスト)

Consider a bivariate case.

Unrestricted Model (Sum of Squared Residuals, denoted by SSR_1):

$$\begin{pmatrix} y_{1,t} \\ y_{2,t} \end{pmatrix} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} + \begin{pmatrix} \phi_{11,1} & \phi_{12,1} \\ \phi_{21,1} & \phi_{22,1} \end{pmatrix} \begin{pmatrix} y_{1,t-1} \\ y_{2,t-1} \end{pmatrix} + \cdots + \begin{pmatrix} \phi_{11,p} & \phi_{12,p} \\ \phi_{21,p} & \phi_{22,p} \end{pmatrix} \begin{pmatrix} y_{1,t-p} \\ y_{2,t-p} \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \end{pmatrix}$$

$H_0 : \phi_{12,1} = \phi_{12,2} = \cdots = \phi_{12,p} = 0$

When H_0 is correct, we say there is no causality from y_2 to y_1.

$\Longrightarrow$ Granger Causality Test.

Restricted Model (Sum of Squared Residuals, denoted by SSR_0):

$$\begin{pmatrix} y_{1,t} \\ y_{2,t} \end{pmatrix} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} + \begin{pmatrix} \phi_{11,1} & 0 \\ \phi_{21,1} & \phi_{22,1} \end{pmatrix} \begin{pmatrix} y_{1,t-1} \\ y_{2,t-1} \end{pmatrix} + \cdots + \begin{pmatrix} \phi_{11,p} & 0 \\ \phi_{21,p} & \phi_{22,p} \end{pmatrix} \begin{pmatrix} y_{1,t-p} \\ y_{2,t-p} \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \end{pmatrix}$$

Asymptotically, we have the following distribution:

$$F = \frac{(\text{SSR}_0 - \text{SSR}_1)/p}{\text{SSR}_1/(T-2p-1)} \sim F(p, T-2p-1),$$

or

$$pF \sim \chi^2(p).$$

In general, we consider testing the Granger causality from y_j to y_i.

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \epsilon_t.$$

$y_t : \ k \times 1, \qquad \mu : \ k \times 1, \qquad \phi_p : \ k \times k, \qquad \epsilon_t : \ k \times 1.$

The null hypothesis is: $H_0 : \ \phi_{ij,1} = \phi_{ij,2} = \cdots = \phi_{ij,p} = 0.$

The alternative hypothesis is: $H_1 : \ \text{not } H_0.$

SSR_0 = Sum of Squared Residuals under H_0

SSR_1 = Sum of Squared Residuals under H_1

Under H_0, the asymptotic distribution is given by:

$$F = \frac{(\text{SSR}_0 - \text{SSR}_1)/p}{\text{SSR}_1/(T - kp - 1)} \sim F(p, T - kp - 1),$$

or

$$pF \sim \chi^2(p).$$

2.3 Impulse Response Function (インパルス応答関数):

$$\frac{\partial y_{i,t+k}}{\partial \epsilon_{j,t}}, \qquad k = 1, 2, \cdots,$$

where $i, j = 1, 2, \cdots, k$.

Example: AR(p) Process:

When y_t is stationary, we obtain:

$$\begin{aligned} y_t &= (I_k - \phi_1 L - \phi_2 L^2 - \ \cdots \ - \phi_p L^p)^{-1} \epsilon_t \\ &= \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \ \cdots \end{aligned}$$

$$\frac{\partial y_{i,t+k}}{\partial \epsilon_{j,t}} = \theta_{ij,k}, \qquad k = 1, 2, \cdots,$$

where $\theta_{ij,k}$ denotes the (i, j)th element of θ_k.

3 Unit Root (単位根) and Cointegration (共和分)

3.1 Unit Root (単位根) Test (Dickey-Fuller (DF) Test)

1. Why is a unit root problem important?

   (a) Economic variables increase over time in general.

   One of the assumptions of OLS is stationarity on y_t and x_t.

   This assumption implies that $\frac{1}{T}X'X$ converges to a fixed matrix as T is large.

That is, asymptotic normality of OLS estimator goes not hold.

(b) In nonstationary time series, the unit root is the most important.

In the case of unit root, OLSE of the first-order autoregressive coefficient is consistent.

OLSE is $\sqrt{T}$-consistent in the case of stationary AR(1) process, but OLSE is T-consistent in the case of nonstationay AR(1) process.

(c) A lot of economic variables increase over time.

It is important to check an economic variable is trend stationary (i.e., $y_t = a_0 + a_1 t + \epsilon_t$) or difference stationary (i.e., $y_t = b_0 + y_{t-1} + \epsilon_t$).

Consider k-step ahead prediction for both cases.

(Trend Stationarity) $\quad y_{t+k|t} = a_0 + a_1(t+k)$

(Difference Stationarity) $\quad y_{t+k|t} = b_0 k + y_t$

2. **The Case of $|\phi_1| < 1$:**

$$y_t = \phi_1 y_{t-1} + \epsilon_t, \qquad \epsilon_t \sim \text{i.i.d. } N(0, \sigma_\epsilon^2), \qquad y_0 = 0, \qquad t = 1, \cdots, T$$

Then, OLSE of ϕ_1 is:

$$\hat{\phi}_1 = \frac{\sum_{t=1}^{T} y_{t-1} y_t}{\sum_{t=1}^{T} y_{t-1}^2}.$$

In the case of $|\phi_1| < 1$,

$$\hat{\phi}_1 = \phi_1 + \frac{\frac{1}{T}\sum_{t=1}^{T} y_{t-1}\epsilon_t}{\frac{1}{T}\sum_{t=1}^{T} y_{t-1}^2} \longrightarrow \phi_1 + \frac{\mathrm{E}(y_{t-1}\epsilon_t)}{\mathrm{E}(y_{t-1}^2)} = \phi_1.$$

Note as follows:

$$\frac{1}{T}\sum_{t=1}^{T} y_{t-1}\epsilon_t \longrightarrow \mathrm{E}(y_{t-1}\epsilon_t) = 0.$$

By the central limit theorem,

$$\frac{\overline{y\epsilon} - \mathrm{E}(\overline{y\epsilon})}{\sqrt{\mathrm{V}(\overline{y\epsilon})}} \longrightarrow N(0,1)$$

where

$$\overline{y\epsilon} = \frac{1}{T}\sum_{t=1}^{T} y_{t-1}\epsilon_t.$$

$$\mathrm{E}(\overline{y\epsilon}) = 0,$$

$$\mathrm{V}(\overline{y\epsilon}) = \mathrm{V}(\frac{1}{T}\sum_{t=1}^{T} y_{t-1}\epsilon_t) = \mathrm{E}\Big((\frac{1}{T}\sum_{t=1}^{T} y_{t-1}\epsilon_t)^2\Big)$$

$$= \frac{1}{T^2}\mathrm{E}\Big(\sum_{t=1}^{T}\sum_{s=1}^{T} y_{t-1}y_{s-1}\epsilon_t\epsilon_s\Big) = \frac{1}{T^2}\mathrm{E}\Big(\sum_{t=1}^{T} y_{t-1}^2\epsilon_t^2\Big) = \frac{1}{T}\sigma^2\gamma(0).$$

Therefore,

$$\frac{\overline{y\epsilon}}{\sqrt{\sigma^2\gamma(0)/T}} = \frac{1}{\sigma_\epsilon\sqrt{\gamma(0)}}\frac{1}{\sqrt{T}}\sum_{t=1}^{T} y_{t-1}\epsilon_t \longrightarrow N(0,1),$$

which is rewritten as:

$$\frac{1}{\sqrt{T}}\sum_{t=1}^{T} y_{t-1}\epsilon_t \longrightarrow N(0, \sigma_\epsilon^2\gamma(0)).$$

Using $\frac{1}{T}\sum_{t=1}^{T} y_{t-1}^2 \longrightarrow \mathrm{E}(y_{t-1}^2) = \gamma(0)$, we have the following asymptotic distribution:

$$\sqrt{T}(\hat{\phi}_1 - \phi_1) = \frac{\frac{1}{\sqrt{T}}\sum_{t=1}^{T} y_{t-1}\epsilon_t}{\frac{1}{T}\sum_{t=1}^{T} y_{t-1}^2} \longrightarrow N\left(0, \frac{\sigma_\epsilon^2}{\gamma(0)}\right) = N\left(0, 1-\phi_1^2\right).$$

Note that $\gamma(0) = \dfrac{\sigma_\epsilon^2}{1-\phi_1^2}$.

3. In the case of $\phi_1 = 1$, as expected, we have:

$$\sqrt{T}(\hat{\phi}_1 - 1) \longrightarrow 0.$$

That is, $\hat{\phi}_1$ has the distribution which converges in probability to $\phi_1 = 1$ (i.e., degenerated distribution).

Is this true?

4. **The Case of $\phi_1 = 1$:** $\Longrightarrow$ Random Walk Process

$y_t = y_{t-1} + \epsilon_t$ with $y_0 = 0$ is written as:

$$y_t = \epsilon_t + \epsilon_{t-1} + \epsilon_{t-2} + \cdots + \epsilon_1.$$

Therefore, we can obtain:

$$y_t \sim N(0, \sigma_\epsilon^2 t).$$

The variance of y_t depends on time t. $\implies$ y_t is nonstationary.

5. Remember that $\hat{\phi}_1 = \phi_1 + \dfrac{\sum y_{t-1}\epsilon_t}{\sum y_{t-1}^2}$.

   (a) First, consider the numerator $\sum y_{t-1}\epsilon_t$.

We have $y_t^2 = (y_{t-1} + \epsilon_t)^2 = y_{t-1}^2 + 2y_{t-1}\epsilon_t + \epsilon_t^2$.

Therefore, we obtain:

$$y_{t-1}\epsilon_t = \frac{1}{2}(y_t^2 - y_{t-1}^2 - \epsilon_t^2).$$

Taking into account $y_0 = 0$, we have:

$$\sum_{t=1}^{T} y_{t-1}\epsilon_t = \frac{1}{2}y_T^2 - \frac{1}{2}\sum_{t=1}^{T}\epsilon_t^2.$$

Divided by $\sigma_\epsilon^2 T$ on both sides, we have the following:

$$\frac{1}{\sigma_\epsilon^2 T}\sum_{t=1}^{T} y_{t-1}\epsilon_t = \frac{1}{2}\left(\frac{y_T}{\sigma_\epsilon\sqrt{T}}\right)^2 - \frac{1}{2\sigma_\epsilon^2}\frac{1}{T}\sum_{t=1}^{T}\epsilon_t^2.$$

From $y_t \sim N(0, \sigma_\epsilon^2 t)$, we obtain the following result:

$$\left(\frac{y_T}{\sigma_\epsilon \sqrt{T}}\right)^2 \sim \chi^2(1).$$

Moreover, the second term is derived from:

$$\frac{1}{T}\sum_{t=1}^{T} \epsilon_t^2 \longrightarrow \sigma_\epsilon^2.$$

Therefore,

$$\frac{1}{\sigma_\epsilon^2 T}\sum_{t=1}^{T} y_{t-1}\epsilon_t = \frac{1}{2}\left(\frac{y_T}{\sigma\sqrt{T}}\right)^2 - \frac{1}{2\sigma_\epsilon^2}\frac{1}{T}\sum_{t=1}^{T}\epsilon_t^2 \longrightarrow \frac{1}{2}(\chi^2(1) - 1).$$

(b) Next, consider $\sum y_{t-1}^2$.

$$\mathrm{E}\left(\sum_{t=1}^{T} y_{t-1}^2\right) = \sum_{t=1}^{T} \mathrm{E}(y_{t-1}^2) = \sum_{t=1}^{T} \sigma_\epsilon^2 (t-1) = \sigma_\epsilon^2 \frac{T(T-1)}{2}.$$

Thus, we obtain the following result:

$$\frac{1}{T^2}\mathrm{E}\left(\sum_{t=1}^{T} y_{t-1}^2\right) \longrightarrow \text{ a fixed value.}$$

Therefore,

$$\frac{1}{T^2}\sum_{t=1}^{T} y_{t-1}^2 \longrightarrow \text{ a distribution.}$$

6. Summarizing the results up to now, $T(\hat{\phi}_1 - \phi_1)$, not $\sqrt{T}(\hat{\phi}_1 - \phi_1)$, has limiting distribution in the case of $\phi_1 = 1$.

$$T(\hat{\phi}_1 - \phi_1) = \frac{(1/T)\sum y_{t-1}\epsilon_t}{(1/T^2)\sum y_{t-1}^2} \longrightarrow \text{ a distribution.}$$

7. Basic Concepts of Random Walk Process:

   (a) Model: $y_t = y_{t-1} + \epsilon_t, \qquad y_0 = 0, \qquad \epsilon_t \sim N(0, 1).$

   Then,

$$y_t = \epsilon_t + \epsilon_{t-1} + \cdots + \epsilon_1.$$

Therefore,

$$y_t \sim N(0, t).$$

$\implies$ Nonstationary Process (i.e., variance depends on time t.)

Difference between y_s and y_t $(s > t)$ is:

$$y_s - y_t = \epsilon_s + \epsilon_{s-1} + \cdots + \epsilon_{t+2} + \epsilon_{t+1}.$$

The distribution of $y_s - y_t$ is:

$$y_s - y_t \sim N(0, s - t).$$

(b) Rewrite as follows:

$$\begin{aligned} y_t &= y_{t-1} + \epsilon_t \\ &= y_{t-1} + e_{1,t} + e_{2,t} + \cdots + e_{N,t}, \end{aligned}$$

where $\epsilon_t = e_{1,t} + e_{2,t} + \cdots + e_{N,t}$.

$e_{1,t}, e_{2,t}, \cdots, e_{N,t}$ are iid with $e_{i,t} \sim N(0, 1/N)$.

That is, suppose that there are N subperiods between time t and time $t + 1$.

The limit when $N \to \infty$ is a **continuous time (連続時間)** process known as **standard Brownian motion** or **Wiener process**.

The value of this process at time t is denoted by $W(t)$.

Definition:

Standard Brownian motion $W(t)$ denotes a continuous-time variable at time t and a stochastic function.

$W(t)$ for $t \in [0, 1]$ satisfies the following:

i. $W(0) = 0$

ii. For any time periods $0 \leq r_1 < r_2 < \cdots < r_k \leq 1$, $W(r_2) - W(r_1)$, $W(r_3) - W(r_2), \cdots, W(r_k) - W(r_{k-1})$ are independently multivariate normal with $W(s) - W(t) \sim N(0, s - t)$ for $s > t$.

iii. $W(t)$ is continuous in t with probability 1.

An example:

$$\sigma W(t) \sim N(0, \sigma^2 t),$$

which denotes the Brownian motion with variance σ^2.

Another example;

$$W(t)^2 \sim t \times \chi^2(1).$$

(c) Assume $\epsilon_t \sim$ iid $(0, \sigma_\epsilon^2)$. Define $X_T(r)$ for $r \in [0,1]$ as follows:

$$X_T(r) = \begin{cases} 0, & 0 \le r < \dfrac{1}{T} \\ \dfrac{\epsilon_1}{T}, & \dfrac{1}{T} \le r < \dfrac{2}{T} \\ \dfrac{\epsilon_1 + \epsilon_2}{T}, & \dfrac{2}{T} \le r < \dfrac{3}{T} \\ \vdots & \vdots \\ \dfrac{\epsilon_1 + \epsilon_2 + \cdots + \epsilon_T}{T}, & r = 1 \end{cases}$$

Let $[Tr]$ be the largest integer which is less than or equal to $T \times r$.

$$X_T(r) \equiv \frac{1}{T} \sum_{t=1}^{[Tr]} \epsilon_t, \qquad \sqrt{T} X_T(r) \longrightarrow N(0, r\sigma_\epsilon^2).$$

Note that

$$\frac{1}{T} \sum_{t=1}^{[Tr]} \epsilon_t = \frac{[Tr]}{T} \frac{1}{[Tr]} \sum_{t=1}^{[Tr]} \epsilon_t,$$

$$\frac{[Tr]}{T} \longrightarrow r, \qquad \frac{1}{\sqrt{[Tr]}} \sum_{t=1}^{[Tr]} \epsilon_t \longrightarrow N(0, \sigma_\epsilon^2),$$

$$\sqrt{T} X_T(r) = \frac{[Tr]}{T} \sqrt{\frac{T}{[Tr]}} \frac{1}{\sqrt{[Tr]}} \sum_{t=1}^{[Tr]} \epsilon_t, \qquad \sqrt{\frac{T}{[Tr]}} \longrightarrow \frac{1}{\sqrt{r}}.$$

Therefore, we obtain:

$$\sqrt{T}X_T(r) \longrightarrow N(0, r\sigma_\epsilon^2).$$

Moreover, we have the following results:

$$\frac{\sqrt{T}\left(X_T(r_2) - X_T(r_1)\right)}{\sigma_\epsilon} \longrightarrow N(0, r_2 - r_1),$$
$$\frac{\sqrt{T}X_T(r)}{\sigma_\epsilon} \longrightarrow W(r)$$

For example, consider:

$$X_T(1) = \frac{1}{T}\sum_{t=1}^{T} \epsilon_t.$$

Then,

$$\frac{\sqrt{T}X_T(1)}{\sigma_\epsilon} = \frac{1}{\sigma_\epsilon \sqrt{T}}\sum_{t=1}^{T} \epsilon_t \longrightarrow W(1) = N(0,1).$$

(d) Consider $y_t = y_{t-1} + \epsilon_t,\ y_0 = 0$ and $\epsilon_t \sim N(0, \sigma_\epsilon^2)$.

$X_T(r)$ is defined as follows:

$$X_T(r) = \begin{cases} 0, & 0 \le r < \frac{1}{T}, \\ \frac{y_1}{T}, & \frac{1}{T} \le r < \frac{2}{T}, \\ \frac{y_2}{T}, & \frac{2}{T} \le r < \frac{3}{T}, \\ \quad \vdots & \quad \vdots \\ \frac{y_{T-1}}{T}, & \frac{T-1}{T} \le r < 1, \\ \frac{y_T}{T}, & r = 1. \end{cases}$$

Define $S_T(r)$ as follows:

$$S_T(r) = \begin{cases} 0, & 0 \le r < \dfrac{1}{T}, \\ \dfrac{y_1^2}{T}, & \dfrac{1}{T} \le r < \dfrac{2}{T}, \\ \dfrac{y_2^2}{T}, & \dfrac{2}{T} \le r < \dfrac{3}{T}, \\ \quad \vdots & \quad \vdots \\ \dfrac{y_{T-1}^2}{T}, & \dfrac{T-1}{T} \le r < 1, \\ \dfrac{y_T^2}{T}, & r = 1. \end{cases}$$

To obtain $\int_0^1 X_T(r)\mathrm{d}r$ and $\int_0^1 S_T(r)\mathrm{d}r$, we compute a sum of rectangulars as follows:

$$\begin{aligned}\int_0^1 X_T(r)\mathrm{d}r &\approx \frac{y_1}{T}\left(\frac{2}{T}-\frac{1}{T}\right)+\frac{y_2}{T}\left(\frac{3}{T}-\frac{2}{T}\right)+\cdots+\frac{y_{T-1}}{T}\left(1-\frac{T-1}{T}\right)\\ &=\frac{y_1}{T^2}+\frac{y_2}{T^2}+\cdots+\frac{y_{T-1}}{T^2}=\frac{1}{T^2}\sum_{t=1}^{T}y_t,\end{aligned}$$

$$\begin{aligned}\int_0^1 S_T(r)\mathrm{d}r &\approx \frac{y_1^2}{T}\left(\frac{2}{T}-\frac{1}{T}\right)+\frac{y_2^2}{T}\left(\frac{3}{T}-\frac{2}{T}\right)+\cdots+\frac{y_{T-1}^2}{T}\left(1-\frac{T-1}{T}\right)\\ &=\frac{y_1^2}{T^2}+\frac{y_2^2}{T^2}+\cdots+\frac{y_{T-1}^2}{T^2}=\frac{1}{T^2}\sum_{t=1}^{T}y_t^2.\end{aligned}$$

We have already known that $\sqrt{T}X_T(r) \longrightarrow \sigma_\epsilon W(r)$.

Therefore,

$$\int_0^1 \sqrt{T}X_T(r)\mathrm{d}r \longrightarrow \sigma_\epsilon \int_0^1 W(r)\mathrm{d}r.$$

That is,

$$\frac{1}{T^{3/2}}\sum_{t=1}^{T} y_t \longrightarrow \sigma_\epsilon \int_0^1 W(r)\mathrm{d}r.$$

From $S_T(r) \equiv \left(\sqrt{T} X_T(r)\right)^2$,

$$S_T(r) \longrightarrow \sigma_\epsilon^2 (W(r))^2 ,$$

which is called the continuous mapping theorem.

(*) **Continuous Mapping Theorem (連続写像定理):**

if $x_T \longrightarrow x$ (convergence in distribution) and $g(\cdot)$ is a continuous function, then $g(x_T) \longrightarrow g(x)$ (convergence in distribution).

Threfore, we have the follwoing result:

$$\frac{1}{T^2}\sum_{t=1}^{T} y_t^2 \longrightarrow \int_0^1 S_T(r)\mathrm{d}r = \sigma_\epsilon^2 \int_0^1 (W(r))^2\,\mathrm{d}r.$$

(e) Decompose $T^{-3/2}\sum_{t=1}^{T} y_{t-1}$ as follows:

$$T^{-3/2}\sum_{t=1}^{T} y_{t-1} = T^{-3/2}(\epsilon_1 + (\epsilon_1 + \epsilon_2) + (\epsilon_1 + \epsilon_2 + \epsilon_3) + \cdots + (\epsilon_1 + \epsilon_2 + \cdots + \epsilon_{T-1}))$$

$$= T^{-3/2}((T-1)\epsilon_1 + (T-2)\epsilon_2 + (T-3)\epsilon_3 + \cdots$$

$$+2\epsilon_{T-2} + \epsilon_{T-1})$$

$$= T^{-3/2}\sum_{t=1}^{T}(T-t)\epsilon_t = T^{-1/2}\sum_{t=1}^{T}\epsilon_t - T^{-3/2}\sum_{t=1}^{T}t\epsilon_t$$

We utilize the following fact:

$$\begin{pmatrix} T^{-1/2}\sum_{t=1}^{T}\epsilon_t \\ T^{-3/2}\sum_{t=1}^{T}t\,\epsilon_t \end{pmatrix} \longrightarrow N\left(\begin{pmatrix}0\\0\end{pmatrix}, \sigma_\epsilon^2\begin{pmatrix}1 & \frac{1}{2}\\ \frac{1}{2} & \frac{1}{3}\end{pmatrix}\right)$$

ϵ_t is stationary. $\Longrightarrow$ Apply CLT to $(1/T)\sum_{t=1}^{T}\epsilon_t$.

$t\epsilon_t/T$ is stationary. $\implies$ Apply CLT to $(1/T)\sum_{t=1}^{T} t\epsilon_t/T$.

