應用財務計量經濟學- HW1

魏上傑

2023-03-18

1.4

若 Y_t 為嚴格定態 (strict stationary)，說明 Y_t 為具有相同分配之隨機變數。i.e., strict stationary $\implies$ identical distributions.

Let $c_1, ..., c_k$ be constants. Then by definition of strict stationarity, we have

$$P(X_{t_1} \leq c_1, ..., X_{t_k} \leq c_k) = P(X_{t_1+h} \leq c_1, ..., X_{t_k+h} \leq c_k),$$

$\forall k = 1, 2, ...$ and $\forall h = 0, \pm 1, \pm 2, ...$

In particular, we have

$$P(X_{t_1} \leq c_1) = P(X_{t_1+h} \leq c_1)$$

, by choosing $t_k = t_1$. This is true for any t_1 and h.

$\therefore$ if a sequence is strictly stationary, then all the random variables in the sequence have the same distribution.

2.2

在台灣央行下載外匯存底資料(1987M5~2021M12, 單位: 百萬美元)

(a). 讀入資料，命名為 FR

```
library(readxl)
fx <- read_excel("Reserves.xlsx")
colnames(fx) <- c("year","FR")

fx$FR <- fx$FR*(10^6) # 轉為單位美元
```

```
fx$year <- NULL # 原先年資料難以操作 (民國年)，所以刪除
# 新建西元年資料
fx$date <- seq(as.Date("1987-01-01"), as.Date("2022-12-01"), by="month")
fx <- fx[order(fx$date),]
```

(b). 將 FR 取對數，命名為 LFR

```r
fx$LFR <- log(fx$FR)
```

```r
str(fx)
```

```
## tibble [432 x 3] (S3: tbl_df/tbl/data.frame)
##  $ FR  : num [1:432] 4.91e+10 5.18e+10 5.45e+10 5.75e+10 5.99e+10 ...
##  $ date: Date[1:432], format: "1987-01-01" "1987-02-01" ...
##  $ LFR : num [1:432] 24.6 24.7 24.7 24.8 24.8 ...
```

