Time Series HW10

38/50.

B082040005 高念慈

2023-04-28

1. 7

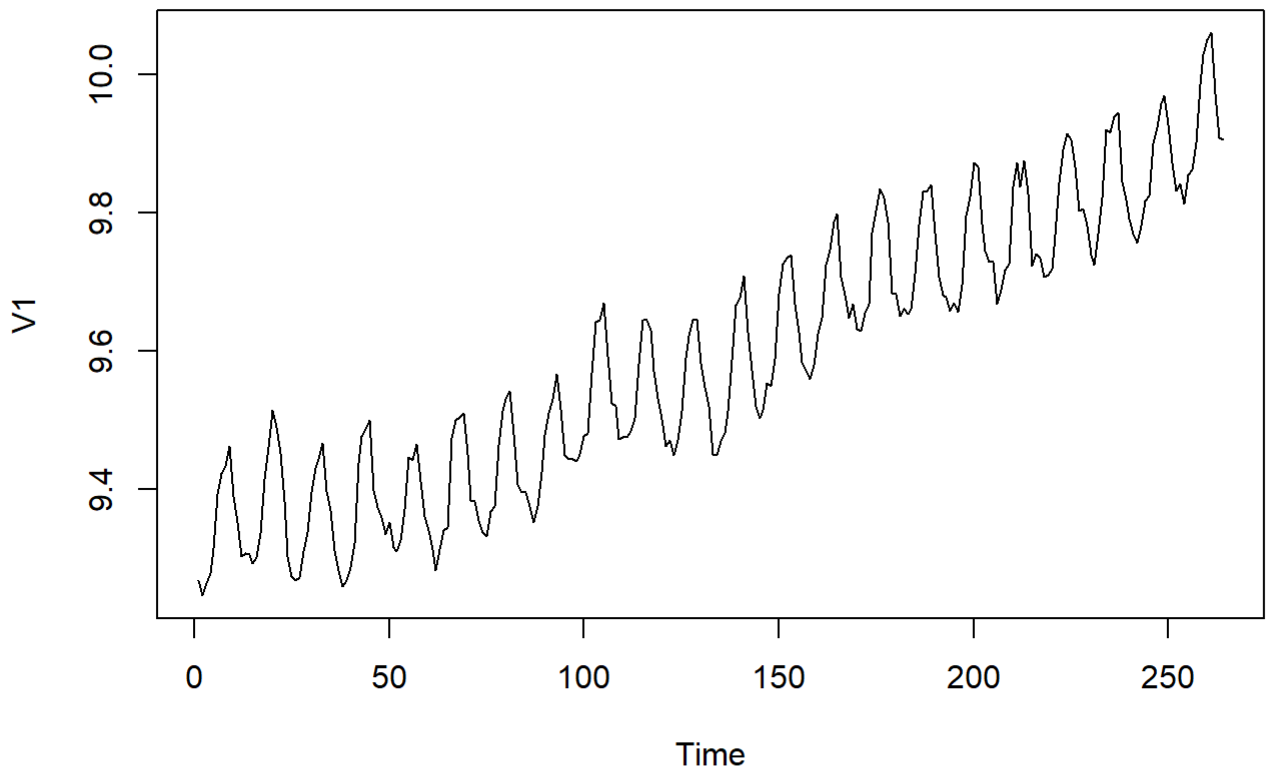
Consider the demand (需求) of electricity of a manufacturing sector (部門)
in the United States. The data are logged (紀錄), denote the demand of a
fixed day of each month, and are in power6.txt.
Build an ARIMA time series model for the series and
use the fitted model to produce 1- to 24-stepahead forecasts

電力需求對數

```
data1 = read.table("https://faculty.chicagobooth.edu/-/media/faculty/ruey-s-tsay/teaching/fts
3/power6.txt", header=F)
head(data1)
```