Using a matrix form, we ca rewrite as follows:

$$T^{-3/2}\sum_{t=1}^{T} y_{t-1} = (1 \quad -1)\begin{pmatrix} T^{-1/2}\sum_{t=1}^{T}\epsilon_t \\ T^{-3/2}\sum_{t=1}^{T} t\,\epsilon_t \end{pmatrix}.$$

Then, the variance of $T^{-3/2}\sum_{t=1}^{T} y_{t-1}$ is given by:

$$\mathrm{V}\Big(T^{-3/2}\sum_{t=1}^{T} y_{t-1}\Big) = \sigma_\epsilon^2(1 \quad -1)\begin{pmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{3} \end{pmatrix}\begin{pmatrix} 1 \\ -1 \end{pmatrix} = \frac{\sigma_\epsilon^2}{3}.$$

Therefore, $T^{-3/2}\sum_{t=1}^{T} y_{t-1} \sim N(0, \sigma_\epsilon^2/3)$.

We have already known:

$$T^{-3/2}\sum_{t=1}^{T} y_{t-1} \longrightarrow \sigma_\epsilon \int_0^1 W(r)\mathrm{d}r,$$

$$\frac{1}{T}\sum_{t=1}^{T} \epsilon_t \longrightarrow \sigma_\epsilon W(1).$$

That is, the following relationship holds:

$$\sigma_\epsilon \int_0^1 W(r)\mathrm{d}r \approx T^{-3/2}\sum_{t=1}^{T} y_{t-1} = T^{-1/2}\sum_{t=1}^{T} \epsilon_t - T^{-3/2}\sum_{t=1}^{T} t\epsilon_t$$

$$\approx \sigma_\epsilon W(1) - T^{-3/2}\sum_{t=1}^{T} t\epsilon_t$$

Therefore, we obatain the following result:

$$T^{-3/2}\sum_{t=1}^{T} t\epsilon_t \longrightarrow \sigma_\epsilon W(1) - \sigma_\epsilon \int_0^1 W(r)\mathrm{d}r = N(0, \frac{\sigma_\epsilon^2}{3}).$$

(f) **Some Formulas:** Model: $y_t = y_{t-1} + \epsilon_t$.

i. $T^{-1/2}\sum_{t=1}^{T} \epsilon_t \longrightarrow \sigma_\epsilon\, W(1) = N(0, \sigma_\epsilon^2)$

ii. $T^{-1}\sum_{t=1}^{T} y_{t-1}\epsilon_t \longrightarrow \frac{1}{2}\sigma_\epsilon^2\left((W(1))^2 - 1\right) = \frac{1}{2}\sigma_\epsilon^2\left(\chi^2(1) - 1\right)$

Note that we obtain $(W(1))^2 \sim \chi^2(1)$ from $W(1) = N(0, 1)$.

iii. $T^{-3/2}\sum_{t=1}^{T} t\,\epsilon_t \longrightarrow \sigma_\epsilon\, W(1) - \sigma_\epsilon \int_0^1 W(r)\mathrm{d}r = N(0, \frac{\sigma_\epsilon^2}{3})$

iv. $T^{-3/2}\sum_{t=1}^{T} y_{t-1} \longrightarrow \sigma_\epsilon \int_0^1 W(r)\mathrm{d}r = N(0, \frac{\sigma_\epsilon^2}{3})$

v. $T^{-2}\sum_{t=1}^{T} y_{t-1}^2 \longrightarrow \sigma_\epsilon^2 \int_0^1 (W(r))^2\,\mathrm{d}r$

vi. $T^{-5/2}\sum_{t=1}^{T} t\, y_{t-1} \longrightarrow \sigma_\epsilon \int_0^1 r\, W(r)\mathrm{d}r$

vii. $T^{-3}\sum_{t=1}^{T} t\, y_{t-1}^2 \longrightarrow \sigma_\epsilon^2 \int_0^1 r\, (W(r))^2\,\mathrm{d}r$

viii. $T^{-(\nu+1)}\sum_{t=1}^{T} t^\nu \longrightarrow \frac{1}{\nu+1}$ for $\nu = 0, 1, \cdots$.

8. Asymptotic Distribution of AR(1) Model:

(a) $\boldsymbol{H_0 : y_t = y_{t-1} + \epsilon_t}$ **and** $\boldsymbol{H_1 : y_t = \phi_1 y_{t-1} + \epsilon_t}$ **for** $\boldsymbol{|\phi_1| < 1}$

OLSE of ϕ_1, denoted by $\hat{\phi}_1$, is given by:

$$\hat{\phi}_1 = \frac{\sum_{t=1}^{T} y_{t-1} y_t}{\sum_{t=1}^{T} y_{t-1}^2} = \phi_1 + \frac{\sum_{t=1}^{T} y_{t-1} \epsilon_t}{\sum_{t=1}^{T} y_{t-1}^2}$$

Using $\phi_1 = 1$ and some formulas shown above, we obtain:

$$T(\hat{\phi}_1 - 1) = \frac{T^{-1} \sum_{t=1}^{T} y_{t-1} u_t}{T^{-2} \sum_{t=1}^{T} y_{t-1}^2} \longrightarrow \frac{\frac{1}{2}\left((W(1))^2 - 1\right)}{\int_0^1 (W(r))^2 \, \mathrm{d}r}$$

Remember that

$$T^{-1}\sum_{t=1}^{T} y_{t-1}u_t \;\longrightarrow\; \frac{1}{2}\sigma_\epsilon^2\Big((W(1))^2 - 1\Big)$$

and

$$T^{-2}\sum_{t=1}^{T} y_{t-1}^2 \;\longrightarrow\; \sigma_\epsilon^2 \int_0^1 (W(r))^2\,\mathrm{d}r,$$

where $(W(1))^2 = \chi^2(1)$.

We say that $\hat{\phi}_1$ is **super-consistent (超一致性)** or ***T*-consistent**.

Remember that when $|\phi_1| < 1$ we have $\sqrt{T}(\hat{\phi}_1 - \phi_1) \longrightarrow N(0, 1 - \phi_1^2)$, and in this case we say that $\hat{\phi}_1$ is $\sqrt{T}$**-consistent**.

Conventional t test statistic is given by:

$$t_T = \frac{\hat{\phi}_1 - 1}{s_\phi},$$

where

$$s_\phi = \left(s_T^2 \Big/ \sum_{t=1}^{T} y_{t-1}^2 \right)^{1/2} \quad \text{and} \quad s_T^2 = \frac{1}{T-1} \sum_{t=1}^{T} (y_t - \hat{\phi}_1 y_{t-1})^2.$$

Next, consider t statistic.

The t test statistic, denoted by t_T, is represented as follows:

$$t_T = \frac{\hat{\phi}_1 - 1}{s_\phi} = \frac{T(\hat{\phi}_1 - 1)}{T s_\phi}$$

The denominator is:

$$T s_\phi = \left(s_T^2 \Big/ \frac{1}{T^2} \sum_{t=1}^{T} y_{t-1}^2 \right)^{1/2}$$

$$\longrightarrow \left(\sigma_\epsilon^2 \Big/ \left(\sigma_\epsilon^2 \int_0^1 (W(r))^2 \, \mathrm{d}r \right) \right)^{1/2} = \left(\int_0^1 (W(r))^2 \, \mathrm{d}r \right)^{-1/2},$$

where $s^2 \longrightarrow \sigma_\epsilon^2$ is utilized.

Therefore, we have the following asymptotic distribution:

$$t_T = \frac{\hat{\phi}_1 - 1}{s_\phi} \longrightarrow \frac{\frac{1}{2}\left((W(1))^2 - 1\right)}{\int_0^1 (W(r))^2 \, \mathrm{d}r} \Bigg/ \left(\int_0^1 (W(r))^2 \, \mathrm{d}r \right)^{-1/2}$$

$$= \frac{\frac{1}{2}\left((W(1))^2 - 1\right)}{\left(\int_0^1 (W(r))^2 \, \mathrm{d}r \right)^{1/2}}.$$

Therefore, the distribution of the t_T statistic shown above is different from the *t* distribution.

(b) $\boldsymbol{H_0 : y_t = y_{t-1} + \epsilon_t}$ **and** $\boldsymbol{H_1 : y_t = \alpha_0 + \phi_1 y_{t-1} + \epsilon_t}$ **for** $\boldsymbol{|\phi_1| < 1}$

$$\begin{pmatrix} \hat{\alpha}_0 \\ \hat{\phi}_1 \end{pmatrix} = \begin{pmatrix} T & \sum y_{t-1} \\ \sum y_{t-1} & \sum y_{t-1}^2 \end{pmatrix}^{-1} \begin{pmatrix} \sum y_t \\ \sum y_{t-1} y_t \end{pmatrix}$$

$$= \begin{pmatrix} \alpha_0 \\ \phi_1 \end{pmatrix} + \begin{pmatrix} T & \sum y_{t-1} \\ \sum y_{t-1} & \sum y_{t-1}^2 \end{pmatrix}^{-1} \begin{pmatrix} \sum \epsilon_t \\ \sum y_{t-1} \epsilon_t \end{pmatrix}$$

In the true model, $\alpha_0 = 0$ and $\phi_1 = 1$.

$$\begin{pmatrix} \hat{\alpha}_0 \\ \hat{\phi}_1 - 1 \end{pmatrix} = \begin{pmatrix} T & \sum y_{t-1} \\ \sum y_{t-1} & \sum y_{t-1}^2 \end{pmatrix}^{-1} \begin{pmatrix} \sum \epsilon_t \\ \sum y_{t-1}\epsilon_t \end{pmatrix}$$

$$= \begin{pmatrix} O_p(T) & O_p(T^{3/2}) \\ O_p(T^{3/2}) & O_p(T^2) \end{pmatrix}^{-1} \begin{pmatrix} O_p(T^{1/2}) \\ O_p(T) \end{pmatrix}$$

(*) For random variable x and constant k, $x = O_p(k)$ implies that x/k converges in distribution.

To change each element of the matrices to $O_p(1)$, we use the following matrix:

$$\Gamma = \begin{pmatrix} T^{1/2} & 0 \\ 0 & T \end{pmatrix}.$$

Multiplying the above matrix from the left, we obtain the following:

$$\begin{aligned}
\Gamma\begin{pmatrix}\hat{\alpha}_0\\ \hat{\phi}_1-1\end{pmatrix} &= \begin{pmatrix}T^{1/2}\hat{\alpha}_0\\ T(\hat{\phi}_1-1)\end{pmatrix} = \Gamma\begin{pmatrix}O_p(T) & O_p(T^{3/2})\\ O_p(T^{3/2}) & O_p(T^2)\end{pmatrix}^{-1}\Gamma\Gamma^{-1}\begin{pmatrix}O_p(T^{1/2})\\ O_p(T)\end{pmatrix}\\
&= \left(\Gamma^{-1}\begin{pmatrix}O_p(T) & O_p(T^{3/2})\\ O_p(T^{3/2}) & O_p(T^2)\end{pmatrix}\Gamma^{-1}\right)^{-1}\Gamma^{-1}\begin{pmatrix}O_p(T^{1/2})\\ O_p(T)\end{pmatrix}\\
&= \left(\Gamma^{-1}\begin{pmatrix}T & \sum y_{t-1}\\ \sum y_{t-1} & \sum y_{t-1}^2\end{pmatrix}\Gamma^{-1}\right)^{-1}\Gamma^{-1}\begin{pmatrix}\sum \epsilon_t\\ \sum y_{t-1}\epsilon_t\end{pmatrix}\\
&= \begin{pmatrix}1 & T^{-3/2}\sum y_{t-1}\\ T^{-3/2}\sum y_{t-1} & T^{-2}\sum y_{t-1}^2\end{pmatrix}^{-1}\begin{pmatrix}T^{-1/2}\sum \epsilon_t\\ T^{-1}\sum y_{t-1}\epsilon_t\end{pmatrix}.
\end{aligned}$$

Each matrix converges in distribution as follows:

$$\begin{pmatrix} 1 & T^{-3/2}\sum y_{t-1} \\ T^{-3/2}\sum y_{t-1} & T^{-2}\sum y_{t-1}^2 \end{pmatrix} \longrightarrow \begin{pmatrix} 1 & \sigma_\epsilon \int_0^1 W(r)\mathrm{d}r \\ \sigma_\epsilon \int_0^1 W(r)\mathrm{d}r & \sigma_\epsilon^2 \int_0^1 (W(r))^2\,\mathrm{d}r \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 \\ 0 & \sigma_\epsilon \end{pmatrix} \begin{pmatrix} 1 & \int_0^1 W(r)\mathrm{d}r \\ \int_0^1 W(r)\mathrm{d}r & \int_0^1 (W(r))^2\,\mathrm{d}r \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & \sigma_\epsilon \end{pmatrix},$$

$$\begin{pmatrix} T^{-1/2}\sum \epsilon_t \\ T^{-1}\sum y_{t-1}\epsilon_t \end{pmatrix} \longrightarrow \begin{pmatrix} \sigma_\epsilon W(1) \\ \frac{1}{2}\sigma_\epsilon^2\left((W(1))^2 - 1\right) \end{pmatrix} = \sigma_\epsilon \begin{pmatrix} 1 & 0 \\ 0 & \sigma_\epsilon \end{pmatrix} \begin{pmatrix} W(1) \\ \frac{1}{2}\left((W(1))^2 - 1\right) \end{pmatrix}.$$

Therefore,

$$\begin{pmatrix} T^{1/2}\hat{\alpha}_0 \\ T(\hat{\phi}_1 - 1) \end{pmatrix} \longrightarrow \left(\begin{pmatrix} 1 & 0 \\ 0 & \sigma_\epsilon \end{pmatrix} \begin{pmatrix} 1 & \int_0^1 W(r)\mathrm{d}r \\ \int_0^1 W(r)\mathrm{d}r & \int_0^1 (W(r))^2\,\mathrm{d}r \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & \sigma_\epsilon \end{pmatrix} \right)^{-1}$$
$$\times \sigma_\epsilon \begin{pmatrix} 1 & 0 \\ 0 & \sigma_\epsilon \end{pmatrix} \begin{pmatrix} W(1) \\ \frac{1}{2}\left((W(1))^2 - 1\right) \end{pmatrix}.$$

Finally, $T(\hat{\phi}_1 - 1)$ converges to the following distribution:

$$T(\hat{\phi}_1 - 1) \longrightarrow \frac{\frac{1}{2}\left((W(1))^2 - 1\right) - W(1)\int_0^1 W(r)\mathrm{d}r}{\int_0^1 (W(r))^2\,\mathrm{d}r - \left(\int_0^1 W(r)\mathrm{d}r\right)^2}.$$

The t test statistic is:

$$t_T = \frac{\hat{\phi}_1 - 1}{\left(s_\phi^2\right)^{1/2}} = \frac{T(\hat{\phi}_1 - 1)}{\left(T^2 s_\phi^2\right)^{1/2}},$$

where

$$s_\phi^2 = s_T^2 \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} T & \sum y_{t-1} \\ \sum y_{t-1} & \sum y_{t-1}^2 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

$$s_T^2 = \frac{1}{T-2} \sum_{t=1}^{T} (y_t - \hat{\alpha}_0 - \hat{\phi}_1 y_{t-1})^2.$$

The denominator $T^2 s_\phi^2$ converges in distribution as follows:

$$T^2 s_\phi^2 \longrightarrow \sigma_\epsilon^2 \begin{pmatrix} 0 & 1 \end{pmatrix} \left(\begin{pmatrix} 1 & 0 \\ 0 & \sigma_\epsilon \end{pmatrix} \begin{pmatrix} 1 & \int_0^1 W(r)\mathrm{d}r \\ \int_0^1 W(r)\mathrm{d}r & \int_0^1 (W(r))^2\,\mathrm{d}r \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & \sigma_\epsilon \end{pmatrix} \right)^{-1} \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$= \frac{1}{\int_0^1 (W(r))^2\,\mathrm{d}r - \left(\int_0^1 W(r)\mathrm{d}r \right)^2}$$

Thus, the t test statistic converges to the following distribution:

$$t_T \longrightarrow \frac{\frac{1}{2}\left((W(1))^2 - 1\right) - W(1)\int_0^1 W(r)\mathrm{d}r}{\left(\int_0^1 (W(r))^2\,\mathrm{d}r - \left(\int_0^1 W(r)\mathrm{d}r\right)^2\right)^{1/2}}.$$

(c) $\boldsymbol{H_0 : y_t = \alpha_0 + y_{t-1} + \epsilon_t}$ **and** $\boldsymbol{H_1 : y_t = \alpha_0 + \phi_1 y_{t-1} + \epsilon_t}$ **for** $\boldsymbol{|\phi_1|} < 1$

The model is written as follows:

$$y_t = y_0 + \alpha_0 t + (\epsilon_1 + \epsilon_2 + \cdots + \epsilon_t)$$

$$= y_0 + \alpha_0 t + u_t,$$

where $u_t = \epsilon_1 + \epsilon_2 + \cdots + \epsilon_t$.

○ For $\sum_{t=1}^{T} y_{t-1}$,

$$\begin{aligned}\sum_{t=1}^{T} y_{t-1} &= \sum_{t=1}^{T} y_0 + \sum_{t=1}^{T} \alpha_0(t-1) + \sum_{t=1}^{T} u_{t-1} \\ &= O_p(T) + O_p(T^2) + O_p(T^{3/2}).\end{aligned}$$

Therefore, we obtain:

$$T^{-2} \sum_{t=1}^{T} y_{t-1} \longrightarrow \frac{\alpha_0}{2}.$$

○ For $\sum_{t=1}^{T} y_{t-1}^2$,

$$\sum_{t=1}^{T} y_{t-1}^2 = \sum_{t=1}^{T} (y_0 + \alpha_0(t-1) + u_{t-1})^2$$

$$= \sum_{t=1}^{T} y_0^2 + \sum_{t=1}^{T} \alpha_0^2(t-1)^2 + \sum_{t=1}^{T} u_{t-1}^2 + \sum_{t=1}^{T} 2y_0\alpha_0(t-1) + \sum_{t=1}^{T} 2y_0 u_{t-1} + \sum_{t=1}^{T} 2\alpha_0(t-1)u_{t-1}$$

$$= O_p(T) + O_p(T^3) + O_p(T^2) + O_p(T^2) + O_p(T^{3/2}) + O_p(T^{5/2})$$

Therefore, we have:

$$T^{-3} \sum_{t=1}^{T} y_{t-1}^2 \longrightarrow \frac{\alpha_0^2}{3}$$

○ For $\sum_{t=1}^{T} y_{t-1}\epsilon_t$,

$$\begin{aligned}\sum_{t=1}^{T} y_{t-1}\epsilon_t &= \sum_{t=1}^{T} (y_0 + \alpha_0(t-1) + u_{t-1})\,\epsilon_t \\ &= \sum_{t=1}^{T} y_0\epsilon_t + \sum_{t=1}^{T} \alpha_0(t-1)\epsilon_t + \sum_{t=1}^{T} u_{t-1}\epsilon_t \\ &= O_p(T^{1/2}) + O_p(T^{3/2}) + O_p(T).\end{aligned}$$

Therefore, we have:

$$T^{-3/2} \sum_{t=1}^{T} y_{t-1}\epsilon_t \longrightarrow N(0, \frac{\alpha_0^2}{3}\sigma^2\epsilon).$$

Therefore, OLSE is:

$$\begin{pmatrix} \hat{\alpha}_0 - \alpha_0 \\ \hat{\phi}_1 - 1 \end{pmatrix} = \begin{pmatrix} T & \sum y_{t-1} \\ \sum y_{t-1} & \sum y_{t-1}^2 \end{pmatrix}^{-1} \begin{pmatrix} \sum \epsilon_t \\ \sum y_{t-1}\epsilon_t \end{pmatrix}$$
$$= \begin{pmatrix} O_p(T) & O_p(T^2) \\ O_p(T^2) & O_p(T^3) \end{pmatrix}^{-1} \begin{pmatrix} O_p(T^{1/2}) \\ O_p(T^{3/2}) \end{pmatrix}.$$

Set:

$$\Gamma = \begin{pmatrix} T^{1/2} & 0 \\ 0 & T^{3/2} \end{pmatrix}.$$

Multiplying Γ from the left,

$$\begin{pmatrix} T^{1/2}(\hat{\alpha}_0 - \alpha_0) \\ T^{3/2}(\hat{\phi}_1 - 1) \end{pmatrix} \longrightarrow N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \sigma_\epsilon^2 \begin{pmatrix} 1 & \frac{\alpha_0}{2} \\ \frac{\alpha_0}{2} & \frac{\alpha_0^2}{3} \end{pmatrix} \right).$$

(d) $\boldsymbol{H_0 : y_t = \alpha_0 + y_{t-1} + \epsilon_t}$ **and**

$\boldsymbol{H_1 : y_t = \alpha_0 + \alpha_1 t + \phi_1 y_{t-1} + \epsilon_t}$ **for** $\boldsymbol{|\phi_1| < 1}$

(abbr.)

9. The distributions of the *t* statistic: $\dfrac{\hat{\phi}_1 - 1}{s_{\phi}}$

***t* Distribution**

T	0.010	0.025	0.050	0.100	0.900	0.950	0.975	0.990
25	−2.49	−2.06	−1.71	−1.32	1.32	1.71	2.06	2.49
50	−2.40	−2.01	−1.68	−1.30	1.30	1.68	2.01	2.40
100	−2.36	−1.98	−1.66	−1.29	1.29	1.66	1.98	2.36
250	−2.34	−1.97	−1.65	−1.28	1.28	1.65	1.97	2.34
500	−2.33	−1.96	−1.65	−1.28	1.28	1.65	1.96	2.33
∞	−2.33	−1.96	−1.64	−1.28	1.28	1.64	1.96	2.33

(a) $H_0: y_t = y_{t-1} + \epsilon_t$

$H_1: y_t = \phi_1 y_{t-1} + \epsilon_t$ **for** $\phi_1 < 1$ **or** $-1 < \phi_1$

T	0.010	0.025	0.050	0.100	0.900	0.950	0.975	0.990
25	−2.66	−2.26	−1.95	−1.60	0.92	1.33	1.70	2.16
50	−2.62	−2.25	−1.95	−1.61	0.91	1.31	1.66	2.08
100	−2.60	−2.24	−1.95	−1.61	0.90	1.29	1.64	2.03
250	−2.58	−2.23	−1.95	−1.62	0.89	1.29	1.63	2.01
500	−2.58	−2.23	−1.95	−1.62	0.89	1.28	1.62	2.00
∞	−2.58	−2.23	−1.95	−1.62	0.89	1.28	1.62	2.00

(b) $H_0: y_t = y_{t-1} + \epsilon_t$

$H_1: y_t = \alpha_0 + \phi_1 y_{t-1} + \epsilon_t$ **for** $\phi_1 < 1$ **or** $-1 < \phi_1$

T	0.010	0.025	0.050	0.100	0.900	0.950	0.975	0.990
25	−3.75	−3.33	−3.00	−2.63	−0.37	0.00	0.34	0.72
50	−3.58	−3.22	−2.93	−2.60	−0.40	−0.03	0.29	0.66
100	−3.51	−3.17	−2.89	−2.58	−0.42	−0.05	0.26	0.63
250	−3.46	−3.14	−2.88	−2.57	−0.42	−0.06	0.24	0.62
500	−3.44	−3.13	−2.87	−2.57	−0.43	−0.07	0.24	0.61
∞	−3.43	−3.12	−2.86	−2.57	−0.44	−0.07	0.23	0.60

(d) $H_0: y_t = \alpha_0 + y_{t-1} + \epsilon_t$

$H_1: y_t = \alpha_0 + \alpha_1 t + \phi_1 y_{t-1} + \epsilon_t$ **for** $\phi_1 < 1$ **or** $-1 < \phi_1$

T	0.010	0.025	0.050	0.100	0.900	0.950	0.975	0.990
25	−4.38	−3.95	−3.60	−3.24	−1.14	−0.80	−0.50	−0.15
50	−4.15	−3.80	−3.50	−3.18	−1.19	−0.87	−0.58	−0.24
100	−4.04	−3.73	−3.45	−3.15	−1.22	−0.90	−0.62	−0.28
250	−3.99	−3.69	−3.43	−3.13	−1.23	−0.92	−0.64	−0.31
500	−3.98	−3.68	−3.42	−3.13	−1.24	−0.93	−0.65	−0.32
∞	−3.96	−3.66	−3.41	−3.12	−1.25	−0.94	−0.66	−0.33

3.2 Serially Correlated Errors

Consider the case where the error term is serially correlated.