(c). 計算外匯存底年增率，命名為 D12FR

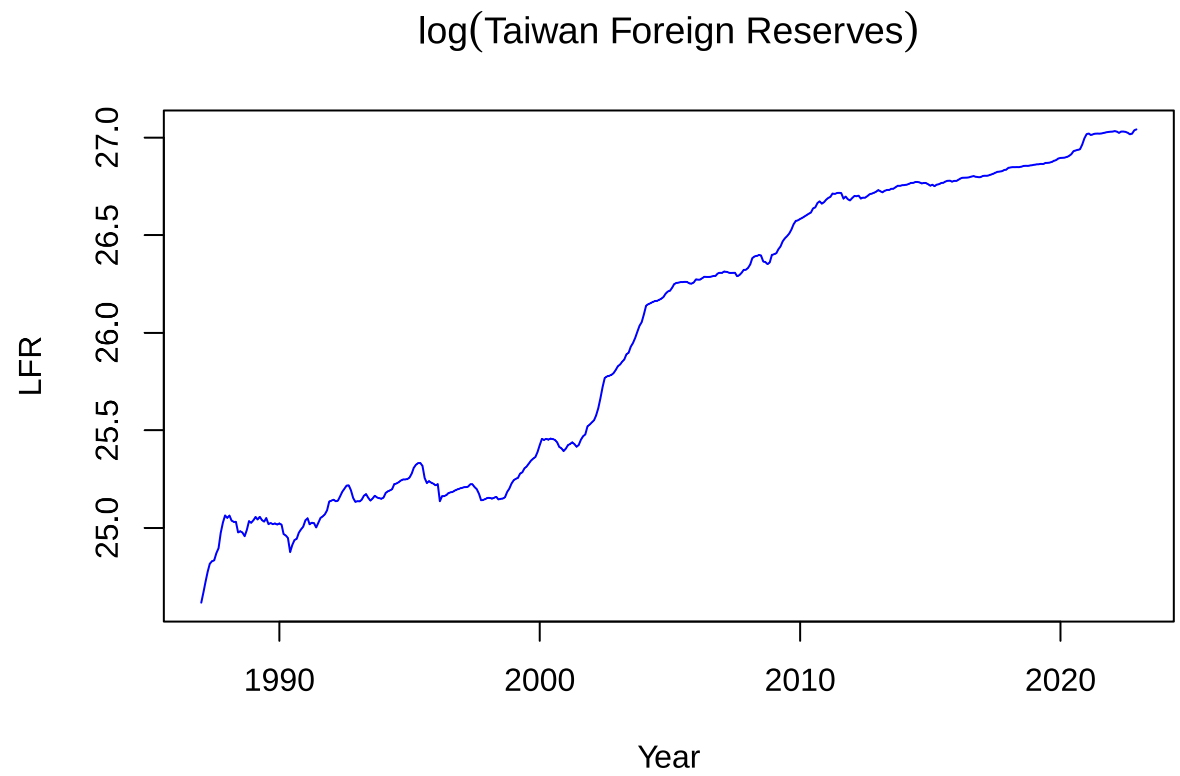
```r
library(dplyr, quietly = TRUE)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

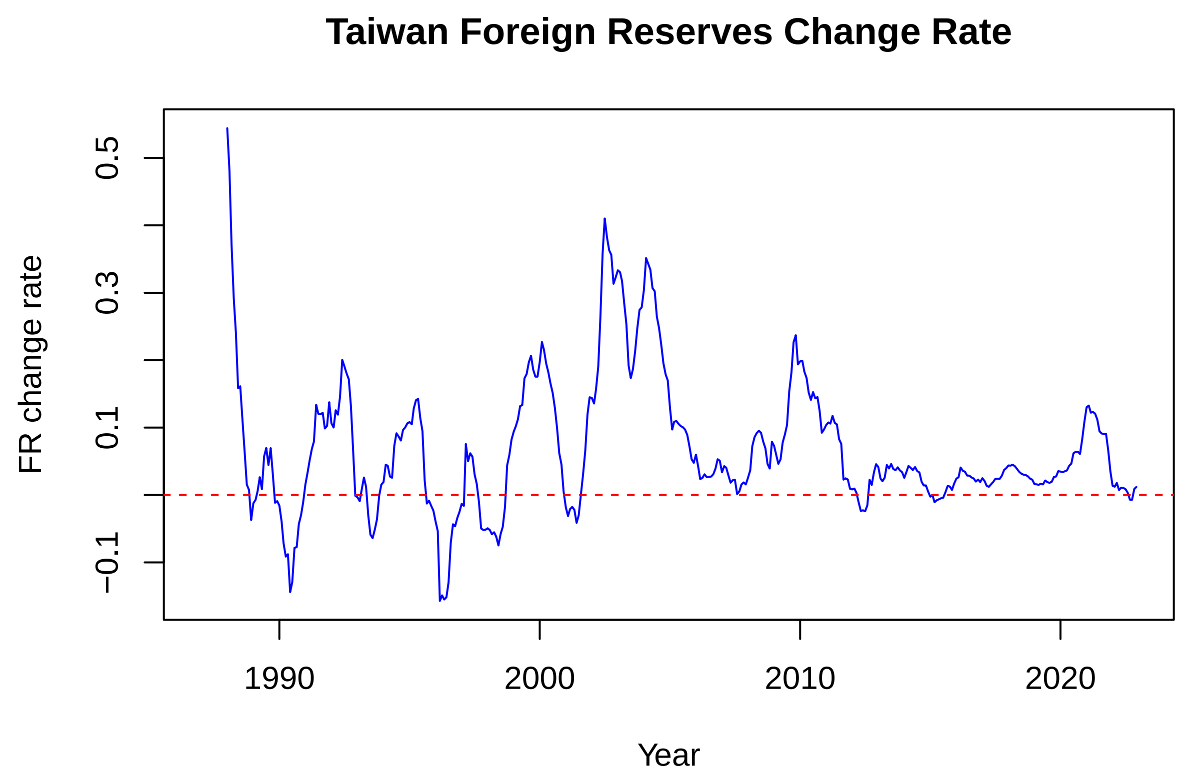
```r
fx$prev_year_FR <- lag(fx$FR, 12)

fx$D12FR <- (fx$FR-fx$prev_year_FR)/fx$prev_year_FR
```

(d). 畫出 LFR, D12FR

```r
plot(y=fx$LFR, x=fx$date,type="l", col="blue",
     xlab="Year", ylab="LFR",
     main=expression(log("Taiwan Foreign Reserves")))
```




```
plot(y=fx$D12FR, x=fx$date,type="l", col="blue",
     xlab="Year", ylab="FR change rate",
     main="Taiwan Foreign Reserves Change Rate")
abline(h=0, col="red", lty="dashed")
```



(e). 建構虛擬變數 Dum

```
fx$Dum=0
fx$Dum <- ifelse(fx$D12FR>=0,1,0)