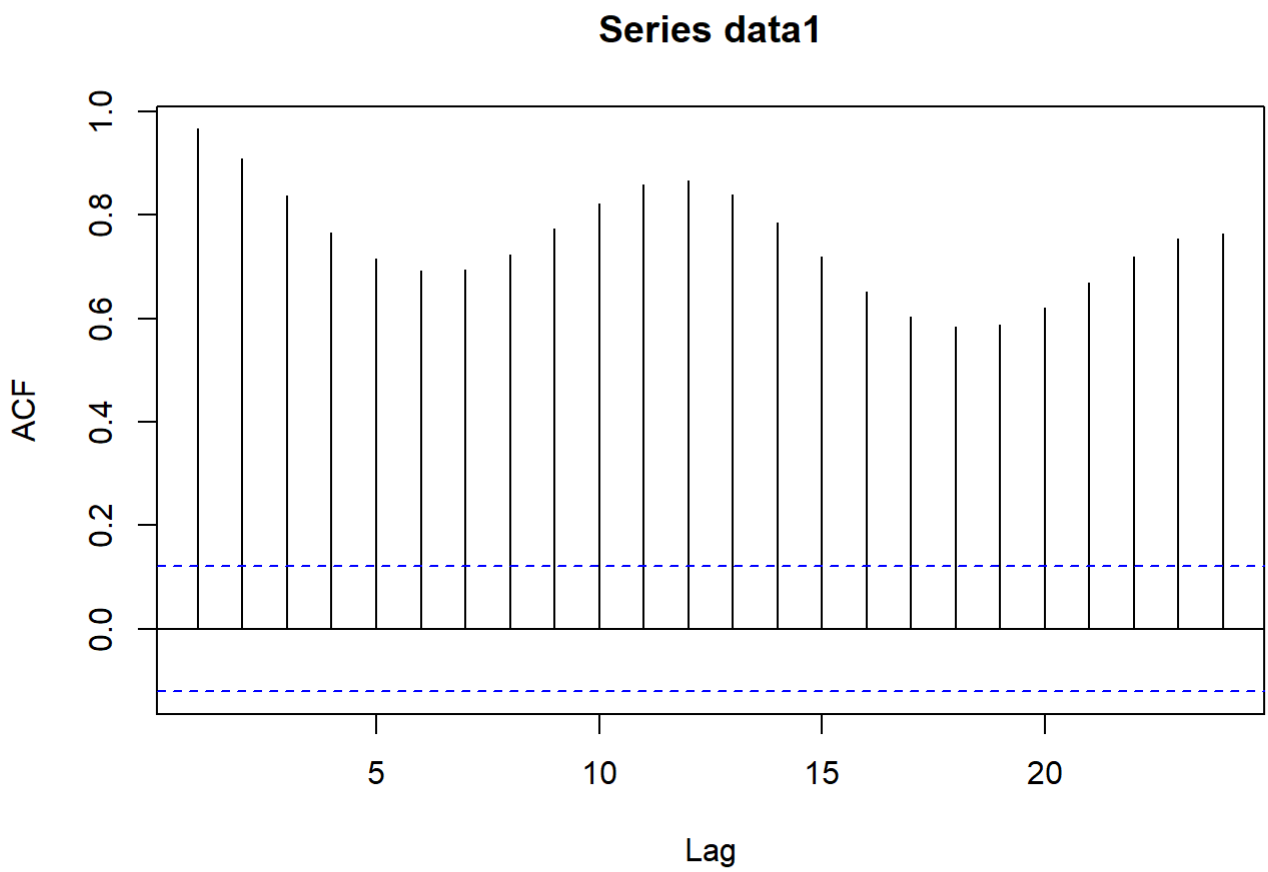
```
##        V1
## 1 9.2691
## 2 9.2465
## 3 9.2639
## 4 9.2784
## 5 9.3147
## 6 9.3932
```

```
# plot(1:nrow(data1),data1$V1)
data1 = ts(data1)
plot(data1)
```