3.2.1 Augmented Dickey-Fuller (ADF) Test

Consider the following AR(p) model:

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \epsilon_t, \qquad \epsilon_t \sim \text{iid}(0, \sigma_\epsilon^2),$$

which is rewritten as:

$$\phi(L) y_t = \epsilon_t.$$

When the above model has a unit root, we have $\phi(1) = 0$, i.e., $\phi_1 + \phi_2 + \cdots + \phi_p = 1$.

The above AR(p) model is written as:

$$y_t = \rho y_{t-1} + \delta_1 \Delta y_{t-1} + \delta_2 \Delta y_{t-2} + \cdots + + \delta_{p-1} \Delta y_{t-p+1} + \epsilon_t,$$

where $\rho = \phi_1 + \phi_2 + \cdots + \phi_p$ and $\delta_j = -(\phi_{j+1} + \phi_{j+2} + \cdots + \phi_p)$.

The null and alternative hypotheses are:

$$H_0 : \rho = 1 \text{ (Unit root)},$$

$$H_1 : \rho < 1 \text{ (Stationary)}.$$

Use the t test, where we have the same asymptotic distributions.

We can utilize the same tables as before.

Choose p by AIC or SBIC.

Use $N(0,1)$ to test $H_0: \delta_j = 0$ against $H_1: \delta_j \neq 0$ for $j = 1, 2, \cdots, p-1$.

Reference

Kurozumi (2008) "Economic Time Series Analysis and Unit Root Tests: Development and Perspective," *Japan Statistical Society*, Vol.38, Series J, No.1, pp.39 – 57.

Download the above paper from:

http://ci.nii.ac.jp/vol_issue/nels/AA11989749/ISS0000426576_ja.html

3.2.2 Phillips-Perron (PP) Test

The model is given by:

$$y_t = \phi_1 y_{t-1} + u_t, \qquad u_t = \sum_{s=0}^{\infty} \psi_s \epsilon_{t-s}, \qquad \epsilon_t \sim \text{ iid}(0, \sigma_\epsilon^2),$$

where $\psi_0 = 0$ and $\sum_{s=0}^{\infty} s|\psi_s| < \infty$.

Note that the errors are serially correlated and heteroskedastic.

The autocovariance function of u_t is:

$$\gamma(\tau) = \text{E}(u_t u_{t-\tau}) = \sigma_\epsilon^2 \sum_{s=0}^{\infty} \psi_s \psi_{s+\tau}, \qquad \tau = 0, 1, 2, \cdots.$$

Define the long-run variance of u_t as:

$$\lambda^2 = \lim_{T\to\infty} \frac{1}{T}\mathrm{E}((\sum_{t=1}^{T} u_t)^2) = \sum_{\tau=-\infty}^{\infty} \gamma(\tau) = \gamma(0) + 2\sum_{\tau=1}^{\infty} \gamma(\tau) = \sigma_\epsilon^2(\sum_{j=0}^{\infty} \psi_j)^2.$$

The PP test statistic $\tilde{t}_T$ is:

$$\tilde{t}_T = \left(\frac{\gamma(0)}{\lambda^2}\right)^{1/2} t_T - \frac{1}{2\lambda}\frac{T s_\phi}{s_T}(\lambda^2 - \gamma(0)),$$

where

t_T denotes the t statistic of $\hat{\phi}_1$, s_ϕ is the standard error of $\hat{\phi}_1$, and

$s_T^2 = \dfrac{1}{T-1}\sum_{t=1}^{T}(y_t - \hat{\phi}_1 y_{t-1})^2.$

Estimate λ by:

$$\hat{\lambda} = \hat{\gamma}(0) + 2\sum_{\tau=1}^{q} k_1(\frac{\tau}{q+1})\hat{\gamma}(\tau),$$

which is called **Newey-West estimator**, where $k_1(x) = 1-|x|$ for $x \leq 1$ and $k_1(x) = 0$ for $x > 1$, which is called **Bartlett kernel**, or

$$\hat{\lambda} = \hat{\gamma}(0) + 2\sum_{\tau=1}^{q} k_2(\frac{\tau}{q+1})\hat{\gamma}(\tau),$$

where $k_2(x) = 1 - 6x^2 + 6x^3$ for $0 \leq x \leq \frac{1}{2}$, $k_2(x) = 2(1-x)^3$ for $\frac{1}{2} \leq x \leq 1$ and $k_2(x) = 0$ for $x > 1$, which is called **Parzen kernel**, or

$$\hat{\lambda} = \frac{T}{T-1}\left(\hat{\gamma}(0) + \sum_{\tau=1}^{T-1} k_3(\frac{\tau}{q+1})\hat{\gamma}(\tau)\right),$$

where $k_3(x) = \frac{3}{(6\pi x/5)^2}\left(\frac{\sin(6\pi x/5)}{6\pi x/5} - \cos(6\pi x/5)\right)$, which is called the **second-order spectrum kernel**.

We need to choose the bandwidth q.

Use the same statistical tables as before to test $H_0 : \phi_1 = 1$ against $H_1 : \phi_1 < 1$.

Some Formulas:

For proof, we use following formulas.

Let $u_t = \psi(L)\epsilon_t = \sum_{j=0}^{\infty} \psi_j \epsilon_{t-j}$, where $\sum_{j=0}^{\infty} j|\psi_j| < \infty$ and $\{\epsilon_t\}$ is an i.i.d. sequence with mean zero, variance σ^2 and finite fourth moment.

Define:

$\gamma(j) = \mathrm{E}(u_t u_{t-j}) = \sigma^2 \sum_{s=0}^{\infty} \psi_s \psi_{s+j}$ for $j = 0, 1, 2, \cdots,$

$\lambda = \sigma \sum_{j=0}^{\infty} \psi_j = \sigma\psi(1),$

$\xi_t = \sum_{i=1}^{t} u_i$ for $t = 1, 2, \cdots, T$ and $\xi_0 = 0.$

Then,

1. $T^{-1/2} \sum_{t=1}^{T} u_t \longrightarrow \lambda W(1)$

2. $T^{-1/2} \sum_{t=1}^{T} u_{t-j}\epsilon_t \longrightarrow N(0, \sigma^2\gamma(0)), \quad \text{for } j = 1, 2, \cdots$

3. $T^{-1} \sum_{t=1}^{T} u_t u_{t-j} \longrightarrow \gamma(j), \quad \text{for } j = 1, 2, \cdots$

4. $T^{-1} \sum_{t=1}^{T} \xi_{t-1}\epsilon_t \longrightarrow \frac{1}{2}\sigma\lambda(W(1)^2 - 1)$

5. $T^{-1}\sum_{t=1}^{T}\xi_{t-1}u_{t-j} \longrightarrow \begin{cases} \frac{1}{2}(\lambda^2 W(1)^2 - \gamma(0)), & \text{for } j = 0, \\ \frac{1}{2}(\lambda^2 W(1)^2 - \gamma(0)) + \sum_{i=0}^{j-1}\gamma(i), & \text{for } j = 1, 2, \cdots \end{cases}$

6. $T^{-3/2}\sum_{t=1}^{T}\xi_{t-1} \longrightarrow \lambda\int_0^1 W(r)\mathrm{d}r$

7. $T^{-3/2}\sum_{t=1}^{T}tu_{t-j} \longrightarrow \lambda\left(W(1) - \int_0^1 W(r)\mathrm{d}r\right), \quad \text{for } j = 0, 1, 2, \cdots$

8. $T^{-2}\sum_{t=1}^{T}\xi_{t-1}^2 \longrightarrow \lambda^2\int_0^1 (W(r))^{-2}\mathrm{d}r$

9. $T^{-5/2}\sum_{t=1}^{T} t\xi_{t-1} \longrightarrow \lambda \int_0^1 rW(r)\mathrm{d}r$

10. $T^{-3}\sum_{t=1}^{T} t\xi_{t-1} \longrightarrow \lambda^2 \int_0^1 r(W(r))^2\mathrm{d}r$

11. $T^{-(\mu-1)}\sum_{t=1}^{T} t^{\mu} \longrightarrow \frac{1}{\mu+1}, \quad \text{for } \mu = 0, 1, 2, \cdots$

3.3 Cointegration (共和分)

1. For a scalar y_t, when $\Delta y_t = y_t - y_{t-1}$ is a white noise (i.e., iid), we write $\Delta y_t \sim I(1)$.

2. **Definition of Cointegration:**

   Suppose that each series in a $g \times 1$ vector y_t is $I(1)$, i.e., each series has unit root, and that a linear combination of each series (i.e, $a'y_t$ for a nonzero vector a) is $I(0)$, i.e., stationary.

   Then, we say that y_t has a cointegration.

3. **Example:**

Suppose that $y_t = (y_{1,t},\ y_{2,t})'$ is the following vector autoregressive process:

$$y_{1,t} = \phi_1 y_{2,t} + \epsilon_{1,t},$$

$$y_{2,t} = y_{2,t-1} + \epsilon_{2,t}.$$

Then,

$$\Delta y_{1,t} = \phi_1 \epsilon_{2,t} + \epsilon_{1,t} - \epsilon_{1,t-1}, \quad \text{(MA(1) process)},$$

$$\Delta y_{2,t} = \epsilon_{2,t},$$

where both $y_{1,t}$ and $y_{2,t}$ are $I(1)$ processes.

The linear combination $y_{1,t} - \phi_1 y_{2,t}$ is $I(0)$.

In this case, we say that $y_t = (y_{1,t},\ y_{2,t})'$ is cointegrated with $a = (1,\ -\phi_1)$.

$a = (1,\ -\phi_1)$ is called the cointegrating vector, which is not unique.

Therefore, the first element of a is set to be one.

4. Suppose that $y_t \sim I(1)$ and $x_t \sim I(1)$.

   For the regression model $y_t = x_t\beta + u_t$, OLS does not work well if we do not have the β which satisfies $u_t \sim I(0)$.

   $\Longrightarrow$ **Spurious regression (見せかけの回帰)**

5. Suppose that $y_t \sim I(1)$, y_t is a $g \times 1$ vector and $y_t = \begin{pmatrix} y_{1,t} \\ y_{2,t} \end{pmatrix}$.

$y_{2,t}$ is a $k \times 1$ vector, where $k = g - 1$.

Consider the following regression model:

$$y_{1,t} = \alpha + \gamma' y_{2,t} + u_t, \qquad t = 1, 2, \cdots, T.$$

OLSE is given by:

$$\begin{pmatrix} \hat{\alpha} \\ \hat{\gamma} \end{pmatrix} = \begin{pmatrix} T & \sum y'_{2,t} \\ \sum y_{2,t} & \sum y_{2,t} y'_{2,t} \end{pmatrix}^{-1} \begin{pmatrix} \sum y_{1,t} \\ \sum y_{1,t} y_{2,t} \end{pmatrix}.$$

Next, consider testing the null hypothesis $H_0 : R\gamma = r$, where R is a $m \times k$ matrix ($m \leq k$) and r is a $m \times 1$ vector.

The F statistic, denoted by F_T, is given by:

$$F_T = \frac{1}{m}(R\hat{\gamma} - r)' \left(s_T^2 \begin{pmatrix} 0 & R \end{pmatrix} \begin{pmatrix} T & \sum y_{2,t}' \\ \sum y_{2,t} & \sum y_{2,t} y_{2,t}' \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ R' \end{pmatrix} \right)^{-1} (R\hat{\gamma} - r),$$

where

$$s_T^2 = \frac{1}{T - g} \sum_{t=1}^{T} (y_{1,t} - \hat{\alpha} - \hat{\gamma}' y_{2,t})^2.$$

When we have the γ such that $y_{1,t} - \gamma y_{2,t}$ is stationary, OLSE of γ, i.e., $\hat{\gamma}$, is not statistically equal to zero.

When the sample size T is large enough, H_0 is rejected by the F test.

6. Phillips, P.C.B. (1986) "Understanding Spurious Regressions in Econometrics," *Journal of Econometrics*, Vol.33, pp.95 – 131.

   Consider a $g \times 1$ vector y_t whose first difference is described by:

$$\Delta y_t = \Psi(L)\epsilon_t = \sum_{s=0}^{\infty} \Psi_s \epsilon_{t-s},$$

   for ϵ_t an i.i.d. $g \times 1$ vector with mean zero , variance $\mathrm{E}(\epsilon_t \epsilon_t') = PP'$, and finite fourth moments and where $\{s\Psi_s\}_{s=0}^{\infty}$ is absolutely summable.

Let $k = g - 1$ and $\Lambda = \Psi(1)P$.

Partition y_t as $y_t = \begin{pmatrix} y_{1,t} \\ y_{2,t} \end{pmatrix}$ and $\Lambda\Lambda'$ as $\Lambda\Lambda' = \begin{pmatrix} \Sigma_{11} & \Sigma_{21}' \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}$, where $y_{1,t}$ and Σ_{11} are scalars, $y_{2,t}$ and Σ_{21} are $k \times 1$ vectors, and Σ_{22} is a $k \times k$ matrix.

Suppose that $\Lambda\Lambda'$ is nonsingular,and define $\sigma_1^{*2} = \Sigma_{11} - \Sigma_{21}'\Sigma_{22}^{-1}\Sigma_{21}$.

Let L_{22} denote the Cholesky factor of Σ_{22}^{-1}, i.e., L_{22} is the lower triangular matrix satisfying $\Sigma_{22}^{-1} = L_{22}L_{22}'$.

Then, (a) – (c) hold.

(a) OLSEs of α and γ in the regression model $y_{1,t} = \alpha + \gamma' y_{2,t} + u_t$, denoted by $\hat{\alpha}_T$ and $\hat{\gamma}_T$, are characterized by:

$$\begin{pmatrix} T^{-1/2}\hat{\alpha}_T \\ \hat{\gamma}_T - \Sigma_{22}^{-1}\Sigma_{21} \end{pmatrix} \longrightarrow \begin{pmatrix} \sigma_1^* h_1 \\ \sigma_1^* L_{22} h_2 \end{pmatrix},$$

where

$$\begin{pmatrix} h_1 \\ h_2 \end{pmatrix} = \begin{pmatrix} 1 & \int_0^1 W_2^*(r)'\mathrm{d}r \\ \int_0^1 W_2^*(r)\mathrm{d}r & \int_0^1 W_2^*(r)W_2^*(r)'\mathrm{d}r \end{pmatrix}^{-1} \begin{pmatrix} \int_0^1 W_1^*(r)\mathrm{d}r \\ \int_0^1 W_2^*(r)W_1^*(r)\mathrm{d}r \end{pmatrix},$$

where $W_1^*(r)$ and $W_2^*(r)$ denote scalar and g-dimensional standard Brownian motions, and $W_1^*(r)$ is independent of $W_2^*(r)$.

(b) The sum of squared residuals, denoted by $\mathrm{RSS}_T = \sum_{t=1}^{T} \hat{u}_t^2$, satisfies

$$T^{-2}\mathrm{RSS}_T \longrightarrow \sigma_1^{*2} H,$$

where

$$H = \int_0^1 (W_1^*(r))^2 \mathrm{d}r - \left(\begin{pmatrix} \int_0^1 W_1^*(r)\mathrm{d}r \\ \int_0^1 W_2^*(r) W_1^*(r)\mathrm{d}r \end{pmatrix}' \begin{pmatrix} h_1 \\ h_2 \end{pmatrix} \right)^{-1}.$$

(c) The F_T test satisfies:

$$
\begin{aligned}
T^{-1}F_T \longrightarrow\ & \frac{1}{m}(\sigma_1^* R^* h_2 - r^*)' \\
& \times \left(\sigma_1^{*2} H \begin{pmatrix} 0 & R^* \end{pmatrix} \begin{pmatrix} 1 & \int_0^1 W_2^*(r)' \mathrm{d}r \\ \int_0^1 W_2^*(r) \mathrm{d}r & \int_0^1 W_2^*(r) W_2^*(r)' \mathrm{d}r \end{pmatrix}^{-1} \begin{pmatrix} 0 & R^* \end{pmatrix}' \right)^{-1} \\
& \times (\sigma_1^* R^* h_2 - r^*),
\end{aligned}
$$

where $R^* = RL_{22}$ and $r^* = r - R\Sigma_{22}^{-1}\Sigma_{21}$.

(a) indicates that OLSE $\hat{\gamma}_T$ is not consistent.

(b) indicates that $s_T^2 = \dfrac{1}{T-g}\displaystyle\sum_{t=1}^{T} \hat{u}_t^2$ diverges.

(c) indicates that F_T diverges.

$\Longrightarrow$ **Spurious regression (見せかけの回帰)**

7. **Resolution for Spurious Regression:**

Suppose that $y_{1,t} = \alpha + \gamma' y_{2,t} + u_t$ is a spurious regression.

(1) Estimate $y_{1,t} = \alpha + \gamma' y_{2,t} + \phi y_{1,t-1} + \delta y_{2,t-1} + u_t$.

Then, $\hat{\gamma}_T$ is $\sqrt{T}$-consistent, and the t test statistic goes to the standard normal distribution under $H_0 : \gamma = 0$.

(2) Estimate $\Delta y_{1,t} = \alpha + \gamma' \Delta y_{2,t} + u_t$. Then, $\hat{\alpha}_T$ and $\hat{\beta}_T$ are $\sqrt{T}$-consistent, and the t test and F test make sense.

(3) Estimate $y_{1,t} = \alpha + \gamma' y_{2,t} + u_t$ by the Cochrane-Orcutt method, assuming that u_t is the first-order serially correlated error.

Usually, choose (2).

However, there are two exceptions.

(i) The true value of ϕ is not one, i.e., less than one.

(ii) $y_{1,t}$ and $y_{2,t}$ are the cointegrated processes.

In these two cases, taking the first difference leads to the misspecified regression.

8. **Cointegrating Vector:**

   Suppose that each element of y_t is $I(1)$ and that $a'y_t$ is $I(0)$.

   a is called a **cointegrating vector (共和分ベクトル)**, which is not unique.

   Set $z_t = a'y_t$, where z_t is scalar, and a and y_t are $g \times 1$ vectors.

For $z_t \sim I(0)$ (i.e., stationary),

$$T^{-1} \sum_{t=1}^{T} z_t^2 = T^{-1} \sum_{t=1}^{T} (a' y_t)^2 \longrightarrow \mathrm{E}(z_t^2).$$

For $z_t \sim I(1)$ (i.e., nonstationary, i.e., a is not a cointegrating vector),

$$T^{-2} \sum_{t=1}^{T} (a' y_t)^2 \longrightarrow \lambda^2 \int_0^1 (W(r))^2 \, \mathrm{d}r,$$

where $W(r)$ denotes a standard Brownian motion and λ^2 indicates variance of $(1 - L)z_t$.

If a is not a cointegrating vector, $T^{-1} \sum_{t=1}^{T} z_t^2$ diverges.

$\Longrightarrow$ We can obtain a consistent estimate of a cointegrating vector by minimizing $\sum_{t=1}^{T} z_t^2$ with respect to a, where a normalization condition on a has to be imposed.

The estimator of the a including the normalization condition is super-consistent (T-consistent).

Stock, J.H. (1987) “Asymptotic Properties of Least Squares Estimators of Cointegrating Vectors,” *Econometrica*, Vol.55, pp.1035 – 1056.

Proposition:

Let $y_{1,t}$ be a scalar, $y_{2,t}$ be a $k \times 1$ vector, and $(y_{1,t}, y'_{2,t})'$ be a $g \times 1$ vector, where $g = k + 1$.

Consider the following model:

$$y_{1,t} = \alpha + \gamma' y_{2,t} + z_t^*$$

$$\Delta y_{2,t} = u_{2,t}$$

$$\begin{pmatrix} z_t^* \\ u_{2,t} \end{pmatrix} = \Psi^*(L)\epsilon_t$$

ϵ_t is a $g \times 1$ i.i.d. vector with $\mathrm{E}(\epsilon_t) = 0$ and $\mathrm{E}(\epsilon_t \epsilon_t') = PP'$.

OLSE is given by:

$$\begin{pmatrix} \hat{\alpha} \\ \hat{\gamma} \end{pmatrix} = \begin{pmatrix} T & \sum y_{2,t}' \\ \sum y_{2,t} & \sum y_{2,t} y_{2,t}' \end{pmatrix}^{-1} \begin{pmatrix} \sum y_{1,t} \\ \sum y_{1,t} y_{2,t} \end{pmatrix}.$$

Define λ_1^*, which is a $g \times 1$ vector, and Λ_2^*, which is a $k \times g$ matrix, as follows:

$$\Psi^*(1)\, P = \begin{pmatrix} \lambda_1^{*\prime} \\ \Lambda_2^* \end{pmatrix}.$$

Then, we have the following results:

$$\begin{pmatrix} T^{1/2}(\hat{\alpha}-\alpha) \\ T(\hat{\gamma}-\gamma) \end{pmatrix} \longrightarrow \begin{pmatrix} 1 & \left(\Lambda_2^* \int W(r)\mathrm{d}r\right)' \\ \Lambda_2^* \int W(r)\mathrm{d}r & \Lambda_2^* \left(\int (W(r))\,(W(r))'\,\mathrm{d}r\right) \Lambda_2^{*\prime} \end{pmatrix}^{-1} \begin{pmatrix} h_1 \\ h_2 \end{pmatrix},$$

where

$$\begin{pmatrix} h_1 \\ h_2 \end{pmatrix} = \begin{pmatrix} \lambda_1^{*\prime} W(1) \\ \Lambda_2^* \left(\int W(r)\,(\mathrm{d}W(r))'\right) \lambda_1^* + \sum_{\tau=0}^{\infty} \mathrm{E}(u_{2,t} z_{t+\tau}^{*\prime}) \end{pmatrix}.$$

$W(r)$ denotes a g-dimensional standard Brownian motion.