mean(fx$Dum, na.rm=T)
```

```
## [1] 0.8142857
```

3.1

Y_t 是 MA(1) 序列：

$$Y_t = \epsilon_t + \theta\epsilon_{t-1}, \epsilon_t \sim WN(0, \sigma^2)$$

(a) 計算 $\gamma(j), j \geq 0$

- if j=0:

$$\begin{aligned}\gamma(0) &= Cov(Y_t, Y_t) \\ &= Cov(\epsilon_t + \theta\epsilon_{t-1}, \epsilon_t + \theta\epsilon_{t-1}) \\ &= Var(\epsilon_t) + \theta^2 Var(\epsilon_{t-1}) + 2\theta Cov(\epsilon_t, \epsilon_{t-1}) \\ &= \sigma^2 + \theta^2\sigma^2 + 2\theta[E(\epsilon_t\epsilon_{t-1}) - E(\epsilon_t)E(\epsilon_{t-1})] \\ &= \sigma^2 + \theta^2\sigma^2\end{aligned}$$

- if j=1:

$$\gamma(1) = Cov(Y_t, Y_{t-1}) = Cov(\epsilon_t + \theta\epsilon_{t-1}, \epsilon_{t-1} + \theta\epsilon_{t-2}) = \theta\sigma^2$$

- if j=2:

$$\gamma(1) = Cov(Y_t, Y_{t-2}) = Cov(\epsilon_t + \theta\epsilon_{t-1}, \epsilon_{t-2} + \theta\epsilon_{t-3}) = 0$$

Therefore,

$$\gamma(j) = \begin{cases} \sigma^2 + \theta^2\sigma^2, & \text{if } j = 0 \\ \theta\sigma^2, & \text{if } j = 1 \\ 0, & \text{if } j > 1 \end{cases}$$

(b) Y_t 是否定態?

Y_t 是 weakly stationary, 因為 $E(Y_t) = 0, \forall t, Var(Y_t) = \sigma^2 + \theta^2\sigma^2 < \infty$, 並且 Autocovariance 只與"j"有關

事實上，finite-order MA process 都是定態。

3.2

$\epsilon_t \sim WN(0, \sigma^2)$

(I)

$$Y_t = 1.2Y_{t-1} - 0.2Y_{t-2} + \epsilon_t = 1.2LY_t - 0.2L^2Y_t + \epsilon_t \implies (1 - 1.2L + 0.2L^2)Y_t = \epsilon_t$$

```
polyroot(z=c(1,-1.2,0.2))
```

```
## [1] 1+0i 5-0i
```

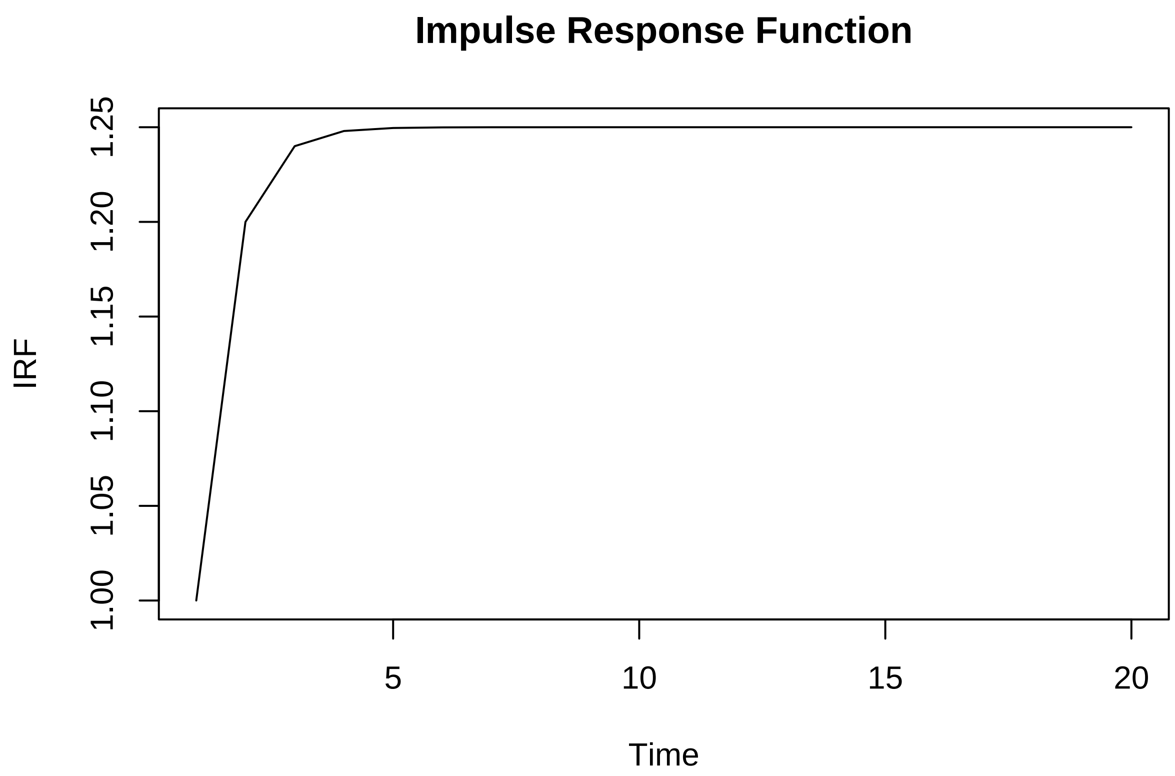
```
abs(polyroot(z=c(1,-1.2,0.2)))
```

```
## [1] 1 5
```

兩個根分別為 1,5，沒有全部大於 1，因此非定態。

```
set.seed(123)
e <- rnorm(n = 200, mean = 0, sd = 1)

y <- numeric(length = 200)
y.1 <- 0
y.2 <- 0
phi1 <- 1.2
phi2 <- -0.2

for (i in 1:length(y)) {
  y[i] <- phi1*y.1 + phi2*y.2 + e[i]
  y.2 <- y.1
  y.1 <- y[i]
}

# 計算 IRF
irf <- numeric(length = 20)
irf[1] <- 1
```

```r
for (i in 2:length(irf)) {
  if (i == 2) {
    irf[i] <- phi1
  } else {
    irf[i] <- irf[i-1]*phi1 + irf[i-2]*phi2
  }
}