V1
10.0
9.8
9.6
9.4
0
50
100
150
200
250
Time


```
acf(data1)
```



Series data1
ACF
1.0
0.8
0.6
0.4
0.2
0.0
5
10
15
20
Lag

```
adfTest(data1,lags=24,type=c("ct"))
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 24
##   STATISTIC:
##     Dickey-Fuller: -2.7733
##   P VALUE:
##     0.2502
##
## Description:
##  Fri Apr 28 09:51:52 2023 by user: user
```

ARIMA

```
m=ar(diff(data1[,1] ), method="mle")
m$order
```

```
## [1] 12
```

```
estmodel1 = arima(data1,order=c(12,1,0))
estmodel1
```

```
##
## Call:
## arima(x = data1, order = c(12, 1, 0))
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5     ar6      ar7      ar8
##       -0.1417  -0.0923  -0.0301  -0.3267  -0.2051  -0.108  -0.1159  -0.3136
## s.e.   0.0566   0.0571   0.0569   0.0565   0.0565   0.057   0.0574   0.0567
##           ar9     ar10    ar11    ar12
##       -0.1719  -0.0901  0.0801  0.3996
## s.e.   0.0564   0.0575  0.0574  0.0575
##
## sigma^2 estimated as 0.0005356:  log likelihood = 613.36,  aic = -1202.71
```

```
rbind(estmodel1$coef-2*sqrt(diag(estmodel1$var.coef)),estmodel1$coef+2*sqrt(diag(estmodel1$va
r.coef)))
```

```
##               ar1         ar2         ar3        ar4         ar5          ar6
## [1,] -0.25501921 -0.20649549 -0.14391956 -0.4396558 -0.31820808 -0.221947849
## [2,] -0.02846179  0.02195138  0.08374351 -0.2137286 -0.09207972  0.005858624
##                ar7         ar8         ar9        ar10        ar11      ar12
## [1,] -0.230787175 -0.4270238 -0.28470123 -0.20509951 -0.03478253 0.2844797
## [2,] -0.001071382 -0.2001493 -0.05906416  0.02496528  0.19493025 0.5146280
```

- ar2
- ar3
- ar6
- ar10
- ar11

```
estmodel2 = arima(data1,
                  order=c(12,1,0),
                  fixed=c(NA,0,0,NA,NA,0,NA,NA,NA,0,0,NA),
                  transform.pars = FALSE)
estmodel2
```

```
## 
## Call:
## arima(x = data1, order = c(12, 1, 0), transform.pars = FALSE, fixed = c(NA, 
##     0, 0, NA, NA, 0, NA, NA, NA, 0, 0, NA))
## 
## Coefficients:
##           ar1  ar2  ar3      ar4      ar5  ar6      ar7      ar8      ar9  ar10
##       -0.1093    0    0  -0.3461  -0.2054    0  -0.1295  -0.2889  -0.1411     0
## s.e.   0.0542    0    0   0.0532   0.0529    0   0.0414   0.0535   0.0530     0
##       ar11    ar12
##          0  0.4271
## s.e.     0  0.0562
## 
## sigma^2 estimated as 0.0005578:  log likelihood = 607.97,  aic = -1201.95
```

```
Box.test(estmodel2$residuals, lag=24, type="Ljung", fitdf=12-5)
```

```
## 
##  Box-Ljung test
## 
## data:  estmodel2$residuals
## X-squared = 46.671, df = 17, p-value = 0.0001367
```

```
estmodel3 = arima(data1,
                  order=c(12,1,0),
                  seasonal = list(order=c(1,1,1), period=12),
                  method="ML")
estmodel3
```