1) OLSE of the cointegrating vector is consistent even though u_t is serially correlated.

2) The consistency of OLSE implies that $T^{-1} \sum \hat{u}_t^2 \longrightarrow \sigma^2$.

3) Because $T^{-1} \sum (y_{1,t} - \bar{y}_1)^2$ goes to infinity, a coefficient of determination, R^2, goes to one.

3.4 Testing Cointegration

3.4.1 Engle-Granger Test

$y_t \sim I(1)$

$y_{1,t} = \alpha + \gamma' y_{2,t} + u_t$

- $u_t \sim I(0) \implies$ Cointegration
- $u_t \sim I(1) \implies$ Spurious Regression

Estimate $y_{1,t} = \alpha + \gamma' y_{2,t} + u_t$ by OLS, and obtain $\hat{u}_t$.

Estimate $\hat{u}_t = \rho \hat{u}_{t-1} + \delta_1 \Delta \hat{u}_{t-1} + \delta_2 \Delta \hat{u}_{t-2} + \cdots + \delta_{p-1} \Delta \hat{u}_{t-p+1} + e_t$ by OLS.

ADF Test:

- $H_0 : \rho = 1$ (Sprious Regression)
- $H_1 : \rho < 1$ (Cointegration)

$\Longrightarrow$ **Engle-Granger Test**

For example, see Engle and Granger (1987), Phillips and Ouliaris (1990) and Hansen (1992).

Asymmptotic Distribution of Residual-Based ADF Test for Cointegration

# of Refressors,	(a) Regressors have no drift				(b) Some regressors have drift			
excluding constant	1%	2.5%	5%	10%	1%	2.5%	5%	10%
1	−3.96	−3.64	−3.37	−3.07	−3.96	−3.67	−3.41	−3.13
2	−4.31	−4.02	−3.77	−3.45	−4.36	−4.07	−3.80	−3.52
3	−4.73	−4.37	−4.11	−3.83	−4.65	−4.39	−4.16	−3.84
4	−5.07	−4.71	−4.45	−4.16	−5.04	−4.77	−4.49	−4.20
5	−5.28	−4.98	−4.71	−4.43	−5.36	−5.02	−4.74	−4.46

J.D. Hamilton (1994), *Time Series Analysis*, p.766.

3.4.2 Error Correction Representation

VAR(p) model:

$$y_t = \alpha + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \epsilon_t,$$

where y_t, α and ϵ_t indicate $g \times 1$ vectors for $t = 1, 2, \cdots, T$, and ϕ_s is a $g \times g$ matrix for $s = 1, 2, \cdots, p$.

Rewrite:

$$y_t = \alpha + \rho y_{t-1} + \delta_1 \Delta y_{t-1} + \delta_2 \Delta y_{t-2} + \cdots + + \delta_{p-1} \Delta y_{t-p+1} + \epsilon_t,$$

where

$$\rho = \phi_1 + \phi_2 + \cdots + \phi_p,$$

$$\delta_s = -(\phi_{s+1} + \delta_{s+2} + \cdots + \phi_p), \qquad \text{for } s = 1, 2, \cdots, p-1.$$

Again, rewrite:

$$\Delta y_t = \alpha + \delta_0 y_{t-1} + \delta_1 \Delta y_{t-1} + \delta_2 \Delta y_{t-2} + \cdots + + \delta_{p-1} \Delta y_{t-p+1} + \epsilon_t,$$

where

$$\delta_0 = \rho - I_g = -\phi(1),$$

for $\phi(L) = I_g - \delta_1 L - \delta_2 L^2 - \cdots - \delta_p L^p$.

If y_t has h cointegrating relations, we have the following error correction representation:

$$\Delta y_t = \alpha - BA'y_{t-1} + \delta_1 \Delta y_{t-1} + \delta_2 \Delta y_{t-2} + \cdots + +\delta_{p-1} \Delta y_{t-p+1} + \epsilon_t,$$

where $A'y_{t-1}$ is a stationary $h \times 1$ vector (i.e., h I(0) processes), and B and A are $g \times h$ matrices.

Note that $\phi(1) = BA'$ for $\phi(L) = I_g - \delta_1 L - \delta_2 L^2 - \cdots - \delta_p L^p$.

Each row of A' denotes the cointegrating vector, i.e., A' consists of h cointegrating vectors.

Suppose that $\epsilon_t \sim N(0, \Sigma)$. The log-likelihood function is:

$$\begin{aligned}
&\log l(\alpha, \delta_1, \cdots, \delta_{p-1}, B|A) \\
&\quad = -\frac{Tg}{2}\log(2\pi) - \frac{T}{2}\log|\Sigma| \\
&\qquad -\frac{1}{2}\sum_{t=1}^{T}(\Delta y_t - \alpha + BA'y_{t-1} - \delta_1\Delta y_{t-1} - \cdots - \delta_{p-1}\Delta y_{t-p+1})'\Sigma^{-1} \\
&\qquad\qquad \times(\Delta y_t - \alpha + BA'y_{t-1} - \delta_1\Delta y_{t-1} - \cdots - \delta_{p-1}\Delta y_{t-p+1})
\end{aligned}$$

Given A and h, maximize $\log l$ with respect to $\alpha, \delta_1, \cdots, \delta_{p-1}, B$.

Then, given h, how do we estimate A? $\Longrightarrow$ Johansen (1988, 1991)

(*) Canonical Correlatoion (正準相関)

$x' = (x_1, x_2, \cdots, x_n)$ and $y' = (y_1, y_2, \cdots, y_m)$, where $n \leq m$.

$$u = a'x = a_1x_1 + a_2x_2 + \cdots + a_nx_n,$$

$$v = b'y = b_1y_1 + b_2y_2 + \cdots + b_my_m,$$

where $\mathrm{V}(u) = \mathrm{V}(v) = 1$ and $\mathrm{E}(x) = \mathrm{E}(y) = 0$ for simplicity.

Define:

$$\mathrm{V}(x) = \Sigma_{xx}, \qquad \mathrm{E}(xy') = \Sigma_{xy}, \qquad \mathrm{V}(y) = \Sigma_{yy}, \qquad \mathrm{E}(yx') = \Sigma_{yx} = \Sigma'_{xy}.$$

The correlation coefficient between u and v, denoted by ρ, is:

$$\rho = \frac{\mathrm{Cov}(u, v)}{\sqrt{\mathrm{V}(u)}\sqrt{\mathrm{V}(v)}} = a'\Sigma_{xy}b,$$

where $\mathrm{V}(u) = a'\Sigma_{xx}a = 1$ and $\mathrm{V}(v) = b'\Sigma_{yy}b = 1$.

Maximize $\rho = a'\Sigma_{xy}b$ subject to $a'\Sigma_{xx}a = 1$ and $b'\Sigma_{yy}b = 1$.

The Lagrangian is:

$$L = a'\Sigma_{xy}b - \frac{1}{2}\lambda(a'\Sigma_{xx}a - 1) - \frac{1}{2}\mu(b'\Sigma_{yy}b - 1).$$

Take a derivative with respect to a and b.

$$\frac{\partial L}{\partial a} = \Sigma_{xy} b - \lambda \Sigma_{xx} a = 0,$$
$$\frac{\partial L}{\partial b} = \Sigma_{xy}' a - \mu \Sigma_{yy} b = 0.$$

Using $a' \Sigma_{xx} a = 1$ and $b' \Sigma_{yy} b = 1$, we obtain:

$$\lambda = \mu = a' \Sigma_{xy} b.$$

From the first equation, we obtain:

$$a = \frac{1}{\lambda} \Sigma_{xx}^{-1} \Sigma_{xy} b,$$

which is substituted into the second equation as follows:

$$\frac{1}{\lambda}\Sigma'_{xy}\Sigma_{xx}^{-1}\Sigma_{xy}b - \lambda\Sigma_{yy}b = 0,$$

i.e.,

$$(\Sigma_{yy}^{-1}\Sigma'_{xy}\Sigma_{xx}^{-1}\Sigma_{xy} - \lambda^2 I_m)b = 0,$$

i.e.,

$$|\Sigma_{yy}^{-1}\Sigma'_{xy}\Sigma_{xx}^{-1}\Sigma_{xy} - \lambda^2 I_m| = 0.$$

The solution of λ^2 is given by the maximum eigen value of $\Sigma_{yy}^{-1}\Sigma'_{xy}\Sigma_{xx}^{-1}\Sigma_{xy}$, and b is the corresponding eigen vector.

Back to the Cointegration:

Estimate the following two regressions:

$$\Delta y_t = b_{1,0} + b_{1,1}\Delta y_{t-1} + b_{1,2}\Delta y_{t-2} + \cdots + b_{1,p-1}\Delta y_{t-p+1} + u_{1,t}$$

$$y_{t-1} = b_{2,0} + b_{2,1}\Delta y_{t-1} + b_{2,2}\Delta y_{t-2} + \cdots + b_{2,p-1}\Delta y_{t-p+1} + u_{2,t}$$

Obtain $\hat{u}_{i,t}$ for $i = 1, 2$ and $t = 1, 2, \cdots, T$, and compute as follow:

$$\hat{\Sigma}_{11} = \frac{1}{T}\sum_{t=1}^{T} \hat{u}_{1,t}\hat{u}'_{1,t}, \qquad \hat{\Sigma}_{22} = \frac{1}{T}\sum_{t=1}^{T} \hat{u}_{2,t}\hat{u}'_{2,t},$$

$$\hat{\Sigma}_{12} = \frac{1}{T}\sum_{t=1}^{T} \hat{u}_{1,t}\hat{u}'_{2,t}, \qquad \hat{\Sigma}_{21} = \hat{\Sigma}'_{12}.$$

From $\hat{\Sigma}_{22}^{-1}\hat{\Sigma}_{21}\hat{\Sigma}_{11}^{-1}\hat{\Sigma}_{12}$, compute h biggest eigenvalues, denoted by $\hat{\lambda}_1, \hat{\lambda}_2, \cdots, \hat{\lambda}_h$, and the corresponding eigen vectors, denoted by $\hat{a}_1, \hat{a}_2, \cdots, \hat{a}_h$, where $\hat{\lambda}_1 > \hat{\lambda}_2 > \cdots > \hat{\lambda}_h$,

The estimate of A, $\hat{A}$, is given by $\hat{A} = (\hat{a}_1, \hat{a}_2, \cdots, \hat{a}_h)$.

How do we obtain h?

3.5 Testing the Number of Cointegrating Vectors

Trace Test (トレース検定):

$H_0: \lambda_{h+1} = 0$ and $H_1: \lambda_h > 0$.

$$2(\log l_1 - \log l_0) = -T \sum_{i=h+1}^{g} \log(1 - \hat{\lambda}_i) \longrightarrow \mathrm{tr}(Q),$$

where

$$Q = \left(\int_0^1 W(r) \mathrm{d}W(r)' \right)' \left(\int_0^1 W(r) W(r)' \mathrm{d}r \right)^{-1} \left(\int_0^1 W(r) \mathrm{d}W(r)' \right).$$

Trace Test for # of Cointegrating Relations

# of Random	(a) Regressors have no drift				(b) Some regressors have drift			
Walks ($g - h$)	1%	2.5%	5%	10%	1%	2.5%	5%	10%
1	11.576	9.658	8.083	6.691	6.936	5.332	3.962	2.816
2	21.962	19.611	17.844	15.583	19.310	17.299	15.197	13.338
3	37.291	34.062	31.256	28.436	35.397	32.313	29.509	26.791
4	55.551	51.801	48.419	45.248	53.792	50.424	47.181	43.964
5	77.911	73.031	69.977	65.956	76.955	72.140	68.905	65.063

J.D. Hamilton (1994), *Time Series Analysis*, p.767.

Largest Eigenvalue Test (最大固有値検定):

H_0 : $\lambda_{h+1} = 0$ and H_1 : $\lambda_h > 0$.

$$2(\log l_1 - \log l_0) = -T \log(1 - \hat{\lambda}_{h+1}) \longrightarrow \text{maxmum eigen value of } Q,$$

Maximum Eigenvalue Test for # of Cointegrating Relations

# of Random	(a) Regressors have no drift				(b) Some regressors have drift			
Walks ($g - h$)	1%	2.5%	5%	10%	1%	2.5%	5%	10%
1	11.576	9.658	8.083	6.691	6.936	5.332	3.962	2.816
2	18.782	16.403	14.595	12.783	17.936	15.810	14.036	12.099
3	26.154	23.362	21.279	18.959	25.521	23.002	20.778	18.697
4	32.616	29.599	27.341	24.917	31.943	29.335	27.169	24.712
5	38.858	35.700	33.262	30.818	38.341	35.546	33.178	30.774

J.D. Hamilton (1994), *Time Series Analysis*, p.768.

4 GMM (Generalized Mothod of Moments, 一般化積率法)

1. Method of Moments (積率法):

   Regression Model: $y_t = x_t\beta + \epsilon_t$

   From the assumption, $\mathrm{E}(x_t'\epsilon_t) = 0$.

   The sample mean is given by:

$$\frac{1}{T}\sum_{t=1}^{T} x_t'\epsilon_t = \frac{1}{T}\sum_{t=1}^{T} x_t'(y_t - x_t\beta) = 0.$$

Therefore,

$$\beta_{MM} = \left(\frac{1}{T} \sum_{t=1}^{T} x_t' x_t \right)^{-1} \left(\frac{1}{T} \sum_{t=1}^{T} x_t' y_t \right),$$

which is equivalent to OLS.

2. Generalized Mothod of Moments (GMM, 一般化積率法):

$$\mathrm{E}\left(h(\theta; w_t)\right) = 0$$

θ is a $k \times 1$ parameter vector to be estimated.

w_t is an observed vector $w_t = (y_t, x_t)$.

$h(\theta; w_t)$ is a $r \times 1$ vector function, where $r \geq k$.

Define $g(\theta; W_T)$ as follows:

$$g(\theta; W_T) = \frac{1}{T} \sum_{t=1}^{T} h(\theta; w_t),$$

where $W_T = \{w_T, w_{T-1}, \cdots, w_1\}$.

Compute:

$$\min_{\theta} \; g(\theta; W_T)' S^{-1} g(\theta; W_T)$$

The solution of θ, denoted by $\hat{\theta}_T$, corresponds to the GMM estimator, where

S is defined as follows:

$$S = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \sum_{\tau=-\infty}^{\infty} \mathrm{E}\left(h(\theta; w_t)h(\theta; w_{t-\tau})'\right).$$

In empirical studies, S is replaced by its estimate, i.e., $\hat{S}_T$.

When $h(\theta; w_t)$, $t = 1, 2, \cdots, T$, are not serially correlated, the following $\hat{S}_T$ is consistent, i.e.,

$$\hat{S}_T = \frac{1}{T} \sum_{t=1}^{T} h(\hat{\theta}_T; w_t)h(\hat{\theta}_T; w_t)' \longrightarrow S.$$

When $h(\theta; w_t)$, $t = 1, 2, \cdots, T$, are serially correlated,

$$\hat{S}_T = \hat{\Gamma}(0) + \sum_{\tau=1}^{q} k(\frac{\tau}{q+1})(\hat{\Gamma}(\tau) + \hat{\Gamma}(\tau)'),$$

where $\hat{\Gamma}(\tau) = \frac{1}{T} \sum_{t=\tau+1}^{T} h(\hat{\theta}_T; w_t) h(\hat{\theta}_T; w_{t-s})'$.

$k(x) = 1 - x \implies$ Bartlett kernel (Newwey-west estimator),

$k(x) \implies$ Parzen kernel, and etc.

Then, we obtain:

$$\sqrt{T}(\hat{\theta}_T - \theta) \longrightarrow N\left(0, (DS^{-1}D')^{-1}\right),$$

where

$$D = \frac{\partial g(\theta; W_T)}{\partial \theta'}.$$

Note that D is a $r \times k$ matrix.

Let $\hat{D}_T$ be an estimate of D.

The variance estimator of $\hat{\theta}_T$ is given by:

$$\hat{D}_T = \frac{\partial g(\hat{\theta}_T; W_T)}{\partial \theta'}.$$

Asymptotic Normality:

Assumption 1 : $\hat{\theta}_T \longrightarrow \theta$,

Assumption 2 : $\sqrt{T}g(\theta; W_T) \longrightarrow N(0, S)$.

Then, we have the following first-order approximation:

$$\begin{aligned} g(\theta; W_T) &\approx g(\hat{\theta}_T; W_T) + \frac{\partial g(\hat{\theta}_T; W_T)}{\partial \theta'}(\theta - \hat{\theta}_T) \\ &= g(\hat{\theta}_T; W_T) + \hat{D}_T(\theta - \hat{\theta}_T), \end{aligned}$$

where $g(\theta; W_T)$ is linearized around $\theta = \hat{\theta}_T$.

The first-order condition for the minimization problem is:

$$\left(\frac{\partial g(\theta; W_T)}{\partial \theta'}\right)' S^{-1}\left(g(\theta; W_T)\right) = 0.$$

Substituting the approximation into the above equation, we obtain the following:

$$\begin{aligned} D'S^{-1}\left(g(\theta; W_T)\right) &= D'S^{-1}\left(g(\hat{\theta}_T; W_T) + \hat{D}_T(\theta - \hat{\theta}_T)\right) \\ &= D'S^{-1}g(\hat{\theta}_T; W_T) + D'S^{-1}\hat{D}_T(\theta - \hat{\theta}_T). \end{aligned}$$

Therefore,

$$\sqrt{T}(\hat{\theta}_T - \theta) \approx (D'S^{-1}\hat{D}_T)^{-1}D'S^{-1}\sqrt{T}\left(g(\hat{\theta}_T; W_T) - g(\theta; W_T)\right).$$

Thus, GMM estimator, $\hat{\theta}_T$, has the following asymptotic distribution:

$$\sqrt{T}(\hat{\theta}_T - \theta) \longrightarrow N\left(0, (D'S^{-1}D)^{-1}\right),$$

where $\hat{D}_T \longrightarrow D$ is utilized.

From Assumption 2, we have the following asymptotic distribution:

$$\left(\sqrt{T}g(\theta; W_T)\right)' S^{-1} \left(\sqrt{T}g(\theta; W_T)\right) \longrightarrow \chi^2(r).$$

When θ is replaced by GMM estimator $\hat{\theta}_T$, we have the following distribution:

$$\left(\sqrt{T}g(\hat{\theta}_T; W_T)\right)' \hat{S}_T^{-1} \left(\sqrt{T}g(\hat{\theta}_T; W_T)\right) \longrightarrow \chi^2(r-k),$$

which is called a test of the overidentifying restrictions.

$\Longrightarrow$ J test by Hansen (1982)

k linear combinations consisting of a $r \times 1$ vector $g(\hat{\theta}_T; W_T)$ are zeros. Therefore, the degrees of freedom are $r - k$.

Some Examples:

(a) **OLS:**

Regression Model: $y_t = x_t\beta + \epsilon_t, \quad \mathrm{E}(x_t\epsilon_t) = 0$

$h(\theta; w_t)$ is taken as:

$$h(\theta; w_t) = x_t(y_t - x_t\beta).$$

(b) **IV (Instrumental Variable, 操作変数法):**

Regression Model: $y_t = x_t\beta + \epsilon_t, \quad \mathrm{E}(x_t\epsilon_t) \neq 0, \quad \mathrm{E}(z_t\epsilon_t) = 0$

$h(\theta; w_t)$ is taken as:

$$h(\theta; w_t) = z_t(y_t - x_t\beta),$$

where z_t is a vector of instrumental variables.

(c) **NLS (Nonlinear Least Squares, 非線形最小二乗法)**:

Regression Model: $f(y_t, x_t, \beta) = \epsilon_t, \quad \mathrm{E}(x_t\epsilon_t) \neq 0, \quad \mathrm{E}(z_t\epsilon_t) = 0$

$h(\theta; w_t)$ is taken as:

$$h(\theta; w_t) = z_t f(y_t, x_t, \beta)$$

where z_t is a vector of instrumental variables.

5　Bayesian Estimation (ベイズ推定)

Greenberg, E. (2013) *Introduction to Bayesian Econometrics* (2nd ed.)

安藤知寛 (2010)『ベイズ統計モデリング』(朝倉書店)

豊田秀樹編 (2008)『マルコフ連鎖モンテカルロ法』(朝倉書店)

Dey, D.K. and Rao, C.R., (2005) *Handbook of Statistics, Vol.25: Bayesian Thinking: Modeling and Computation*
繁桝・岸野・大森監訳 (2011)『ベイズ統計分析ハンドブック』(朝倉書店)

5.1 Introduction

Two Events: A and B

Conditional Probability:

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

Posterior Distribution (事後分布): $f_{\theta|y}(\theta|y)$:

$$f_{\theta|y}(\theta|y) = \frac{f_{y|\theta}(y|\theta)\, f_\theta(\theta)}{f_y(y)} = \frac{f_{y|\theta}(y|\theta)\, f_\theta(\theta)}{\int f_{y|\theta}(y|\theta)\, f_\theta(\theta) \mathrm{d}\theta} \propto f_{y|\theta}(y|\theta)\, f_\theta(\theta),$$

where $f_\theta(\theta)$ is called the prior distribution (事前分布).

Example 1: Let x be the number of successes in a series of n trials with probability θ of success in each.

That is, x has the binomial probability function, given θ,

$$f_{x|\theta}(x|\theta) = \binom{n}{x}\theta^x(1-\theta)^{n-x}, \qquad x = 0, 1, \cdots, n.$$

θ is assumed to be the beta distribution:

$$f_\theta(\theta) = \frac{1}{B(p,q)}\theta^{p-1}(1-\theta)^{q-1},$$

for $\leq \theta \leq 1$, which corresponds to a prior distribution.

Before applying Bayes' theorem, $f_x(x)$ is given by:

$$\begin{aligned} f_x(x) &= \int f_{x|\theta}(x|\theta) f_\theta(\theta) \mathrm{d}\theta \\ &= \binom{n}{r} \frac{1}{B(p,q)} \int_0^1 \theta^{p+x-1}(1-\theta)^{q+n-x-1} \mathrm{d}\theta \\ &= \binom{n}{r} \frac{B(p+x, q+n-x)}{B(p,q)}. \end{aligned}$$

The posterior distribution of θ is:

$$f_{\theta|x}(\theta|x) = \frac{1}{B(p+x, q+n-x)} \theta^{p+x-1}(1-\theta)^{q+n-x-1},$$

which is also a beta distribution with prameters $p + x$ and $q + n - x$.

The posterior mean and variance are:

$$\mathrm{E}(\theta|x) = \frac{p+x}{p+q+n}, \qquad \mathrm{V}(\theta|x) = \frac{(p+x)(q+n-x)}{(p+q+n)^2(p+q+n+1)}.$$

Example 2: $x|\theta \sim N(\theta, v)$, where v is known.

$\theta \sim N(m, w)$, where m and w are known. $\implies$ prior dist.