# 繪製 IRF
plot(irf, type = "l", main = "Impulse Response Function",
     xlab = "Time", ylab = "IRF")
```



(II)

$$Y_t = 1.2Y_{t-1} - 0.4Y_{t-2} + \epsilon_t = 1.2LY_t - 0.4L^2Y_t + \epsilon_t \implies (1 - 1.2L + 0.4L^2)Y_t = \epsilon_t$$

```r
polyroot(z=c(1,-1.2,0.4))
```

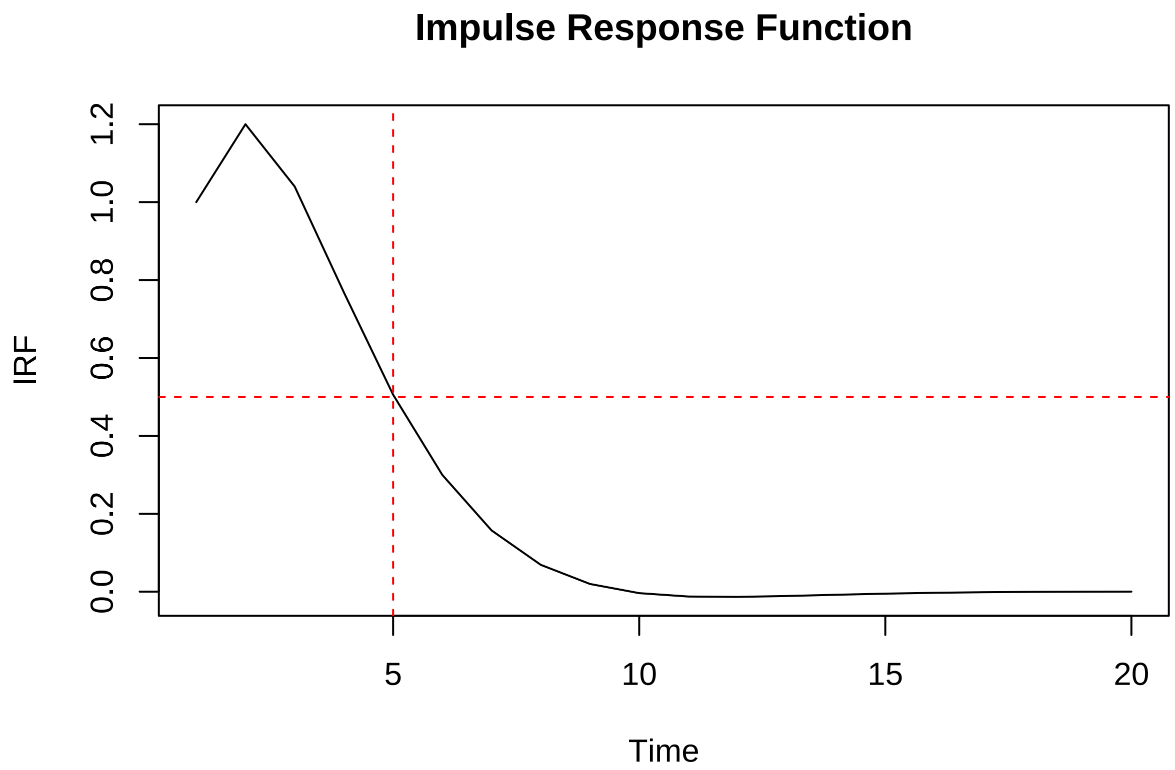
```
## [1] 1.5+0.5i 1.5-0.5i
```

```
abs(polyroot(z=c(1,-1.2,0.4)))
```

```
## [1] 1.581139 1.581139
```

兩個根的絕對值 >1，因此定態。

```
set.seed(123)
e <- rnorm(n = 200, mean = 0, sd = 1)

y <- numeric(length = 200)
y.1 <- 0
y.2 <- 0
phi1 <- 1.2
phi2 <- -0.4

for (i in 1:length(y)) {
  y[i] <- phi1*y.1 + phi2*y.2 + e[i]
  y.2 <- y.1
  y.1 <- y[i]
}

# 計算 IRF
irf <- numeric(length = 20)
irf[1] <- 1
for (i in 2:length(irf)) {
  if (i == 2) {
    irf[i] <- phi1
  } else {
    irf[i] <- irf[i-1]*phi1 + irf[i-2]*phi2
  }
}

# 繪製 IRF
plot(irf, type = "l", main = "Impulse Response Function",
     xlab = "Time", ylab = "IRF")

abline(h=0.5, lty=2, col="red")
abline(v=5, lty=2, col="red")
```




```
# 計算半衰期
distance=1
i_min <- 1
for (i in 1:length(irf)){
  if (abs(irf[i]-0.5)<distance){
    distance <- abs(irf[i]-0.5)
    i_min <- i
  }
}

i_min
```

```
## [1] 5
```

```
irf[i_min-1]
```

```
## [1] 0.768
```

```
irf[i_min]
```

```
## [1] 0.5056
```

```r
irf[i_min+1]
```

```
## [1] 0.29952
```

半衰期約是在 t=5 時。

(III)

$$Y_t = 1.2Y_{t-1} + 1.2Y_{t-2} + \epsilon_t = 1.2LY_t + 1.2L^2Y_t + \epsilon_t \implies (1 - 1.2L - 1.2L^2)Y_t = \epsilon_t$$