Then, the posterior distribution of θ is:

$$\theta|x \ \sim \ N\left(\frac{wx+vm}{w+v}, \frac{vw}{w+v}\right).$$

Example 3: $x_1, x_2, \cdots, x_n$ are mutually independently and identically distributed as $N(\mu, \sigma^2)$, where μ and σ^2 are unknown.

$$f_{x|\theta}(x|\theta) = \prod_{i=1}^{n} (2\pi\sigma^2)^{-1/2} \exp\Big(-\frac{1}{2\sigma^2}(x_i - \mu)^2\Big)$$
$$= (2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{1}{2\sigma^2}(s^2 + n(\overline{x} - \mu)^2)\Big),$$

where $\overline{x} = (1/n)\sum_{i=1}^{n} x_i$ and $s^2 = \sum_{i=1}^{n}(x_i - \overline{x})^2$.

The prior density is:

$$f_\theta(\theta) = k(a, b, w)\sigma^{b+3} \exp\left(-\frac{1}{2\sigma^2}\Big(a + \frac{(\mu - m)^2}{w}\Big)\right),$$

where $k(a,b,w) = \dfrac{a^{b/2}2^{-(b+1)/2}(\pi w)^{-1/2}}{\Gamma(\frac{1}{2}b)}$ is a constant.

The posterior density is:

$$f_{\theta|x}(\theta|x) = k(a_1,b_1,w_1)\sigma^{-(b_1+3)}\exp\left(-\frac{1}{2\sigma^2}\left(a_1+\frac{(\mu-m_1)^2}{w_1}\right)\right),$$

where $w_1 = \dfrac{w}{1+nw}$, $m_1 = \dfrac{m+nw\bar{x}}{1+nw}$, $b_1 = b+n$, $a_1 = a+s^2+\dfrac{n(\bar{x}-m)^2}{1+nw}$.

Inference on μ: The posterior density of μ is:

$$f(\mu|x) = \int_0^\infty f(\theta|x)\mathrm{d}\sigma^2 = k_\mu(t_1,b_1)\left(1+\frac{(\mu-m_1)^2}{b_1t_1}\right)^{-(b_1+1)/2},$$

where $t_1 = \dfrac{w_1 a_1}{b_1}$ and $k_\mu(t_1, b_1) = \dfrac{1}{\sqrt{t_1 k_1} B(\frac{1}{2}, \frac{1}{2} b_1)}$.

Thus, $\dfrac{\mu - m_1}{\sqrt{t_1}}$ has a t distribution with b_1 degrees of freedom.

Inference of σ^2: The posterior density of σ^2 is:

$$f(\sigma^2|x) = \int_{-\infty}^{\infty} f(\theta|x) \mathrm{d}\mu = k_{\sigma^2}(a_1, b_1) \sigma^{-(b_1+2)} \exp\Big(-\frac{a_1}{2\sigma^2}\Big),$$

where $k_{\sigma^2}(a_1, b_1) = \dfrac{(\frac{1}{2} a_1)^{b_1/2}}{\Gamma(\frac{1}{2} b_1)}$.

Thus, $\dfrac{a_1}{\sigma^2}$ is chi-squared with b_1 degrees of freedom.

5.2 Inference

Posterior Distribution (事後分布): $f_{\theta|y}(\theta|y)$

5.2.1 Point Estimate

Posterior Mean (事後平均):

$$\bar{\theta} = \int_{-\infty}^{\infty} \theta f_{\theta|y}(\theta|y)\mathrm{d}\theta.$$

Posterior Mode (事後モード):

$$\hat{\theta} = \mathrm{argmax}_{\theta}\ f_{\theta|x}(\theta|y).$$

Posterior Median (事後メディアン):

$$\tilde{\theta} \text{ such that } \int_{-\infty}^{\tilde{\theta}} f_{\theta|y}(\theta|y)\mathrm{d}\theta = 0.5.$$

5.2.2 Interval Estimate

$$\int_R f_{\theta|y}(\theta|y)\mathrm{d}\theta = 1 - \alpha,$$

where R is called confidence interval.

Bayesian confidence interval (ベイズ信頼区間) or credible interval (信用区間):

$$P(\theta_L < \theta < \theta_U) = 1 - \alpha.$$

θ_L and θ_U lead to lower and upper bounds.

(θ_L, θ_U) is called Bayesian confidence interval or credible interval.

Highest posterior density interval (最高事後密度区間):

$$f_{\theta|y}(\theta_0|y) \geq f_{\theta|y}(\theta_1|y), \quad \text{for } \theta_0 \in R \text{ and } \theta_1 \notin R.$$

5.2.3 Marginal Likelihood (周辺尤度)

Marginal Likelihood $\Longrightarrow$ Fitness of the Model:

$$f_y(y) = \int f_{y|\theta}(y|\theta)\, f_\theta(\theta) \mathrm{d}\theta,$$

which corresponds to the denominator in the posterior distribution.

5.3 Example: Linear Regression

Regression Model:

$$y = X\beta + u, \qquad u \sim N(0, \sigma^2 I_n),$$

where y and u are $n \times 1$ vectors, X is an $n \times k$ matrix and β is a $k \times 1$ vector.

Likelihood Function: $\theta = (\beta, \sigma^2)$

$$f_{y|\theta}(y|\theta) = (2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{1}{2\sigma^2}(y - X\beta)'(y - X\beta)\Big)$$

Prior Distributions:

$$f_\theta(\beta, \sigma^2) = f_{\beta|\sigma^2}(\beta|\sigma^2) f_{\sigma^2}(\sigma^2),$$

where

$$f_{\beta|\sigma^2}(\beta|\sigma^2) = N(\beta_0, \sigma^2 A^{-1}) = (2\pi\sigma^2)^{-k/2}|A|^{1/2} \exp\Big(-\frac{1}{2\sigma^2}(\beta-\beta_0)'A(\beta-\beta_0)\Big),$$

$$f_{\sigma^2}(\sigma^2) = IG\Big(\frac{\nu_0}{2}, \frac{\lambda_0}{2}\Big) = \frac{(\lambda_0/2)^{\nu_0/2}}{\Gamma(\nu_0/2)}(\sigma^2)^{-\nu_0/2-1} \exp\Big(-\frac{\lambda_0}{2\sigma^2}\Big).$$

β_0, A, ν_0 and λ_0 are called the hyper-parameters.

Note that $Y \sim IG(a, b)$ for $X \sim G(a, b)$ and $Y = \frac{1}{X}$.

The posterior distribution of β and σ^2 is:

$$
\begin{aligned}
f_{\theta|y}(\beta,\sigma^2|y) &\propto f_{y|\theta}(y|\beta,\sigma^2)\, f_{\beta|\sigma^2}(\beta|\sigma^2)\, f_{\sigma^2}(\sigma^2) \\
&= (2\pi\sigma^2)^{-n/2} \exp\Big(-\frac{1}{2\sigma^2}(y-X\beta)'(y-X\beta)\Big) \\
&\quad \times (2\pi\sigma^2)^{-k/2}|A|^{1/2} \exp\Big(-\frac{1}{2\sigma^2}(\beta-\beta_0)'A(\beta-\beta_0)\Big) \\
&\quad \times \frac{(\lambda_0/2)^{\nu_0/2}}{\Gamma(\nu_0/2)} (\sigma^2)^{-\nu_0/2-1} \exp\Big(-\frac{\lambda_0}{2\sigma^2}\Big) \\
&\propto (\sigma^2)^{-(n+k+\nu_0)/2-1} \exp\Big(-\frac{(y-X\beta)'(y-X\beta)+(\beta-\beta_0)'A(\beta-\beta_0)+\lambda_0}{2\sigma^2}\Big) \\
&\propto |\sigma^2\hat{A}|^{-1/2} \exp\Big(-\frac{(\beta-\hat{\beta})'\hat{A}^{-1}(\beta-\hat{\beta})}{2\sigma^2}\Big) \times (\sigma^2)^{-\hat{\nu}/2-1} \exp\Big(-\frac{\hat{\lambda}}{2\sigma^2}\Big)
\end{aligned}
$$

$$\propto f_{\beta|\sigma^2,y}(\beta|\sigma^2, y) \times f_{\sigma^2|y}(\sigma^2|y) = N(\hat{\beta}, \sigma^2\hat{A}) \times IG(\frac{\hat{\nu}}{2}, \frac{\hat{\lambda}}{2})$$

where

$$\hat{\beta} = (X'X + A)^{-1}(X'X\hat{\beta}_{OLS} + A\beta_0), \qquad \hat{\beta}_{OLS} = (X'X)^{-1}X'y,$$

$$\hat{A} = (X'X + A)^{-1}, \qquad \hat{\nu} = \nu_0 + n,$$

$$\hat{\lambda} = \lambda_0 + (y - X\hat{\beta})'(y - X\hat{\beta}) + (\beta_0 - \hat{\beta}_{OLS})'((X'X)^{-1} + A^{-1})^{-1}(\beta_0 - \hat{\beta}_{OLS}).$$

The marginal posterior distribution of β is:

$$f_{\beta|y}(\beta|y) = \int f_{\theta|y}(\beta, \sigma^2|y)\mathrm{d}\sigma^2 = \int f_{\beta|\sigma^2,y}(\beta|\sigma^2, y)\, f_{\sigma^2|y}(\sigma^2|y)\mathrm{d}\sigma^2$$
$$\propto \left(1 + \frac{1}{\hat{\nu}}(\beta - \hat{\beta})'\left(\frac{\hat{\lambda}}{\hat{\nu}}\hat{A}\right)^{-1}(\beta - \hat{\beta})\right)^{-(\hat{\nu}+k)/2},$$

which is a k-dimensional t distribution with parameters $\hat{\beta}$, $\frac{\hat{\lambda}}{\hat{\nu}}\hat{A}$ and $\hat{\nu}$.

Note that the k-dimensional t distribution with parameters μ, Σ and ν is given by:

$$f(x) = \frac{\Gamma(\frac{\nu+k}{2})}{\Gamma(\frac{\nu}{2})(\nu\pi)^{k/2}}|\Sigma|^{-1/2}\left(1 + \frac{1}{\nu}(x - \mu)'\Sigma^{-1}(x - \mu)\right)^{-(\nu+k)/2}.$$

The marginal likelihood is:

$$f_y(y) = \frac{f_{y|\theta}(y|\theta)\, f_\theta(\theta)}{f_{\theta|y}(\theta|y)} = \frac{|\hat{A}|^{1/2}|A|^{1/2}(\lambda_0/2)^{\nu_0/2}\Gamma(\hat{\nu}/2)}{\pi^{n/2}\Gamma(\nu_0/2)(\hat{\lambda}/2)^{\hat{\nu}/2}},$$

which is utilized for model selection.

In general, how do we evaluate $f_{\theta|y}(\theta|y)$, $\mathrm{E}(\theta|y)$, $f_y(y)$ and so on?

5.4 On Prior Distribution

5.4.1 Non-informative Prior

$$f_\theta(\theta) = \text{ const.}$$

In this case, the posterior distribution is:

$$f_{\theta|y}(\theta|y) \propto f_{y|\theta}(y|\theta),$$

which is proportional to the likelihood function.

However, we have the case where the integration of prior diverges, i.e.,

$$\int f_\theta(\theta)\mathrm{d}\theta = \infty.$$

In this case, $f_\theta(\theta)$ is called an improper prior.

5.4.2 Jeffreys' Prior

$$f_\theta(\theta) \propto |J(\theta)|^{1/2},$$

where

$$J(\theta) = -\int \frac{\partial^2 \log f_{y|\theta}(y|\theta)}{\partial\theta\partial\theta'} f_{y|\theta}(y|\theta)\mathrm{d}y = -\mathrm{E}\Big(\frac{\partial^2 \log f_{y|\theta}(y|\theta)}{\partial\theta\partial\theta'}\Big),$$

which is Fisher's information matrix.

5.5 Evaluation of Expectation

Posterior distribution $f_{\theta|y}(\theta|y)$

$$\mathrm{E}(\theta|y) = \int \theta f_{\theta|y}(\theta|y)\mathrm{d}\theta = \frac{\int \theta f_{y|\theta}(y|\theta) f_{\theta}(\theta)\mathrm{d}\theta}{\int f_{y|\theta}(y|\theta) f_{\theta}(\theta)\mathrm{d}\theta}.$$

In the case where it is not easy to evaluate $\mathrm{E}(\theta|y)$, how do we do?

Bayesian Method = Evaluation of Integration (Too much to say?)

- Numerical Integration
- Monte Carlo Integration
- Random Number Generation from $f_{\theta|y}(\theta|y)$

5.5.1 Evaluation of Expectation: Numerical Integration

Univariate Case: Consider integration of a function $f(x)$.

Suppose that x is a scalar.

Let $x_0, x_1, x_2, \cdots, x_n$ be n nodes, which are sorted by order of size but not necessarily equal intervals between x_{i-1} and x_i for $i = 1, 2, \cdots, n$.

Rectangular Approximation:

$$\int f(x)\mathrm{d}x \approx \sum_{i=1}^{n} f(x_i)(x_i - x_{i-1}) \quad \text{or} \quad \sum_{i=1}^{n} f(x_{i-1})(x_i - x_{i-1}).$$

Trapezoid Approximation:

$$\int f(x)\mathrm{d}x \approx \sum_{i=1}^{n} \frac{1}{2}(f(x_i) + f(x_{i-1}))(x_i - x_{i-1}).$$

Bivariate Case: Consider integration of a function $f(x, y)$.

Suppose that both x and y are scalars.

Let x_0, x_1, x_2, $\cdots$, x_n be n nodes, which are sorted by order of size not necessarily equal intervals between x_{i-1} and x_i for $i = 1, 2, \cdots, n$.

Let $y_0, y_1, y_2, \cdots, y_m$ be m nodes.

Rectangular Approximation:

$$\int\int f(x,y)\mathrm{d}x\mathrm{d}y \approx \sum_{i=1}^{n}\sum_{j=1}^{m} f(x_i, y_j)(x_i - x_{i-1})(y_j - y_{j-1}).$$

Trapezoid Approximation:

$$\int\int f(x.y)\mathrm{d}x\mathrm{d}y$$
$$\approx \sum_{i=1}^{n}\sum_{j=1}^{m} \frac{1}{4}(f(x_i, y_j) + f(x_i, y_{j-1}) + f(x_{i-1}, y_j) + f(x_{i-1}, y_{j-1}))(x_i - x_{i-1})(y_j - y_{j-1}).$$

Applying to Bayes Method (Rectangular Approximation):

$$\mathrm{E}(\theta|y) = \frac{\int \theta f_{y|\theta}(y|\theta) f_\theta(\theta) \mathrm{d}\theta}{\int f_{y|\theta}(y|\theta) f_\theta(\theta) \mathrm{d}\theta} = \frac{\sum_{i=1}^{n} \theta_i f_{y|\theta}(y|\theta_i) f_\theta(\theta_i)(\theta_i - \theta_{i-1})}{\sum_{i=1}^{n} f_{y|\theta}(y|\theta_i) f_\theta(\theta_i)(\theta_i - \theta_{i-1})}$$

$$= \frac{\sum_{i=1}^{n} \theta_i f_{y|\theta}(y|\theta_i) f_\theta(\theta_i)}{\sum_{i=1}^{n} f_{y|\theta}(y|\theta_i) f_\theta(\theta_i)} = \sum_{i=1}^{n} \theta_i \omega_i, \quad \text{for constant } \theta_i - \theta_{i-1},$$

where

$$\omega_i = \frac{f_{y|\theta}(y|\theta_i) f_\theta(\theta_i)}{\sum_{i=1}^{n} f_{y|\theta}(y|\theta_i) f_\theta(\theta_i)}.$$

Problem of Numerical Integration:

1. Choice of initial and terminal values $\Longrightarrow$ Truncation errors

2. Accumulation of computational errors by computer

3. Increase of computational burden for large dimension.

   $\Longrightarrow$ k dimension, and n nodes for each dimension $\Longrightarrow$ n^k

5.5.2 Evaluation of Expectation: Monte Carlo Integration

Univariate Case: Consider integration of a function $f(x)$.

Suppose that x is a scalar.

Let $x_1, x_2, \cdots, x_n$ be n random draws generated from $g(x)$.

$$\int f(x)\mathrm{d}x = \int \frac{f(x)}{g(x)} g(x)\mathrm{d}x = \mathrm{E}\Big(\frac{f(x)}{g(x)}\Big) \approx \frac{1}{n}\sum_{i=1}^{n} \frac{f(x_i)}{g(x_i)}.$$

$\Longrightarrow$ **Importance Sampling (重点的サンプリング)**

Multivariate Case: Consider integration of a function $f(x)$.

Suppose that x is a vector.

Let $x_1, x_2, \cdots, x_n$ be n random draws generated from $g(x)$.

$$\int f(x)\mathrm{d}x = \int \frac{f(x)}{g(x)} g(x)\mathrm{d}x = \mathrm{E}\Big(\frac{f(x)}{g(x)}\Big) \approx \frac{1}{n}\sum_{i=1}^{n} \frac{f(x_i)}{g(x_i)},$$

which is exacly the same as the univariate case.

Computational burden: $\Longrightarrow$ Univariate case: n, Multivariate case: n

Precision of integration ???

Especially, when $g(x)$ is not close to $f(x)$, approximation is prror.

Applying to Bayes Method:

$$\mathrm{E}(\theta|y) = \frac{\int \theta f_{y|\theta}(y|\theta) f_\theta(\theta) \mathrm{d}\theta}{\int f_{y|\theta}(y|\theta) f_\theta(\theta) \mathrm{d}\theta} = \frac{\int \theta \dfrac{f_{y|\theta}(y|\theta) f_\theta(\theta)}{g(\theta)} g(\theta) \mathrm{d}\theta}{\int \dfrac{f_{y|\theta}(y|\theta) f_\theta(\theta)}{g(\theta)} g(\theta) \mathrm{d}\theta} = \frac{(1/n) \sum_{i=1}^n \theta_i \omega(\theta_i)}{(1/n) \sum_{i=1}^n \omega(\theta_i)},$$

where

$$\omega(\theta_i) = \frac{f_{y|\theta}(y|\theta_i) f_\theta(\theta_i)}{g(\theta_i)}.$$

Choice of $g(\theta)$ — One Solution: Define $l(\theta) \equiv f_{y|\theta}(y|\theta)\, f_\theta(\theta)$.

$$\begin{aligned}\log l(\theta) &\approx \log l(\tilde{\theta}) + \frac{1}{l(\tilde{\theta})}\frac{\partial l(\tilde{\theta})}{\partial \theta}(\theta - \tilde{\theta}) \\ &\quad + \frac{1}{2}(\theta - \tilde{\theta})'\Big(-\frac{1}{l(\tilde{\theta})^2}\frac{\partial l(\tilde{\theta})}{\partial \theta}\frac{\partial l(\tilde{\theta})}{\partial \theta'} + \frac{1}{l(\tilde{\theta})}\frac{\partial^2 l(\tilde{\theta})}{\partial \theta \partial \theta'}\Big)(\theta - \tilde{\theta}) \\ &= -\frac{1}{2}(\theta - \tilde{\theta})'\Big(-\frac{1}{l(\tilde{\theta})}\frac{\partial^2 l(\tilde{\theta})}{\partial \theta \partial \theta'}\Big)(\theta - \tilde{\theta}), \qquad \text{when } \tilde{\theta} \text{ is a mode of } l(\theta).\end{aligned}$$

Thus, $N\Big(\tilde{\theta},\ \Big(-\frac{1}{l(\tilde{\theta})}\frac{\partial^2 l(\tilde{\theta})}{\partial \theta \partial \theta'}\Big)^{-1}\Big)$ might be taken as the importance density $g(\theta)$.

5.5.3 Evaluation of Expectation: Random Number Generation

Generate random draws of θ from the posterior distribution $f_{\theta|y}(\theta|y)$.

Then, $(1/n)\sum_{i=1}^{n}\theta_i$ is taken as a consistent estimator of $\mathrm{E}(\theta|y)$, where θ_i indicates the ith random draw generated from $f_{\theta|y}(\theta|y)$.

Note that $(1/n)\sum_{i=1}^{n}\theta_i \longrightarrow \mathrm{E}(\theta|y)$ under the condition $(1/n)\sum_{i=1}^{n}\theta_i < \infty$.

Bayesian confidence interval, median, quntiles and so on are obtained by sorting θ_1, $\theta_2, \cdots, \theta_n$ in order of size.

$\Longrightarrow$ Sampling methods

5.6 Sampling Method I: Random Number Generation

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 section are based on uniform random draws between zero and one.

5.6.1 Uniform Distribution: $U(0, 1)$

Properties of Uniform Distribution: The most heuristic and simplest distribution is uniform.

The **uniform distribution** between zero and one is given by:

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

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 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.

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 (余り)** when k is divided by n.

n is called the **modulus (商)**.

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 (線形合同法)**.

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).

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:

$$\begin{aligned} ax \bmod n &= (ax - [x/q]n) \bmod n \\ &= \Big(ax - [x/q](aq + r)\Big) \bmod n \end{aligned}$$

$$= \left(a(x - [x/q]q) - [x/q]r\right) \bmod n$$
$$= \left(a(x \bmod q) - [x/q]r\right) \bmod n.$$

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)

```
      subroutine urnd(ix,iy,rn)
c
c  Input:
c    ix, iy:  Seeds
c  Output:
c    rn: Uniform Random Draw U(0,1)
c
     1 kx=ix/53668
       ix=40014*(ix-kx*53668)-kx*12211
       if(ix.lt.0) ix=ix+2147483563
c
       ky=iy/52774
       iy=40692*(iy-ky*52774)-ky*3791
       if(iy.lt.0) iy=iy+2147483399
c
       rn=ix-iy
       if( rn.lt.1.) rn=rn+2147483562
       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.

5.6.2 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.

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 $-\infty < x < \infty$.

Mean, variance and the moment-generating function are given by:

$$\mathrm{E}(X) = 0, \qquad \mathrm{V}(X) = 1, \qquad \phi(\theta) = \exp\left(\frac{1}{2}\theta^2\right).$$

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:

$$X_1 = \sqrt{-2\log(U_1)}\cos(2\pi U_2),$$
$$X_2 = \sqrt{-2\log(U_1)}\sin(2\pi U_2).$$

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:

$$u_1 = \exp\left(-\frac{x_1^2 + x_2^2}{2}\right), \qquad u_2 = \frac{1}{2\pi}\arctan\frac{x_2}{x_1}.$$

We perform transformation of variables in multivariate cases.

From this transformation, the Jacobian is obtained as:

$$J = \begin{vmatrix} \frac{\partial u_1}{\partial x_1} & \frac{\partial u_1}{\partial x_2} \\ \frac{\partial u_2}{\partial x_1} & \frac{\partial u_2}{\partial x_2} \end{vmatrix} = \begin{vmatrix} -x_1 \exp\left(-\frac{1}{2}(x_1^2 + x_2^2)\right) & -x_2 \exp\left(-\frac{1}{2}(x_1^2 + x_2^2)\right) \\ \frac{1}{2\pi}\frac{-x_2}{x_1^2 + x_2^2} & \frac{1}{2\pi}\frac{x_1}{x_1^2 + x_2^2} \end{vmatrix}$$

$$= -\frac{1}{2\pi} \exp\left(-\frac{1}{2}(x_1^2 + x_2^2)\right).$$

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.

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
```

```
c  Input:
c    ix, iy:  Seeds
c  Output:
c    rn: Standard Normal Random Draw N(0,1)
c
       pi= 3.1415926535897932385
       call urnd(ix,iy,rn1)
       call urnd(ix,iy,rn2)
       rn=sqrt(-2.0*log(rn1))*sin(2.0*pi*rn2)
       return
       end
```

`snrnd(ix,iy,rn)` should be used together with the uniform random number generator `urnd(ix,iy,rn)` shown in Section 5.6.1 (p.267).

`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).

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)):

$$P(X > x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}(b_1 t + b_2 t^2 + b_3 t^3 + b_4 t^4 + b_5 t^5), \qquad t = \frac{1}{1 + a_0 x},$$

$$a_0 = 0.2316419, \qquad b_1 = 0.319381530, \qquad b_2 = -0.356563782,$$

$$b_3 = 1.781477937, \quad b_4 = -1.821255978, \quad b_5 = 1.330274429.$$

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)

```
1:        subroutine snperpt(p,x)
2: c
3: c  Input:
4: c    p:  Probability
5: c        (err<p<1-err, where err=1e-20)
6: c  Output:
7: c    x:  N(0,1) Percent Point corresponding to p
8: c
9:        p0=-0.322232431088
10:       p1=-1.0
11:       p2=-0.342242088547
12:       p3=-0.204231210245e-1
13:       p4=-0.453642210148e-4
14:       q0= 0.993484626060e-1
15:       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.

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 $-\infty < x < \infty$.

μ is called a **location parameter** and σ^2 is a **scale parameter**.

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 5.6.2.

$X = \sigma Z + \mu$ 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) and snrnd(ix,iy,rn). It is possible to replace snrnd(ix,iy,rn) by snrnd2(ix,iy,rn) or snrnd3(ix,iy,rn).

Exponential Distribution: The exponential distribution with parameter β is written as:

$$f(x) = \begin{cases} \frac{1}{\beta} e^{-\frac{x}{\beta}}, & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

for $\beta > 0$.

β indicates a scale parameter.

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:

$$X = -\beta \log(U).$$

Then, X is an exponential distribution with parameter β.

Because the transformation is given by $u = \exp(-x/\beta)$, the Jacobian is:

$$J = \frac{\mathrm{d}u}{\mathrm{d}x} = -\frac{1}{\beta} \exp\left(-\frac{1}{\beta}x\right).$$

By transforming the variables, the density function of X is represented as:

$$f(x) = |J| f_u\left(\exp(-\frac{1}{\beta}x)\right) = \frac{1}{\beta}\exp\left(-\frac{1}{\beta}x\right),$$

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)`. When $\beta = 2$, the exponential distribution reduces to the chi-square distribution with 2 degrees of freedom.

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:

$$\Gamma(\alpha + 1) = \alpha\Gamma(\alpha), \qquad \Gamma(1) = 1, \qquad \Gamma\left(\frac{1}{2}\right) = 2\Gamma\left(\frac{3}{2}\right) = \sqrt{\pi}.$$

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 5.6.2.

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 5.6.2.

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:
```

```
c     ix, iy:   Seeds
c     alpha:    Shape Parameter (which should be an integer)
c     beta:     Scale Parameter
c   Output:
c     rn: Gamma Random Draw with alpha and beta
c
       rn=0.0
        do 1 i=1,nint(alpha)
       call exprnd(ix,iy,beta,rn1)
     1 rn=rn+rn1
       return
       end
```

gammarnd(ix,iy,alpha,beta,rn) is utilized together with urnd(ix,iy,rn) and exprnd(ix,iy,rn).

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.

When $\alpha = k/2$ and $\beta = 2$, the gamma distribution reduces to the chi-square distribution with k degrees of freedom.

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 equivalent to the gamma distribution with $\beta = 2$ and $\alpha = k/2$.

The chi-square distribution with $k = 2$ reduces to the exponential distribution with $\beta = 2$, shown in Section 5.6.2.

Mean, variance and the moment-generating function are given by:

$$\mathrm{E}(X) = k, \qquad \mathrm{V}(X) = 2k, \qquad \phi(\theta) = \frac{1}{(1-2\theta)^{k/2}}.$$

F **Distribution:** $F(m,n)$**:** The F distribution with m and n degrees of freedom, denoted by $F(m,n)$, is represented as:

$$f(x) = \begin{cases} \dfrac{\Gamma(\frac{m+n}{2})}{\Gamma(\frac{m}{2})\Gamma(\frac{n}{2})}\left(\dfrac{m}{n}\right)^{\frac{m}{2}} x^{\frac{m}{2}-1}\left(1+\dfrac{m}{n}x\right)^{-\frac{m+n}{2}}, & \text{for } 0 < x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

where m and n are positive integers.

Mean and variance are given by:

$$\mathrm{E}(X) = \frac{n}{n-2}, \qquad \text{for } n > 2,$$
$$\mathrm{V}(X) = \frac{2n^2(m+n-2)}{m(n-2)^2(n-4)}, \qquad \text{for } n > 4.$$

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.

***t* Distribution: $t(k)$:** The *t* distribution (or Student's *t* distribution) with k degrees of freedom, denoted by $t(k)$, is given by:

$$f(x) = \frac{\Gamma(\frac{k+1}{2})}{\Gamma(\frac{k}{2})} \frac{1}{\sqrt{k\pi}} \left(1 + \frac{x^2}{k}\right)^{-\frac{k+1}{2}},$$

for $-\infty < x < \infty$, where k does not have to be an integer but conventionally it is a positive integer.

When k is small, the *t* distribution has fat tails.

The *t* distribution with $k = 1$ is equivalent to 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, i.e.,

$$\left(1 + \frac{x^2}{k}\right)^{-\frac{k+1}{2}} = \left(1 + \frac{1}{h}\right)^{-\frac{hx^2+1}{2}} = \left((1 + \frac{1}{h})^h\right)^{-\frac{1}{2}x^2}\left(1 + \frac{1}{h}\right)^{-\frac{1}{2}} \longrightarrow e^{-\frac{1}{2}x^2},$$

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.

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.

Marsaglia (1984) gives a very fast algorithm for generating t random draws, which is based on a transformed acceptance/rejection method, which will be discussed later.

5.6.3 Inverse Transform Method

In Section 5.6.2, 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 $F(\cdot)$.

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:

$$0 = k(x) \approx k(x^*) + k'(x^*)(x - x^*).$$

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.

5.6.4 Using $U(0, 1)$: Discrete Type

In Sections 5.6.2 and 5.6.3, the random number generators from continuous distributions are discussed, i.e., the transformation of variables in Section 5.6.2 and the inverse transform method in Section 5.6.3 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 proba-

bility 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.

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 Dis-*

tributions, 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.

5.7 Sampling Method II: Random Number Generation

5.7.1 Rejection Sampling (棄却法)

We want to generate random draws from $f(x)$, called the **target density (**目的密度**)**, 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 (**サンプリング密度**)** or **proposal density (**提案密度**)**.

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.,

$$q(x) = \frac{f(x)}{f_*(x)}. \tag{1}$$

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 (棄却法)** in this section, **impor-**

tance resampling (重点的リサンプリング法) in Section 5.7.2 and the **Metropolis-Hastings algorithm (メトロポリスーハスティング・アルゴリズム)** in Section 5.7.4.

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 (採択確率)**.

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:

$$P(X \leq x, U \leq \omega(x^*)) = \int_{-\infty}^{x} \int_{0}^{\omega(t)} f_{u,*}(u,t)\, \mathrm{d}u\, \mathrm{d}t = \int_{-\infty}^{x} \int_{0}^{\omega(t)} f_u(u) f_*(t)\, \mathrm{d}u\, \mathrm{d}t$$

$$= \int_{-\infty}^{x} \Big(\int_{0}^{\omega(t)} f_u(u)\, \mathrm{d}u \Big) f_*(t)\, \mathrm{d}t = \int_{-\infty}^{x} \Big(\int_{0}^{\omega(t)} \mathrm{d}u \Big) f_*(t)\, \mathrm{d}t$$
$$= \int_{-\infty}^{x} \Big[u \Big]_{0}^{\omega(t)} f_*(t)\, \mathrm{d}t = \int_{-\infty}^{x} \omega(t) f_*(t)\, \mathrm{d}t = \int_{-\infty}^{x} \frac{q(t)}{c} f_*(t)\, \mathrm{d}t = \frac{F(x)}{c},$$

and the denominator is given by:

$$P(U \leq \omega(x^*)) = P(X \leq \infty, U \leq \omega(x^*)) = \frac{F(\infty)}{c} = \frac{1}{c}.$$

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:

$$P(X \leq x | U \leq \omega(x^*)) = F(x).$$

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 $P(U \leq \omega(x^*)) = 1/c$.

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:

$$c = \sup_x q(x) = \frac{\sigma_*}{\sigma} \exp\Big(\frac{1}{2}\frac{(\mu - \mu_*)^2}{\sigma_*^2 - \sigma^2}\Big).$$

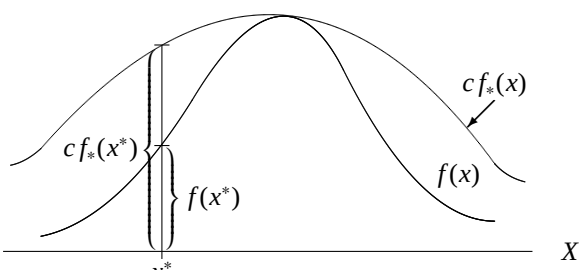
When $\sigma^2 = \sigma_*^2$ and $\mu = \mu_*$, we have $q(x) = 1$, which implies $\omega(x) = 1$.

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 1: Rejection Sampling



From the definition of $\omega(x)$, we have the inequality $f(x) \leq c f_*(x)$.

$c f_*(x)$ and $f(x)$ are displayed in Figure 1.

The ratio of $f(x^*)$ and $c f_*(x^*)$ corresponds to the acceptance probability at x^*, i.e., $\omega(x^*)$.

Thus, for rejection sampling, $c f_*(x)$ 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 $\omega(x) = q(x)/\sup_z q(z)$).

Normal Distribution: $N(0, 1)$**:** First, denote the half-normal distribution by:

$$f(x) = \begin{cases} \dfrac{2}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}, & \text{for } 0 \leq x < \infty, \\ 0, & \text{otherwise.} \end{cases}$$

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:

$$f_*(x) = \begin{cases} \lambda e^{-\lambda x}, & \text{for } 0 \leq x < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

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 $P(U \leq \omega(x^*)) = 1/c$ 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 $u_1 \leq \exp(-\frac{1}{2}(x^* - 1)^2)$, i.e., $-2\log(u_1) \geq (x^* - 1)^2$, 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 $u_3 \leq 1/2$ 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
```

```
15:         call urnd(ix,iy,rn3)
16:           if(rn3.le.0.5) then
17:         rn= y
18:           else
19:         rn=-y
20:           endif
21:         return
22:         end
```

Note that `snrnd6(ix,iy,rn)` should be used together with `urnd(ix,iy,rn)`. Thus, utilizing rejection sampling, we have the standard normal random number generator, which is based on the half-normal distribution.

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.

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.

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 that 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 5.6.3.

gammarnd2(ix,iy,alpha,rn)

```
      subroutine gammarnd2(ix,iy,alpha,rn)
c
c  Use "gammarnd2(ix,iy,alpha,rn)"
c  together with "urnd(ix,iy,rn)".
c
c  Input:
```

```
c    ix, iy: Seeds
c    alpha:  Shape Parameter (0<alpha \le 1)
c  Output:
c    rn: Gamma Random Draw
c        with Parameters alpha and beta=1
c
      e=2.71828182845905
     1 call urnd(ix,iy,rn0)
       call urnd(ix,iy,rn1)
            if( rn0.le.e/(alpha+e) ) then
       rn=( (alpha+e)*rn0/e )**(1./alpha)
       if( rn1.gt.e**(-rn) ) go to 1
            else
       rn=-log((alpha+e)*(1.-rn0)/(alpha*e))
       if( rn1.gt.rn**(alpha-1.) ) go to 1
            endif
       return
       end
```

Note that `gammarnd2(ix,iy,alpha,rn)` should be used with `urnd(ix,iy,rn)`.

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 $\alpha = 10$, 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
```

```
c  Input:
c    ix, iy: Seeds
c    alpha:  Shape Parameter (1<alpha)
c  Output:
c    rn: Gamma Random Draw
c        with Parameters alpha and beta=1
c
       e=2.71828182845905
       a=1./sqrt(2.*alpha-1.)
       b=alpha-log(4.)
       c=alpha+sqrt(2.*alpha-1.)
       d=1.+log(4.5)
     1 call urnd(ix,iy,u1)
       call urnd(ix,iy,u2)
       y=a*log(u1/(1.-u1))
       rn=alpha*(e**y)
       z=u1*u1*u2
       r=b+c*y-rn
       if( r.ge.4.5*z-d ) go to 2
       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 $\beta = 1$.

5.7.2 Importance Resampling (重点的リサンプリング)

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 (1).

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 $\Omega_{j-1} \leq u < \Omega_j$, 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:

$$P(X \leq x) = \int_{-\infty}^{x} f(t)\,\mathrm{d}t = \int_{-\infty}^{x} \frac{f(t)}{f_*(t)} f_*(t)\,\mathrm{d}t = \frac{\int_{-\infty}^{x} q(t) f_*(t)\,\mathrm{d}t}{\int_{-\infty}^{\infty} q(t) f_*(t)\,\mathrm{d}t}$$

$$\approx \frac{(1/n)\sum_{i=1}^{n} q(x_i^*)I(x, x_i^*)}{(1/n)\sum_{j=1}^{n} q(x_j^*)} = \sum_{i=1}^{n} \omega(x_i^*)I(x, x_i^*),$$

where $I(x, x_i^*)$ denotes the indicator function which satisfies $I(x, x_i^*) = 1$ when $x \geq x_i^*$ 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 5.7.1, 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.

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 5.7.1.

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 5.7.1.

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 Steps (iv) and (v), a random draw from $f(x)$ is generated based on Ω_j for $j =$

$1, 2, \cdots, n$.

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 5.7.1, where both $I_1(x)$ and $I_2(x)$ denote the indicator functions defined in Section 5.7.1.

The probability weights are given by:

$$\omega(x_i^*) = \frac{q(x_i^*)}{\sum_{j=1}^n q(x_j^*)} = \frac{f(x_i^*)/f_*(x_i^*)}{\sum_{j=1}^n f(x_j^*)/f_*(x_j^*)}$$

$$= \frac{x_i^{*\alpha-1}e^{-x_i^*}/(x_i^{*\alpha-1}I_1(x_i^*) + e^{-x_i^*}I_2(x_i^*))}{\sum_{j=1}^n x_j^{*\alpha-1}e^{-x_j^*}/(x_j^{*\alpha-1}I_1(x_j^*) + e^{-x_j^*}I_2(x_j^*))},$$

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

The cumulative distribution function of $f_*(x)$ is represented as:

$$F_*(x) = \begin{cases} \dfrac{e}{\alpha + e} x^{\alpha}, & \text{if } 0 < x \leq 1, \\ \dfrac{e}{\alpha + e} + \dfrac{\alpha}{\alpha + e}(1 - e^{-x+1}), & \text{if } x > 1. \end{cases}$$

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 $u_i \leq e/(\alpha + e)$ 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 $\Omega_{i-1} \leq v < \Omega_i$.

As mentioned above, this algorithm yields one random draw.

If we want N random draws, Step (iii) should be repeated N times.

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 $\omega(x_i^*)$.

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.

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

5.7.3 Metropolis-Hastings Algorithm (メトロポリスーハスティングス・アルゴリズム)

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 (1).

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 5.7.1.

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 $P(X \leq x | U \leq \omega(x_{i-1}, x^*))$, which should be the cumulative distribution of X, i.e., $F(x)$.

The probability $P(X \leq x | U \leq \omega(x_{i-1}, x^*))$ is rewritten as follows:

$$P(X \leq x | U \leq \omega(x_{i-1}, x^*)) = \frac{P(X \leq x, U \leq \omega(x_{i-1}, x^*))}{P(U \leq \omega(x_{i-1}, x^*))},$$

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:

$$
P(U \le \omega(x_{i-1}, x^*)) = P(X \le \infty, U \le \omega(x_{i-1}, x^*)) = \frac{f_*(x_{i-1})}{f(x_{i-1})} F(\infty) = \frac{f_*(x_{i-1})}{f(x_{i-1})}.
$$

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:

$$f^*(u|v) = \omega(v,u)\, f_*(u|v) + \Big(1 - \int \omega(v,u)\, f_*(u|v)\, \mathrm{d}u\Big) p(u), \tag{2}$$

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:

$$f(u) = \int f^*(u|v) f(v)\,\mathrm{d}v. \tag{3}$$

Equation (3) holds if we have the following equation:

$$f^*(v|u) f(u) = f^*(u|v) f(v), \tag{4}$$

which is called the **reversibility condition**.

By taking the integration with respect to v on both sides of equation (4), equation (3) is obtained.

Therefore, we have to check whether the $f^*(u|v)$ shown in equation (2) satisfies equation (4).

It is straightforward to verify that

$$\omega(v,u)\,f_*(u|v)\,f(v) = \omega(u,v)\,f_*(v|u)\,f(u),$$
$$\Big(1 - \int \omega(v,u)\,f_*(u|v)\,\mathrm{d}u\Big)p(u)\,f(v) = \Big(1 - \int \omega(u,v)\,f_*(v|u)\,\mathrm{d}v\Big)p(v)\,f(u).$$

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 $\mathrm{Cov}(x_{i-1}, x_i) > 0$, 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 5.7.1, importance resampling in Section 5.7.2 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:

$$\frac{1}{N}\sum_{i=1}^{N} g(x_i) \longrightarrow \mathrm{E}(g(x)) = \int g(x) f(x)\,\mathrm{d}x, \qquad \text{as } N \longrightarrow \infty,$$

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 $\mathrm{E}(g(x))$, 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:

$$p(x) \approx p(x_{i-1}) + p'(x_{i-1})(x - x_{i-1}) + \frac{1}{2}p''(x_{i-1})(x - x_{i-1})^2, \tag{5}$$

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).

Case 1: $\boldsymbol{p''(x_{i-1}) < -\epsilon}$: Equation (5) 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: $\boldsymbol{p''(x_{i-1}) \geq -\epsilon}$ **and** $\boldsymbol{p'(x_{i-1}) < 0}$: 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: $\boldsymbol{p''(x_{i-1}) \geq -\epsilon}$ **and** $\boldsymbol{p'(x_{i-1}) > 0}$: 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: $\boldsymbol{p''(x_{i-1}) \geq -\epsilon}$ **and** $\boldsymbol{p'(x_{i-1}) = 0}$: 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.

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.

Normal Distribution: $N(0,1)$**:** As in Sections 5.7.1 and 5.7.2, 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 5.7.1 and 5.7.2, 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.

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 5.7.1 and `gammarnd4` of 5.7.2, 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 5.7.1.

Then, the acceptance probability is given by:

$$\omega(x_{i-1}, x^*) = \min\Big(\frac{q(x^*)}{q(x_{i-1})}, 1\Big) = \min\Big(\frac{f(x^*)/f_*(x^*)}{f(x_{i-1})/f_*(x_{i-1})}, 1\Big)$$
$$= \min\Big(\frac{x^{*\alpha-1}e^{-x^*}/(x^{*\alpha-1}I_1(x^*) + e^{-x^*}I_2(x^*))}{x_{i-1}^{\alpha-1}e^{-x_{i-1}}/(x_{i-1}^{\alpha-1}I_1(x_{i-1}) + e^{-x_{i-1}}I_2(x_{i-1}))}, 1\Big).$$

As shown in Section 5.7.1, 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, 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 $i = -M+1, -M+2, \cdots, N$.

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 $u \le \omega(x_{i-1}, x^*)$ 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 $i = -M+1, -M+2, \cdots, N$.

5.7.4 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 5.7.1), importance resampling (Section 5.7.2) and the Metropolis-Hastings algorithm (Section 5.7.3).

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:

$$f_{12}(u_1, u_2) = \begin{cases} k, & \text{if } 0 \le u_1 \le \sqrt{h(u_2/u_1)}, \\ 0, & \text{otherwise,} \end{cases}$$

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:

$$f_x(x) = \int_0^{\sqrt{h(x)}} f_{xy}(x, y)\, \mathrm{d}y = \int_0^{\sqrt{h(x)}} ky\, \mathrm{d}y = k\left[\frac{y^2}{2}\right]_0^{\sqrt{h(x)}} = \frac{k}{2} h(x) = f(x),$$

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 $x^2h(x)$ 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 5.7.4 and 5.7.4, we show two examples as applications of the ratio-of-uniforms method.

Especially, in Section 5.7.4, the parallelogram domain of (U_1, U_2) is taken as an example.

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:

$$b = \sup_x \sqrt{h(x)} = 1,$$
$$c = \inf_x x\sqrt{h(x)} = -\sqrt{2e^{-1}},$$
$$d = \sup_x x\sqrt{h(x)} = \sqrt{2e^{-1}}.$$

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., $1/\sqrt{2e/\pi} \approx 0.7602$.

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
```

```
      1 call urnd(ix,iy,rn1)
        call urnd(ix,iy,rn2)
        rn2=(2.*rn2-1.)*sqrt(2.*e1)
        if(-4.*rn1*rn1*log(rn1).lt.rn2*rn2 ) go to 1
        rn=rn2/rn1
        return
        end
```

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 5.7.1 – 5.7.3, 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:

$$h(x) = x^{\alpha-1}e^{-x},$$

$$b = \sup_x \sqrt{h(x)} = \left(\frac{\alpha - 1}{e}\right)^{(\alpha-1)/2},$$

$$c = \inf_x x\sqrt{h(x)} = 0,$$

$$d = \sup_x x\sqrt{h(x)} = \left(\frac{\alpha + 1}{e}\right)^{(\alpha+1)/2}.$$

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)
      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)`.

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).

To see efficiency or inefficiency of the generator above, we compute the acceptance probability in Step (ii) as follows:

$$\frac{\int h(x)\,\mathrm{d}x}{2b(d-c)} = \frac{e^{\alpha}\Gamma(\alpha)}{2(\alpha-1)^{(\alpha-1)/2}(\alpha+1)^{(\alpha+1)/2}}. \tag{6}$$

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)

```
      subroutine gammarnd7(ix,iy,alpha,rn)
c
c  Use "gammarnd7(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
      c0=1.857764
      c1=alpha-1.
      c2=( alpha-1./(6.*alpha) )/c1
      c3=2./c1
      c4=c3+2.
      c5=1./sqrt(alpha)
    1 call urnd(ix,iy,u1)
      call urnd(ix,iy,u2)
      if(alpha.gt.2.5) u1=u2+c5*(1.-c0*u1)
      if(0.ge.u1.or.u1.ge.1.) go to 1
      w=c2*u2/u1
      if(c3*u1+w+1./w.le.c4) go to 2
      if(c3*log(u1)-log(w)+w.ge.1.) go to 1
    2 rn=c1*w
      return
      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 better.

Gamma Distribution: $G(\alpha, \beta)$ **for** $\alpha > 0$ **and** $\beta > 0$**:** Combining `gammarnd2` on p.353 and `gammarnd7` on p.441, 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
```

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.