```r
polyroot(z=c(1,-1.2,-1.2))
```

```
## [1]  0.540833-0i -1.540833+0i
```

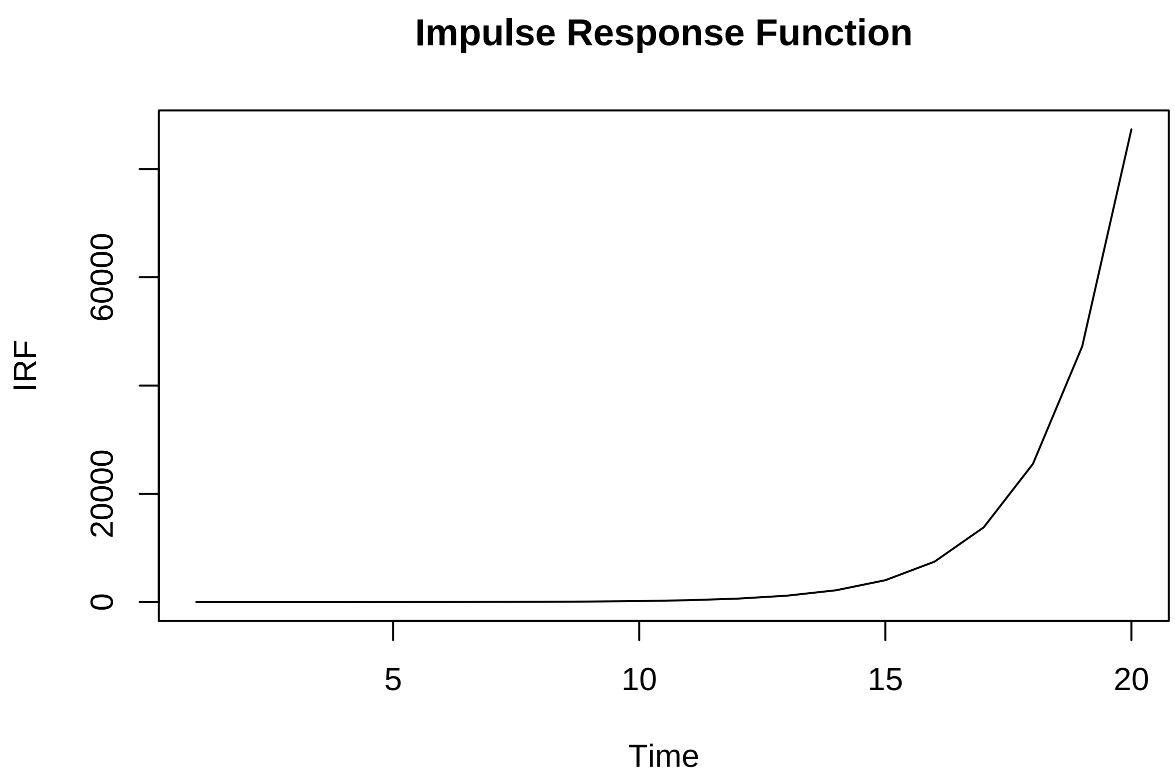
```r
abs(polyroot(z=c(1,-1.2,-1.2)))
```

```
## [1] 0.540833 1.540833
```

其中一根的絕對值 <1，因此非定態。

```r
set.seed(123)
e <- rnorm(n = 200, mean = 0, sd = 1)

y <- numeric(length = 200)
y.1 <- 0
y.2 <- 0
phi1 <- 1.2
phi2 <- 1.2

for (i in 1:length(y)) {
  y[i] <- phi1*y.1 + phi2*y.2 + e[i]
  y.2 <- y.1
  y.1 <- y[i]
}

# 計算 IRF
irf <- numeric(length = 20)
irf[1] <- 1
for (i in 2:length(irf)) {
  if (i == 2) {
```

```
    irf[i] <- phi1
  } else {
    irf[i] <- irf[i-1]*phi1 + irf[i-2]*phi2
  }
}

# 繪製 IRF
plot(irf, type = "l", main = "Impulse Response Function",
     xlab = "Time", ylab = "IRF")
```



(IV)

$$Y_t = -1.2Y_{t-1} + \epsilon_t = -1.2LY_t + \epsilon_t \implies (1 + 1.2L)Y_t = \epsilon_t$$