5.7.5 Gibbs Sampling

The sampling methods introduced in Sections 5.7.1 – 5.7.3 can be applied to the cases of both univariate and multivariate distributions.

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 (4), 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.

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 $f(x,y,z)$.

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 $f_*(x|z) = f_*(x)$.

Comparison of Three Sampling Methods

	μ_* \ σ_*		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 (59.25)	0.000 (100.00)	0.000 (74.89)	0.000 (59.04)	0.000 (40.99)	0.000 (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 (36.28)	0.000 (47.98)	0.001 (55.75)	0.000 (51.19)	0.000 (38.68)	0.000 (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 (8.79)	0.005 (15.78)	0.001 (26.71)	0.001 (33.78)	0.000 (32.50)	0.001 (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 (1.68)	0.073 (3.53)	−0.002 (9.60)	0.000 (17.47)	0.001 (24.31)	−0.001 (23.40)

Comparison of Three Sampling Methods

	$\mu_* \backslash \sigma_*$		0.5	1.0	1.5	2.0	3.0	4.0
$E(X^2) = 1$	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

Comparison of Three Sampling Methods

	$\mu_* \backslash \sigma_*$		0.5	1.0	1.5	2.0	3.0	4.0
$E(X^3) = 0$	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

Comparison of Three Sampling Methods: CPU Time (Seconds)

$\mu_* \backslash \sigma_*$		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

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.

The results obtained from each table are as follows.

$\mathrm{E}(X)$ 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.

$\mathrm{E}(X^2)$ should be close to one because we have $\mathrm{E}(X^2) = \mathrm{V}(X) = 1$ 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 $\mathrm{E}(X)$.

$\mathrm{E}(X^3)$ 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 $E(X) = 0$, from the table a higher acceptance rate generally shows a better performance.

The high acceptance rate implies high randomness of the generated random draws. 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).

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 5.7.2 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 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 $\mathrm{E}(X)$ and $\mathrm{V}(X)$.

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.

However, we should keep in mind that MH also has the problem which choice of the sampling density is very important.

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 Algo-

rithm," *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, "Illustra-

tion 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 Sam-

pler 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.

6 Bayesian Estimation — Examples

6.1 Heteroscedasticity Model

In Section 6.1, Tanizaki and Zhang (2001) is re-computed using the random number generators.

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 consid-

ered 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).

6.1.1 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 6.1, 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.

6.1.2 Multiplicative Heteroscedasticity Regression Model

The multiplicative heteroscedasticity model discussed by Harvey (1976) can be shown as follows:

$$y_t = X_t\beta + u_t, \qquad u_t \sim N(0, \sigma_t^2), \tag{7}$$

$$\sigma_t^2 = \sigma^2 \exp(q_t\alpha), \tag{8}$$

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 (7) and (8) 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 (8) as:

$$\sigma_t^2 = \exp(z_t \gamma). \tag{9}$$

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 (9), rather than (8).

The generalized least squares (GLS) estimator of β, denoted by $\hat{\beta}_{GLS}$, is given by:

$$\hat{\beta}_{GLS} = \Big(\sum_{t=1}^{n} \exp(-z_t\gamma)X_t'X_t\Big)^{-1} \sum_{t=1}^{n} \exp(-z_t\gamma)X_t'y_t, \tag{10}$$

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 $\hat{\gamma}_{2SE}$:

$$\mathrm{E}(\hat{\gamma}_{2SE}) = \gamma - 1.2704(\sum_{t=1}^{n} z_t' z_t)^{-1} \sum_{t=1}^{n} z_t', \tag{11}$$

$$\mathrm{V}(\hat{\gamma}_{2SE}) = 4.9348(\sum_{t=1}^{n} z_t' z_t)^{-1}.$$

For the second term in equation (11), 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 (10).

Maximum Likelihood Estimator (MLE): The density of $Y_n = (y_1, y_2, \cdots, y_n)$ based on (7) and (9) is:

$$f(Y_n|\beta,\gamma) \propto \exp\left(-\frac{1}{2}\sum_{t=1}^{n}\left(\exp(-z_t\gamma)(y_t - X_t\beta)^2 + z_t\gamma\right)\right), \tag{12}$$

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:

$$\begin{aligned} \mathrm{V}(\hat{\theta}_{MLE}) &= \left(- \mathrm{E}\left(\frac{\partial^2 \log f(Y_n|\theta)}{\partial\theta\partial\theta'} \right) \right)^{-1} \\ &= \begin{pmatrix} \left(\sum_{t=1}^{n} \exp(-z_t\gamma) X_t' X_t \right)^{-1} & 0 \\ 0 & 2(\sum_{t=1}^{n} z_t' z_t)^{-1} \end{pmatrix}. \end{aligned} \tag{13}$$

Thus, from (13), 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: $V(\hat{\gamma}_{M2SE}) = 4.9348(\sum_{t=1}^{n} z_t' z_t)^{-1}$.

6.1.3 Bayesian Estimation

We assume that the prior distributions of the parameters β and γ are noninformative, which are represented by:

$$f_\beta(\beta) = \text{constant}, \qquad f_\gamma(\gamma) = \text{constant}. \tag{14}$$

Combining the prior distributions (14) and the likelihood function (12), the posterior distribution $f_{\beta\gamma}(\beta, \gamma|y)$ is obtained as follows:

$$f_{\beta\gamma}(\beta, \gamma|Y_n) \propto \exp\left(-\frac{1}{2}\sum_{t=1}^{n}\left(\exp(-z_t\gamma)(y_t - X_t\beta)^2 + z_t\gamma\right)\right).$$

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 5.7.5, 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:

$$f_{\gamma|\beta}(\gamma|\beta, Y_n) \propto \exp\left(-\frac{1}{2}\sum_{t=1}^{n}\left(\exp(-z_t\gamma)(y_t - X_t\beta)^2 + z_t\gamma\right)\right), \tag{15}$$

$$f_{\beta|\gamma}(\beta|\gamma, Y_n) = N(B_1, H_1), \tag{16}$$

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 (16) is simple since it is a k-variate normal distribution with mean B_1 and variance H_1.

However, since the J-variate distribution (15) does not take the form of any standard density, it is not easy to sample from (15).

In this case, the MH algorithm discussed in Section 5.7.3 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 (15).

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 (16).

(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 (7) and (8), however, there is asymptotically no correlation between $\hat{\beta}_{MLE}$ and $\hat{\gamma}_{MLE}$, as shown in (13).

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:

$$f_*(\gamma) = N(\gamma^+, c^2\Sigma^+), \tag{17}$$

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 (17).

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, 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 6.1.4.

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 6.1.4, we compare the two Bayesian estimators (i.e, BM2SE and BMLE) with the two traditional estimators (i.e., M2SE and MLE).

6.1.4 Monte Carlo Study

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:

$$y_t = \beta_1 + \beta_2 x_{2,t} + \beta_3 x_{3,t} + u_t, \tag{18}$$

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,

$$\sigma_t^2 = \exp(\gamma_1 + \gamma_2 x_{2,t}), \qquad \text{for } t = 1, 2, \cdots, n. \tag{19}$$

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 (18) and (19), Judge, Hill, 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 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 $(\gamma_1, \gamma_2) =$

Table 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

$(-2, 0.25)$ 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 (18), 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 6.1.2 and 6.1.3 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.

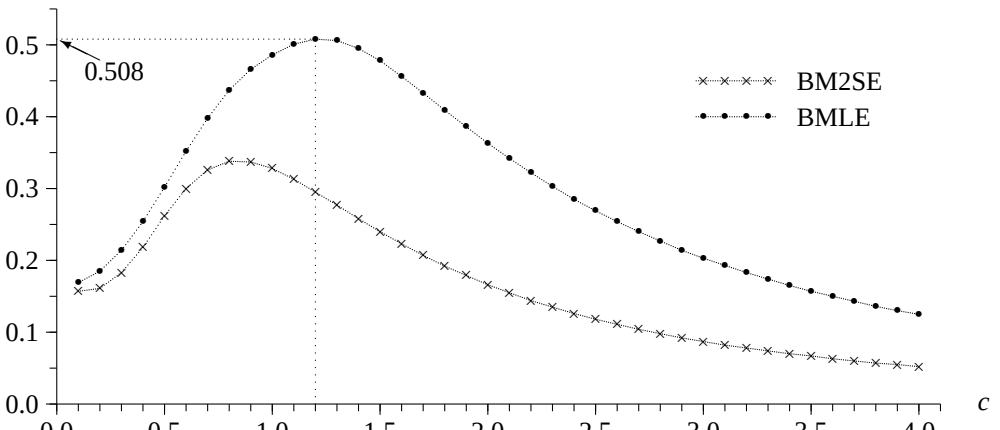
AVE and RMSE are obtained as follows:

$$\text{AVE} = \frac{1}{G}\sum_{g=1}^{G}\hat{\theta}_j^{(g)}, \qquad \text{RMSE} = \Big(\frac{1}{G}\sum_{g=1}^{G}(\hat{\theta}_j^{(g)} - \theta_j)^2\Big)^{1/2},$$

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 $\hat{\theta}_{BMLE}$.

Figure 2: Acceptance Rates in Average: $M = 5000$ and $N = 10^4$



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 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 2, 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 6.1.6).

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 2).

However, the sampling density with the largest acceptance probability is not neces-

sarily 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/`.

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 3, where BMLE is compared with M2SE and MLE.

The case of $n = 20$, $M = 5000$ and $N = 10^4$ is examined in Table 3.

A discussion on choice of M and N is given in Appendix 6.1.6, where we examine

whether $M = 5000$ and $N = 10^4$ are sufficient.

Table 3: 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

Table 3: The AVE, RMSE and Quartiles: $n = 20$ — Cont.

		β_1	β_2	β_3	γ_1	γ_2
	True Value	10	1	1	−2	0.25
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

Table 3: The AVE, RMSE and Quartiles: $n = 20$ — Cont.

		β_1	β_2	β_3	γ_1	γ_2
	True Value	10	1	1	−2	0.25
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

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 3, 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 3 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 as the regression coefficients are unbiased in the small sample.

Thus, Table 3 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 3 – 7.

Figure 3: Empirical Distributions of β_1

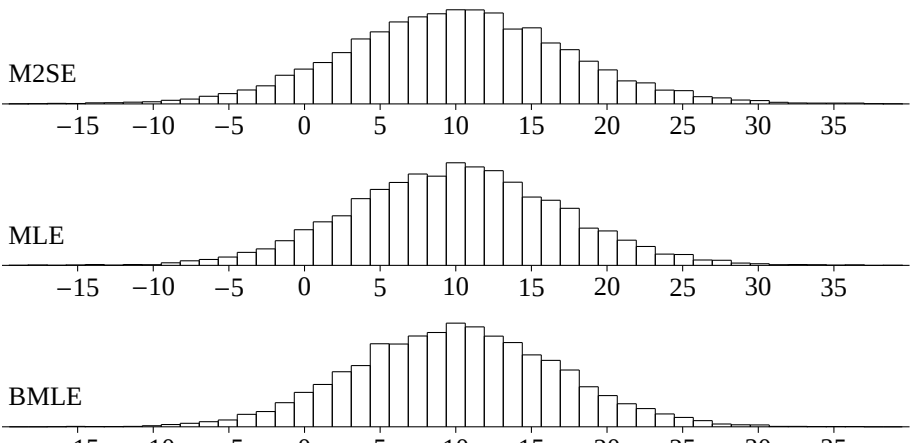


Figure 4: Empirical Distributions of β_2

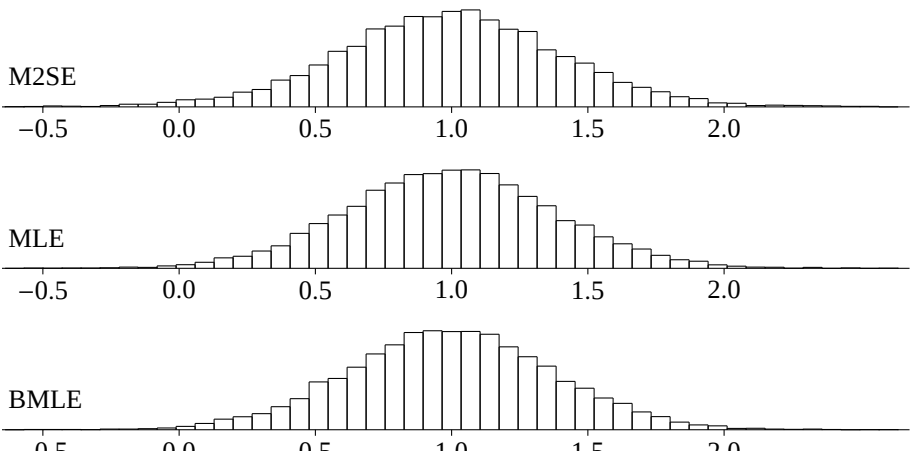


Figure 5: Empirical Distributions of β_3

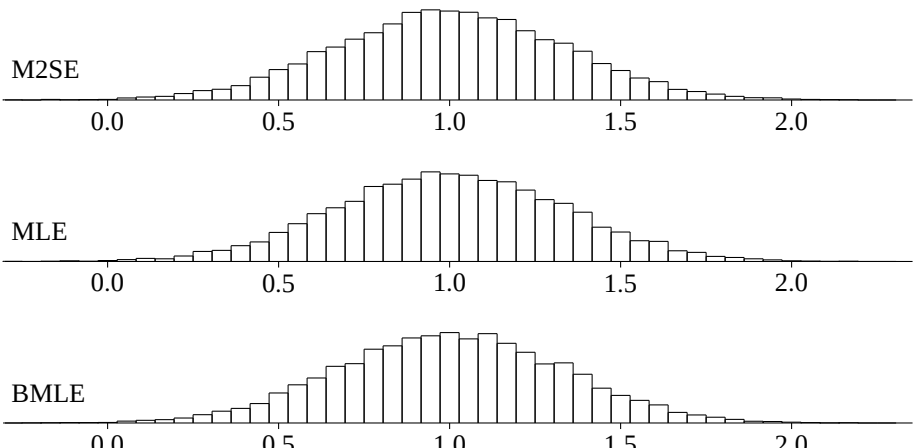


Figure 6: Empirical Distributions of γ_1

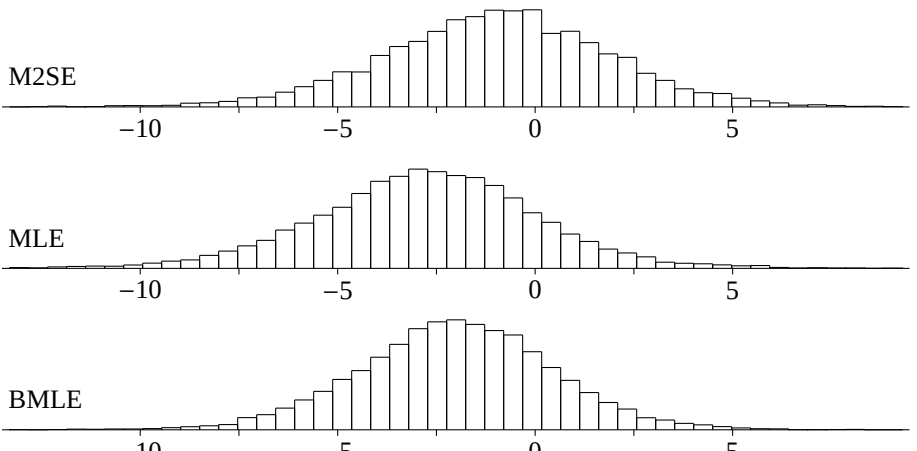
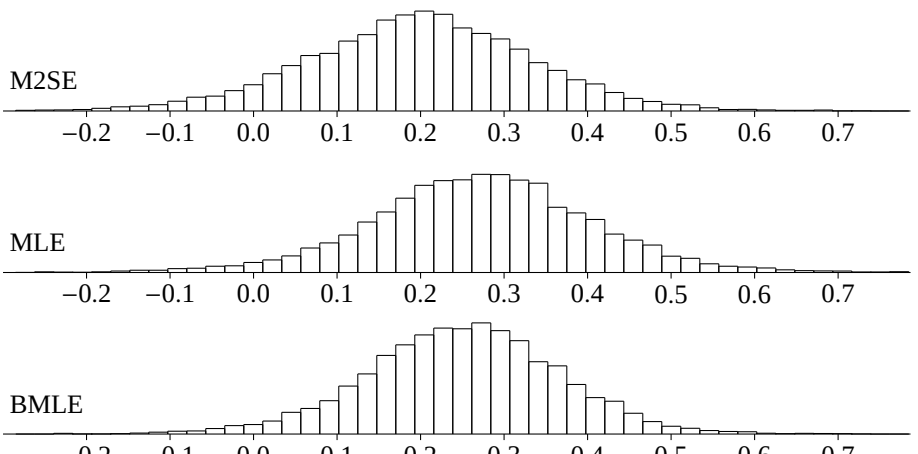


Figure 7: Empirical Distributions of γ_2



For the posterior densities of β_1 (Figure 3), β_2 (Figure 4), β_3 (Figure 5) and γ_1 (Figure 6), all of M2SE, MLE and BMLE are almost symmetric (also, see Skewness in Table 3).

For the posterior density of γ_2 (Figure 7), both MLE and BMLE are slightly skewed to the left because Skewness of γ_2 in Table 3 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 3, β_2 in Figure 4 and β_3 in Figure 5, M2SE is biased while MLE and BMLE are distributed around the true value.

For γ_1 in Figure 6 and γ_2 in Figure 7, the empirical distributions of M2SE, MLE and BMLE are quite different.

For γ_1 in Figure 6, 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 7, 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, which should be compared with BMLE in Table 3.

As a result, all the AVEs are very close to the corresponding true parameter values. Therefore, we can conclude from Tables 3 and 4 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 3, and 8.715, 0.455, 0.350, 4.449 and 0.228 in Table 4.

Thus, we can see that RMSE and IR decrease as n is large.

Table 4: 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

6.1.5 Summary

In Section 6.1, 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 6.1.

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 2.

For the sampling density, we utilize the independence chain through Section 6.1.

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 neces-

sarily 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, for M2SE, the estimates of β are unbiased but those of γ are different from the true parameter values (see Table 3).

(iii) From Table 3, 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 3 – 7.

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 3 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.

6.1.6 Appendix: Are $M = 5000$ and $N = 10^4$ Sufficient?

Table 5: 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
	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
$M = 1000$	Kurtosis	3.542	3.358	2.963	3.950	4.042
$N = 10^4$	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

Table 5: BMLE: $n = 20$ and $c = 2.0$ — Cont.

		β_1	β_2	β_3	γ_1	γ_2
	True Value	10	1	1	−2	0.25
	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
$M = 5000$	Kurtosis	3.498	3.347	2.961	3.764	3.840
$N = 5000$	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

In Section 6.1.4, 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 6.1.4 (see p.516 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 5, 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 3 should be compared with Table 5.

From Tables 3 and 5, 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 6.1.4.

We take the case of $M = 5000$ and $N = 10^4$ for safety in Section 6.1.4, although we obtain the results that both $M = 1000$ and $N = 5000$ are large enough.

6.2 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 estima-

tor, 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.

6.2.1 Introduction

In Section 6.2, 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 6.2 that BE is superior to MLE because BEs of both the auto-correlation coefficient and the variance of the error term are closer to the true values, compared with MLEs.

6.2.2 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 :

$$f(Y_n|\beta,\rho,\sigma_\epsilon^2) = f(y_1|\beta,\rho,\sigma_\epsilon^2)\prod_{t=2}^{n} f(y_t|Y_{t-1},\beta,\rho,\sigma_\epsilon^2)$$

$$= (2\pi\sigma_\epsilon^2)^{-n/2}(1-\rho^2)^{1/2}\exp\Big(-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^{n}(y_t^* - X_t^*\beta)^2\Big), \qquad (20)$$

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 (20).

Maximizing equation (20) with respect to β and σ_ϵ^2, we obtain the following expressions:

$$\hat{\beta} \equiv \hat{\beta}(\rho) = (\sum_{t=1}^{n} X_t^{*\prime} X_t^*)^{-1} \sum_{t=1}^{n} X_t^{*\prime} y_t^*,$$

$$\hat{\sigma}_\epsilon^2 \equiv \hat{\sigma}_\epsilon^2(\rho) = \frac{1}{n}\sum_{t=1}^{n}(y_t^* - X_t^*\hat{\beta})^2. \tag{21}$$

By substituting $\hat{\beta}$ and $\hat{\sigma}_\epsilon^2$ into β and σ_ϵ^2 in equation (20), we have the concentrated likelihood function:

$$f(Y_n|\hat{\beta}, \rho, \hat{\sigma}_\epsilon^2) = \left(2\pi\hat{\sigma}_\epsilon^2(\rho)\right)^{-n/2}(1-\rho^2)^{1/2}\exp(-\frac{n}{2}), \tag{22}$$

which is a function of ρ.

Equation (22) 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 (22) 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 $V(\hat{\beta}) = \sigma^2(\sum_{t=1}^{n} X_t^{*\prime} X_t^*)^{-1}$ 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 $V(\hat{\beta})$ is also underestimated, which yields incorrect inference on the regression coefficient β.

Thus, unless ρ is properly estimated, the estimate of $V(\hat{\beta})$ 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 exam-

ple, 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 (21).