```
polyroot(z=c(1,1.2))
```

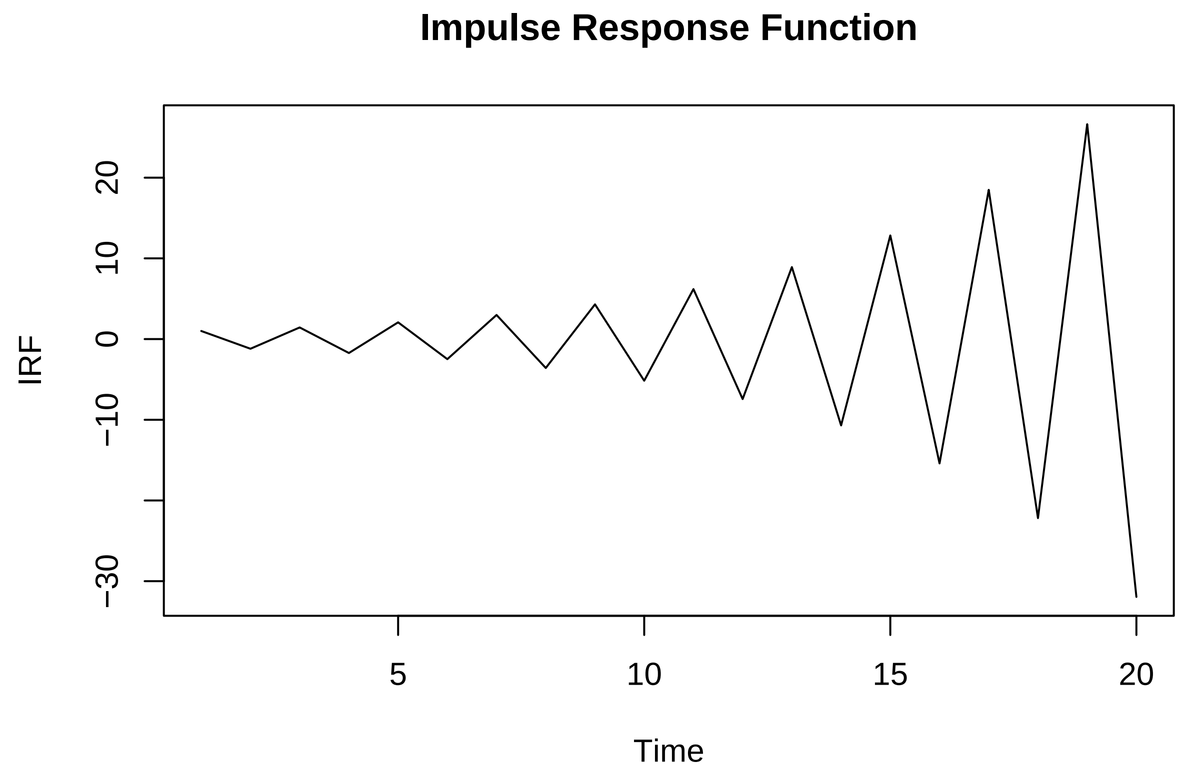
```
## [1] -0.8333333+0i
```

```
abs(polyroot(z=c(1,1.2)))
```

```
## [1] 0.8333333
```

根的絕對值 <1，因此非定態。

```r
set.seed(123)
e <- rnorm(n = 200, mean = 0, sd = 1)

y <- numeric(length = 200)
y.1 <- 0
phi <- -1.2

for (i in 1:length(y)) {
  y[i] <- phi*y.1 + e[i]
  y.1 <- y[i]
}

# 計算 IRF
irf <- numeric(length = 20)
irf[1] <- 1
for (i in 2:length(irf)) {
  irf[i] <- irf[i-1]*phi
}

# 繪製 IRF
plot(irf, type = "l", main = "Impulse Response Function",
     xlab = "Time", ylab = "IRF")
```



(V)

$$
\begin{aligned}
Y_t &= 0.7Y_{t-1} + 0.25Y_{t-2} - 0.175Y_{t-3} + \epsilon_t \\
&= 0.7LY_t + 0.25L^2Y_t - 0.175L^3Y_t + \epsilon_t \\
\implies & (1 - 0.7L - 0.25L^2 + 0.175L^3)Y_t = \epsilon_t
\end{aligned}
$$

```r
polyroot(z=c(1,-0.7,-0.25,0.175))
```

```
## [1]  1.428571-0i -2.000000+0i  2.000000+0i
```

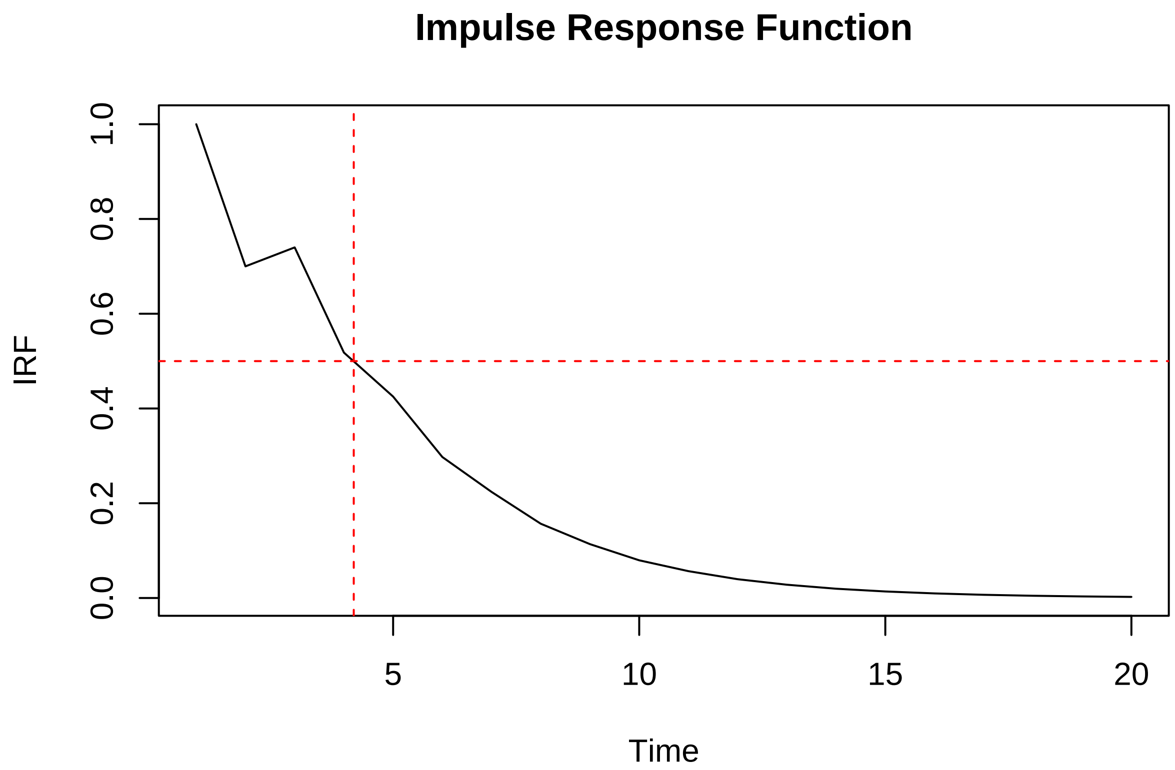
```r
abs(polyroot(z=c(1,-0.7,-0.25,0.175)))
```

```
## [1] 1.428571 2.000000 2.000000
```

根的絕對值 >1，因此定態。

```r
set.seed(123)
e <- rnorm(n = 200, mean = 0, sd = 1)

y <- numeric(length = 200)
```

```r
y.1 <- 0
y.2 <- 0
y.3 <- 0
phi1 <- 0.7
phi2 <- 0.25
phi3 <- -0.175

for (i in 1:length(y)) {
  y[i] <- phi1*y.1 + phi2*y.2+ phi3*y.3 + e[i]
  y.3 <- y.2
  y.2 <- y.1
  y.1 <- y[i]
}

# 計算 IRF
irf <- numeric(length = 20)
irf[1] <- 1
irf[2] <- phi1
for (i in 3:length(irf)) {
  if(i == 3){
    irf[i] <- irf[i-1]*phi1+phi2
  } else {
    irf[i] <- irf[i-1]*phi1 + irf[i-2]*phi2+ irf[i-3]*phi3
  }
}

# 繪製 IRF
plot(irf, type = "l", main = "Impulse Response Function",
     xlab = "Time", ylab = "IRF")
abline(h=0.5, lty=2, col="red")
abline(v=4.2, lty=2, col="red")
```




```
# 計算半衰期
distance=1
i_min <- 1
for (i in 1:length(irf)){
  if (abs(irf[i]-0.5)<distance){
    distance <- abs(irf[i]-0.5)
    i_min <- i
  }
}
i_min
```