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:

$$f_\beta(\beta) \propto \text{constant}, \qquad \text{for } -\infty < \beta < \infty, \tag{23}$$

$$f_\rho(\rho) \propto \text{constant}, \qquad \text{for } -1 < \rho < 1, \tag{24}$$

$$f_{\sigma_\epsilon}(\sigma_\epsilon^2) \propto \frac{1}{\sigma_\epsilon^2}, \qquad \text{for } 0 < \sigma_\epsilon^2 < \infty. \tag{25}$$

In equation (24), 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 6.1 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 (20) and (23) – (25), 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:

$$
\begin{aligned}
& f_{\beta\rho\sigma_\epsilon}(\beta, \rho, \sigma_\epsilon^2|Y_n) \\
& \quad \propto f(Y_n|\beta, \rho, \sigma_\epsilon^2) f_\beta(\beta) f_\rho(\rho) f_{\sigma_\epsilon}(\sigma_\epsilon^2) \\
& \quad \propto (\sigma_\epsilon^2)^{-(n/2+1)} (1-\rho^2)^{1/2} \exp\Big(-\frac{1}{2\sigma_\epsilon^2} \sum_{t=1}^{n} (y_t^* - X_t^* \beta)^2\Big). \qquad (26)
\end{aligned}
$$

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 (26), 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:

$$f_{\beta|\rho\sigma_\epsilon}(\beta|\rho, \sigma_\epsilon^2, Y_n)$$

$$\propto f_{\beta\rho\sigma_\epsilon}(\beta, \rho, \sigma_\epsilon^2|Y_n) \propto \exp\Big(-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^{n}(y_t^* - X_t^*\beta)^2\Big)$$

$$= \exp\Big(-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^{n}\Big((y_t^* - X_t^*\hat{\beta}) - X_t(\beta - \hat{\beta})\Big)^2\Big)$$

$$= \exp\Big(-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^{n}(y_t^* - X_t^*\hat{\beta})^2 - \frac{1}{2\sigma_\epsilon^2}(\beta - \hat{\beta})'(\sum_{t=1}^{n} X_t^{*\prime} X_t^*)(\beta - \hat{\beta})\Big)$$

$$\propto \exp\Big(-\frac{1}{2}(\beta - \hat{\beta})'(\frac{1}{\sigma_\epsilon^2}\sum_{t=1}^{n} X_t^{*\prime} X_t^*)(\beta - \hat{\beta})\Big), \tag{27}$$

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, (27) 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:

$$f_{\rho|\beta\sigma_\epsilon}(\rho|\beta, \sigma_\epsilon^2, Y_n) \propto f_{\beta\rho\sigma_\epsilon}(\beta, \rho, \sigma_\epsilon^2|Y_n)$$
$$\propto (1-\rho^2)^{1/2} \exp\Big(-\frac{1}{2\sigma_\epsilon^2} \sum_{t=1}^n \big(y_t^* - X_t^*\beta\big)^2\Big), \qquad (28)$$

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 (28) 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:

$$\begin{aligned} f_{\sigma_\epsilon|\beta\rho}(\sigma_\epsilon^2|\beta,\rho,Y_n) &\propto f_{\beta\rho\sigma_\epsilon}(\beta,\rho,\sigma_\epsilon^2|Y_n) \\ &\propto \frac{1}{(\sigma_\epsilon^2)^{n/2+1}} \exp\Big(-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^{n}(y_t^* - X_t^*\beta)^2\Big), \end{aligned} \tag{29}$$

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 $\epsilon_t = y_t^* - X_t^*\beta$.

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 $\sigma^2_{\epsilon,i}$ be the ith random draws of β, ρ and σ^2_ϵ.

Take the initial values of $(\beta, \rho, \sigma^2_\epsilon)$ as $(\beta_{-M}, \rho_{-M}, \sigma^2_{\epsilon,-M})$.

(ii) From equation (27), generate β_i given ρ_{i-1}, $\sigma^2_{\epsilon,i-1}$ and Y_n, using $\beta \sim N(\hat{\beta}, \sigma^2_{\epsilon,i-1}(\sum_{t=1}^n X_t^{*\prime} X_t^*)^{-1})$, 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 (28), generate ρ_i given β_i, $\sigma^2_{\epsilon,i-1}$ and Y_n.

Since it is not easy to generate random draws from (27), 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:

$$\begin{aligned}\omega(\rho_{i-1}, \rho^*) &= \min\left(\frac{f_{\rho|\beta\sigma_\epsilon}(\rho^*|\beta_i, \sigma^2_{\epsilon,i-1}, Y_n)/\, f_*(\rho^*|\rho_{i-1})}{f_{\rho|\beta\sigma_\epsilon}(\rho_{i-1}|\beta_i, \sigma^2_{\epsilon,i-1}, Y_n)/\, f_*(\rho_{i-1}|\rho^*)},\ 1\right) \\ &= \min\left(\frac{f_{\rho|\beta\sigma_\epsilon}(\rho^*|\beta_i, \sigma^2_{\epsilon,i-1}, Y_n)}{f_{\rho|\beta\sigma_\epsilon}(\rho_{i-1}|\beta_i, \sigma^2_{\epsilon,i-1}, Y_n)},\ 1\right).\end{aligned}$$

(b) Set $\rho_i = \rho^*$ with probability $\omega(\rho_{i-1}, \rho^*)$ and $\rho_i = \rho_{i-1}$ otherwise.

(iv) From equation (29), 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.

6.2.3 Monte Carlo Experiments

For the exogenous variables, again we take the data used in Section 6.1, in which the true data generating process (DGP) is presented in Judge, Hill, Griffiths and Lee (1980, p.156).

As in equation (18), the DGP is defined as:

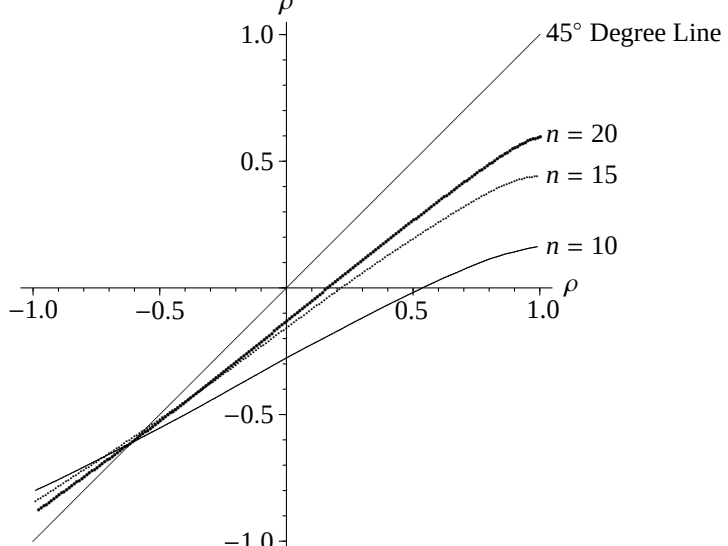
$$y_t = \beta_1 + \beta_2 x_{2,t} + \beta_3 x_{3,t} + u_t, \qquad u_t = \rho u_{t-1} + \epsilon_t, \tag{30}$$

where ϵ_t, $t = 1, 2, \cdots, n$, are normally and independently distributed with $\mathrm{E}(\epsilon_t) = 0$ and $\mathrm{E}(\epsilon_t^2) = \sigma_\epsilon^2$.

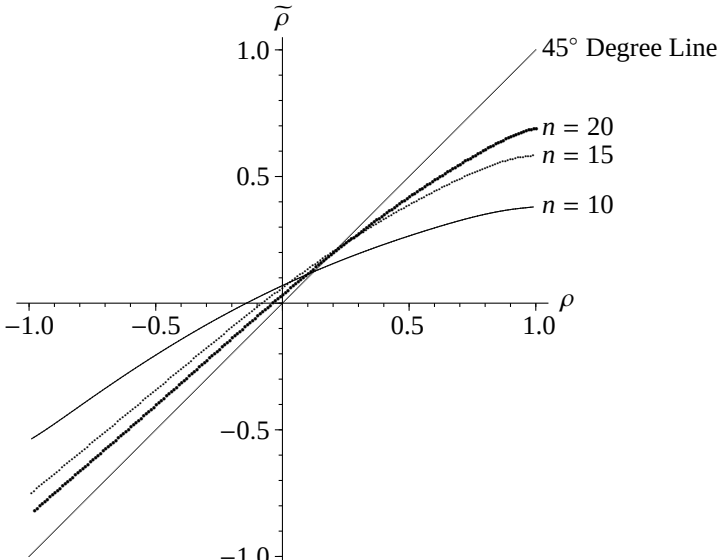
As in Judge, Hill, Griffiths and Lee (1980), the parameter values are set to be $(\beta_1, \beta_2, \beta_3) = (10, 1, 1)$.

We utilize $x_{2,t}$ and $x_{3,t}$ given in Judge, Hill, Griffiths and Lee (1980, pp.156), which is shown in Table 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


ρ
1.0
0.5
−1.0
−0.5
0.5
1.0
ρ
−0.5
−1.0
45° Degree Line
n = 20
n = 15
n = 10

——— $M = 5000$ and $N = 10^4$ ———



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

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

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

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

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 (30), 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 6.2.2, 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:

$$\text{AVE} = \frac{1}{G}\sum_{g=1}^{G}\hat{\theta}_j^{(g)}, \qquad \text{RMSE} = \Big(\frac{1}{G}\sum_{g=1}^{G}(\hat{\theta}_j^{(g)} - \theta_j)^2\Big)^{1/2},$$

for $j = 1, 2, \cdots, 5$, where θ_j denotes the jth element of θ and $\hat{\theta}_j^{(g)}$ represents the 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 6.2.3, we compare the Bayesian estimator (BE) with the maximum likelihood estimator (MLE) through Monte Carlo studies.

In Figures 8 and 9, we focus on the estimates of the autocorrelation coefficient ρ.

In Figure 8 we draw the relationship between ρ and $\hat{\rho}$, where $\hat{\rho}$ denotes the arithmetic average of the 10^4 MLEs, while in Figure 9 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 9 (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 8 and 9.

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 8), 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 9).

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 2 – 5 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 2 is based on the MLEs while Tables 3 – 5 are obtained from the BEs.

Figure 10: Empirical Distributions of β_1

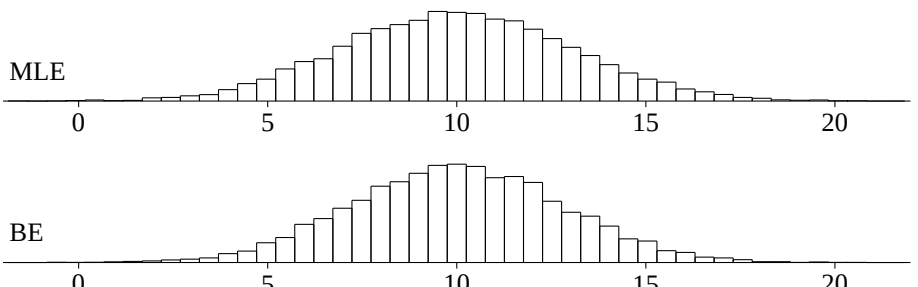


Figure 11: Empirical Distributions of β_2

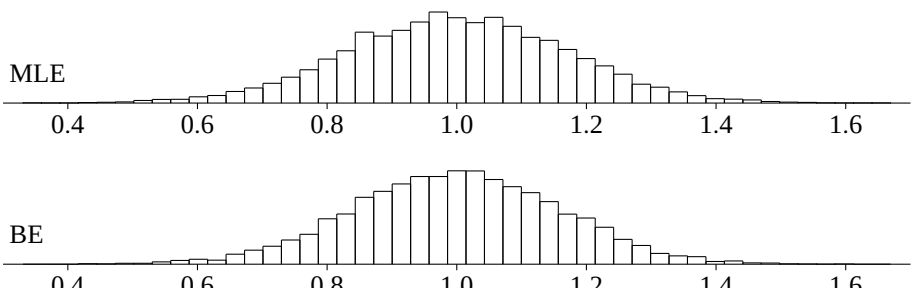


Figure 12: Empirical Distributions of β_3

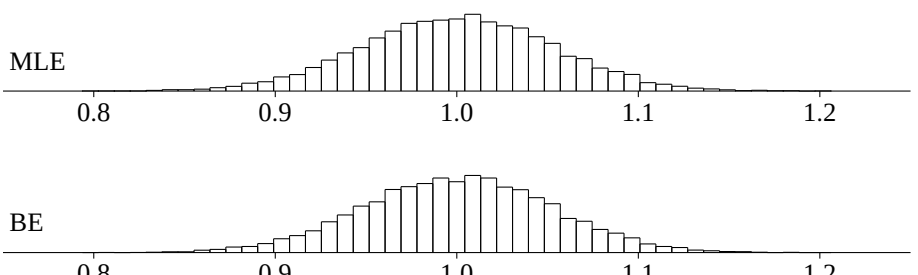


Figure 13: Empirical Distributions of ρ

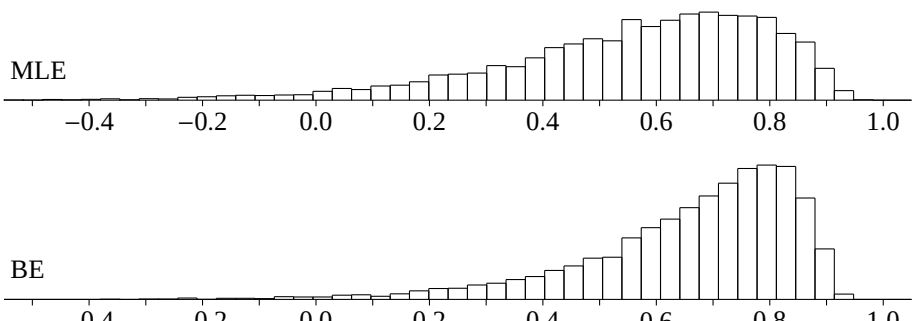
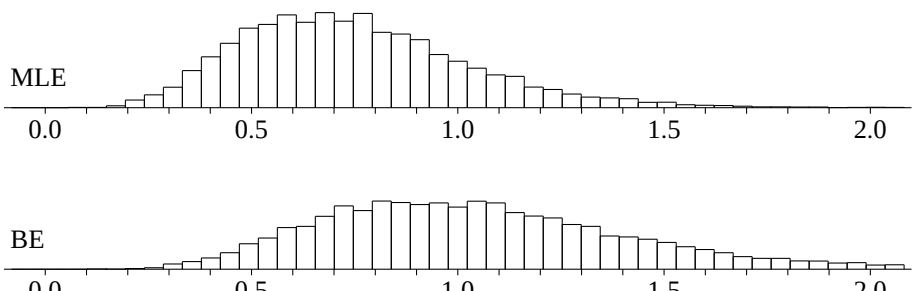


Figure 14: Empirical Distributions of σ_ϵ^2



To check whether M and N are enough large, Tables 3 – 5 are shown for BE.

Comparison between Tables 3 and 4 shows whether $N = 5000$ is large enough and we can see from Tables 3 and 5 whether the burn-in period $M = 1000$ is large enough.

We can conclude that $N = 5000$ is enough if Table 3 is very close to Table 4 and that $M = 1000$ is enough if Table 3 is close to Table 5.

The difference between Tables 3 and 4 is at most 0.034 (see 90% in β_1) and that between Tables 3 and 5 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 2 and 3.

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 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 2) and that of BE is 1.051 (see Table 3).

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 2) while those of BE are 0.380 and 0.384 (see Table 3).

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 2) and that of BE is given by 0.661 (Table 3).

As it is also seen in Figures 8 and 9, 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 2 and -1.389 in Table 3.

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 3 is 5.391 and that in Table 2 is 4.013, the empirical distri-

bution of BE has more kurtosis than that of MLE.

Figures 10 – 14 correspond to the empirical distributions for each parameter, which are constructed from the G estimates used in Tables 2 and 3.

As we can see from Skewness and Kurtosis in Tables 2 and 3, $\hat{\beta}_i$ and $\widetilde{\beta}_i$, $i = 1, 2, 3$, are very similar to normal distributions in Figures 10 – 12.

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 2 and 3.

In Figure 13, 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 14 displays the empirical distributions of σ_ϵ^2. MLE $\hat{\sigma}_\epsilon^2$ is biased and under-estimated, 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.

6.2.4 Summary

In Section 6.2, 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 6.2, 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 8 we have the relationship between ρ and $\hat{\rho}$, and $\widetilde{\rho}$ corresponding to ρ is drawn in Figure 9.

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 2 and 3, 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 An-*

nals of Statistics, Vol.22, No.4, pp.1701 – 1762.

Zellner, A., 1971, *An Introduction to Bayesian Inference in Econometrics*, John Wiley & Sons.

6.3 Marginal Likelihood, Convergence Diagnostic and so on

6.3.1 Marginal Likelihood (周辺尤度)

Model Selection $\Longrightarrow$ Marginal Likelihood

$$f_y(y) = \int f_{y|\theta}(y|\theta)\, f_\theta(\theta) \mathrm{d}\theta$$

Evaluation of Marginal Likelihood $\Longrightarrow$ Proper Prior

(i) Importance Sampling: Use of Prior Distribution

$$f_y(y) = \mathrm{E}_\theta(f_{y|\theta}(y|\theta)) \approx \frac{1}{N}\sum_{i=1}^{N} f_{y|\theta}(y|\theta_i),$$

where θ_i is the ith random draw generated from the prior distribution $f_\theta(\theta)$.

(ii) Importance Sampling: Use of the Appropriate Importance Distribution

$$f_y(y) = \int \frac{f_{y|\theta}(y|\theta)\, f_\theta(\theta)}{g(\theta)} g(\theta) \mathrm{d}\theta = \mathrm{E}\Big(\frac{f_{y|\theta}(y|\theta)\, f_\theta(\theta)}{g(\theta)}\Big)$$
$$\approx \frac{1}{N} \sum_{i=1}^{N} \frac{f_{y|\theta}(y|\theta_i)\, f_\theta(\theta_i)}{g(\theta_i)},$$

where θ_i is the ith random draw generated from the appropriately chosen importance distribution $g(\theta)$.

(iii) Harmonic Mean $\Longrightarrow$ Gelfand and Dey (1994) and Newton and Raftery (1994)

$$\frac{1}{f_y(y)} = \int \frac{g(\theta)}{f_y(y)} \mathrm{d}\theta = \int \frac{g(\theta)}{f_y(y)\, f_{\theta|y}(\theta|y)} f_{\theta|y}(\theta|y) \mathrm{d}\theta$$
$$= \int \frac{g(\theta)}{f_{y|\theta}(y|\theta)\, f_\theta(\theta)} f_{\theta|y}(\theta|y) \mathrm{d}\theta \approx \frac{1}{N} \sum_{i=1}^{N} \frac{g(\theta_i)}{f_{y|\theta}(y|\theta_i)\, f_\theta(\theta_i)},$$

where θ_i is the ith random draw generated from the posterir distribution $f_{\theta|y}(\theta|y)$. Thus, the marginal distribution is evaluated by:

$$f_y(y) \approx \left(\frac{1}{N} \sum_{i=1}^{N} \frac{g(\theta_i)}{f_{y|\theta}(y|\theta_i)\, f_\theta(\theta_i)} \right)^{-1}, \qquad \Longrightarrow \qquad \text{Gelfand and Dey (1994).}$$

When $g(\theta) = f_\theta(\theta)$ is taken, the marginal distribution is given by:

$$f_y(y) \approx \left(\frac{1}{N} \sum_{i=1}^{N} \frac{1}{f_{y|\theta}(y|\theta_i)} \right)^{-1}, \qquad \Longrightarrow \qquad \text{Newton and Raftery (1994).}$$

(iv) Chib (1995) and Chib and Jeliazkov (2001)

$$f_y(y) = \frac{f_{y|\theta}(y|\theta) f_\theta(\theta)}{f_{\theta|y}(\theta|y)}$$

$$\log f_y(y) = \log f_{y|\theta}(y|\hat{\theta}) + \log f_\theta(\hat{\theta}) - \log f_{\theta|y}(\hat{\theta}|y),$$

where $\hat{\theta}$ denotes the Bayes estimates.

We need to evaluate $\log f_{\theta|y}(\hat{\theta}|y)$, using the Gibbs sampler or the MH algorithm.

6.3.2 Convergence Diagnostic (収束判定)

We need to check whether the **burn-in period** is enough and whether MCMC converges to the **invariant distribution (不変分布)**.

Geweke (1992)

Divide the sample path into three periods, excluding the burn-in period..

Test whether the first period is different from the third period.

Suppose that we have the MCMC sequence, i.e., $\theta_{-M+1}, \cdots, \theta_0, \theta_1, \cdots, \theta_N$.

The burn-in period is denoted by $\theta_{-M+1}, \cdots, \theta_0$.

$\theta_1, \cdots, \theta_N$ are divided by three periods.

The first period is given by $\theta_1, \cdots, \theta_{N_1}$.

The second period is given by $\theta_{N_1+1}, \cdots, \theta_{N_2}$.

The third period is given by $\theta_{N_2+1}, \cdots, \theta_N$.

Consider a function $g(\cdot)$.

Define $\bar{g}_1 = \frac{1}{N_1}\sum_{i=1}^{N_1} g(\theta_i)$ and $\bar{g}_3 = \frac{1}{N_3}\sum_{i=N_1+N_2+1}^{N} g(\theta_i)$ for $N_3 = N - N_2 - N_1$.

Estimate $\frac{1}{N_1}\mathrm{V}(\sum_{i=1}^{N_1} g(\theta_i))$ and $\frac{1}{N_3}\mathrm{V}(\sum_{i=N_1+N_2+1}^{N} g(\theta_i))$,
which are denoted by s_1^2 and s_3^2, respectively.

By the central limit theorem,

$$\frac{\overline{g}_1 - \mathrm{E}(\overline{g}_1)}{s_1/\sqrt{N_1}} \longrightarrow N(0,1) \qquad \text{and} \qquad \frac{\overline{g}_3 - \mathrm{E}(\overline{g}_3)}{s_3/\sqrt{N_3}} \longrightarrow N(0,1).$$

Therefore, under the null hypothesis $H_0 :\ \mathrm{E}(\overline{g}_1) = \mathrm{E}(\overline{g}_3)$,

$$\frac{\overline{g}_1 - \overline{g}_3}{\sqrt{s_1^2/N_1 + s_3^2/N_3}} \longrightarrow N(0,1).$$

The case of $g(\theta_i) = \theta_i \implies$ Testing whether the two means (i.e., first-moments) are equal.

The case of $g(\theta_i) = \theta_i^2 \implies$ Testing whether the two second-moments are equal.

Computation of s_1^2 and s_3^2 has to be careful, because $g(\theta_1), \cdots, g(\theta_N)$ are serially correlated.

$\Longrightarrow$ Long-run variance.

Take an example of s_1^2, which is an estimate of $\frac{1}{N_1}\mathrm{V}(\sum_{i=1}^{N_1} g(\theta_i))$.

$$\frac{1}{N_1}\mathrm{V}\left(\sum_{i=1}^{N_1} g(\theta_i)\right) = \frac{1}{N_1}\sum_{i=1}^{N_1}\sum_{j=1}^{N_1}\mathrm{Cov}(g(\theta_i), g(\theta_j))$$

$$= \frac{1}{N_1}(N_1\gamma(0) + 2(N_1-1)\gamma(1) + 2(N_1-2)\gamma(2) + \cdots + 2\gamma(N_1-1))$$

$$= \gamma(0) + 2\sum_{\tau=1}^{N_1-1} k(\frac{\tau}{N_1})\gamma(\tau), \quad \Longrightarrow \quad \text{Bartlett Kernel (Newy-West Est.)}$$

where $\gamma(\tau) = \mathrm{Cov}(g(\theta_i), g(\theta_{i+\tau}))$.

We may choose the other kernels (for example, Parzen kernel or second-order spectrum kernel; see p.166-167) for $k(x)$.

Thus, s_1^2 is estimated by:

$$s_1^2 = \hat{\gamma}(0) + 2 \sum_{\tau=1}^{q} k(\frac{\tau}{q+1})\hat{\gamma}(\tau),$$

for $q \leq N_1 - 1$. $\Longrightarrow$ Choice of q and $k(\cdot)$.