```
## [1] 4
```

```
irf[i_min-1]
```

```
## [1] 0.74
```

```
irf[i_min]
```

```
## [1] 0.518
```

```
irf[i_min+1]
```

```
## [1] 0.4251
```

半衰期約發生在 t=4 時。

3.3

AR(1):

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \epsilon_t, \epsilon_t \sim i.i.d.(0, \sigma^2), |\phi_1| < 1, 0 \leq t < \infty$$

initial value: Y_0 be a constant

若此 AR(1) 模型是定態，則 $E(Y_t) = \mu, \forall 0 \leq t < \infty$

$$\begin{aligned} E(Y_t) &= E(\phi_0 + \phi_1 Y_{t-1} + \epsilon_t) \\ &= \phi_0 + \phi_1 E(Y_{t-1}) + E(\epsilon_t) = \\ &= \phi_0 + \phi_1 E(Y_{t-1}) \\ &\implies \mu = \phi_0 + \phi_1 \mu \\ &\implies \mu = \frac{\phi_0}{1-\phi_1} \end{aligned}$$

In general, $E(Y_0) = Y_0 \neq \frac{\phi_0}{1-\phi_1}$,

另外，$Var(Y_0) = \gamma(Y_0, Y_0) = 0$,

$$\begin{aligned} Var(Y_t) &= Var(\phi_0 + \phi_1 Y_{t-1} + \epsilon_t) \\ &= \phi_1^2 Var(Y_{t-1}) + \sigma^2 \\ &\implies \sigma_Y^2 = \phi_1^2 \sigma_Y^2 + \sigma^2 \\ &\implies \sigma_Y^2 = \frac{\sigma^2}{1-\phi_1^2} \neq 0, \forall t > 0 \end{aligned}$$

$Cov(Y_0, Y_{0+h}) = 0, \forall h$

因此，若 AR(1) 有一固定起始值，則非定態。

3.4

若 Y_0 是隨機變數，則要滿足

- $E(Y_0) = \frac{\phi_0}{1-\phi_1}$
- $Var(Y_0) = \frac{\sigma^2}{1-\phi_1^2}$
- $\gamma(h) = Cov(Y_0, Y_{0+h}), \forall h$

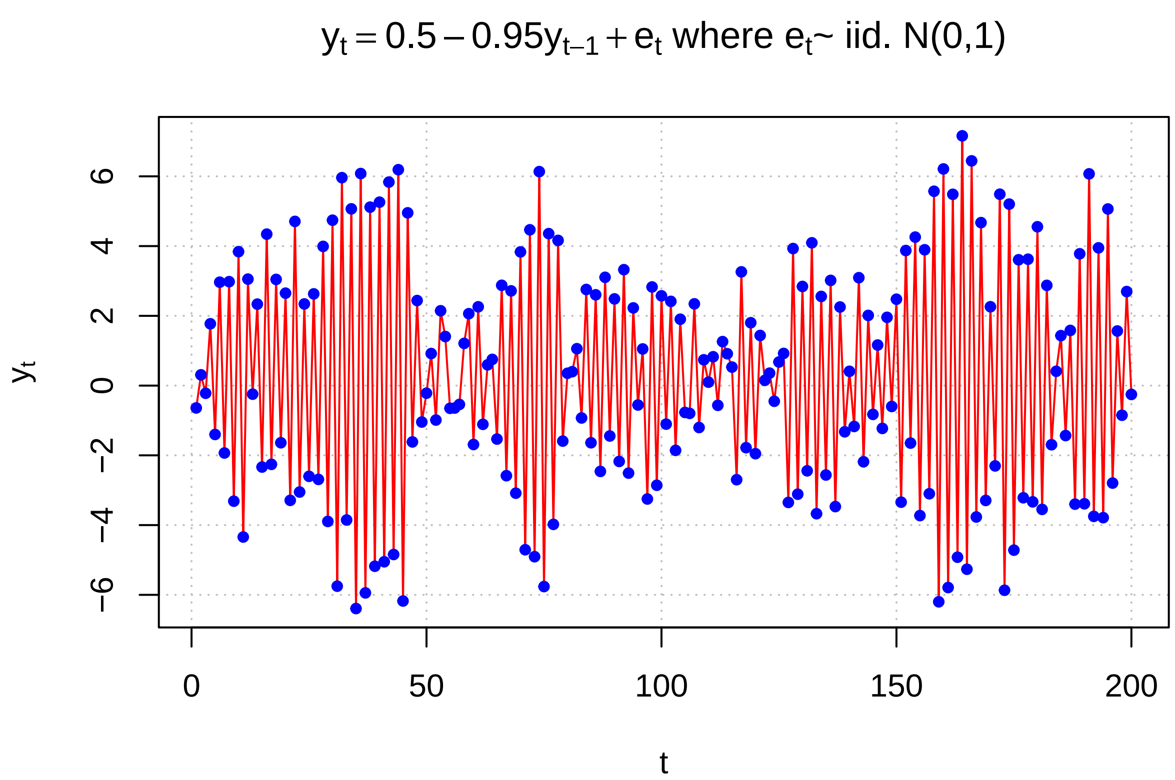
才會至少是弱定態

3.19

(a). AR(1):

$$Y_t = 0.5 - 0.95Y_{t-1} + \epsilon_t, \epsilon_t \sim i.i.d.N(0,1)$$

```
set.seed(7181)
e <- rnorm(n = 200, mean = 0, sd = 1)

y <- numeric(length = 200)
y.1 <- 0
for (i in 1:length(y)) {
  y[i] <- 0.5-0.95*y.1 + e[i]
  y.1 <- y[i]
}
plot(x = y, ylab = expression(y[t]), xlab = "t", type =
  "l", col = c("red"), main =
  expression(paste(y[t] == 0.5-0.95*y[t-1] + e[t], " where ",
  e[t], "~ iid. N(0,1)" )), panel.first=grid(col =
  "gray", lty = "dotted"))
points(x = y, pch = 20, col = "blue")
```



(b). MA(1):

$$Y_t = \epsilon_t - 0.95\epsilon_{t-1}, \epsilon_t \sim i.i.d.N(0,1)$$

```
set.seed(8199)
y <- arima.sim(list(ma=c(-0.95)), n=100, rand.gen=rnorm, sd=1)

plot(x = y, ylab = expression(y[t]), xlab = "t", type =
    "l", col = c("red"), main =
    expression(paste(y[t] == e[t] - 0.95*e[t-1], " where ",
    e[t], " ~ iid. N(0,1)")) , panel.first=grid(col =
    "gray", lty = "dotted"))

points(x = y, pch = 20, col = "blue")
```