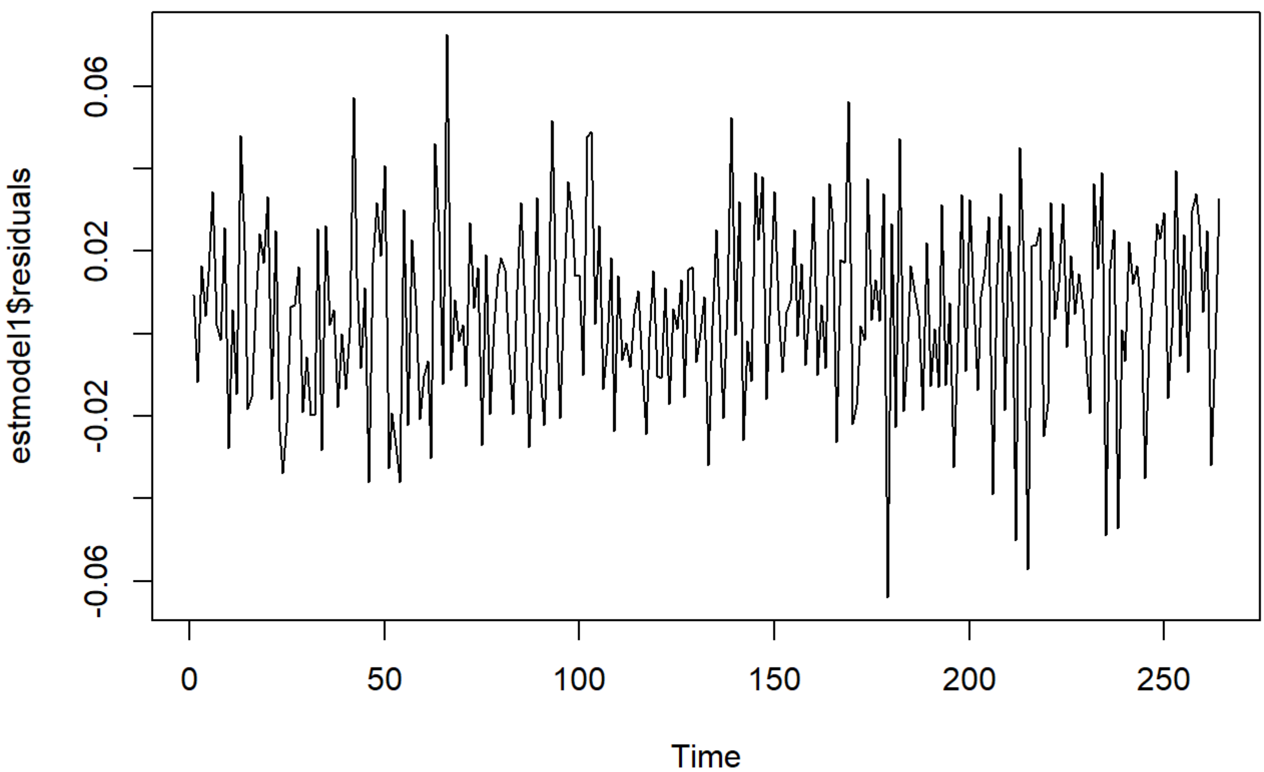
```
##
## Call:
## arima(x = data1, order = c(12, 1, 0), seasonal = list(order = c(1, 1, 1), period = 12),
##     method = "ML")
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5      ar6     ar7     ar8
##       -0.4435  -0.2752  -0.1523  -0.2016  -0.0692  -0.0174  0.0466  0.0105
## s.e.   0.0634   0.0726   0.0719   0.0723   0.0727   0.0745  0.0736  0.0746
##          ar9     ar10    ar11     ar12     sar1     sma1
##       0.0506  0.0384  0.0060  -0.0300  -0.0222  -0.9300
## s.e.  0.0729  0.0721  0.0693   0.1356   0.1609   0.0747
##
## sigma^2 estimated as 0.0003295:  log likelihood = 637.54,  aic = -1247.07
```

```
Box.test(estmodel3$residuals, lag=24, type="Ljung")
```

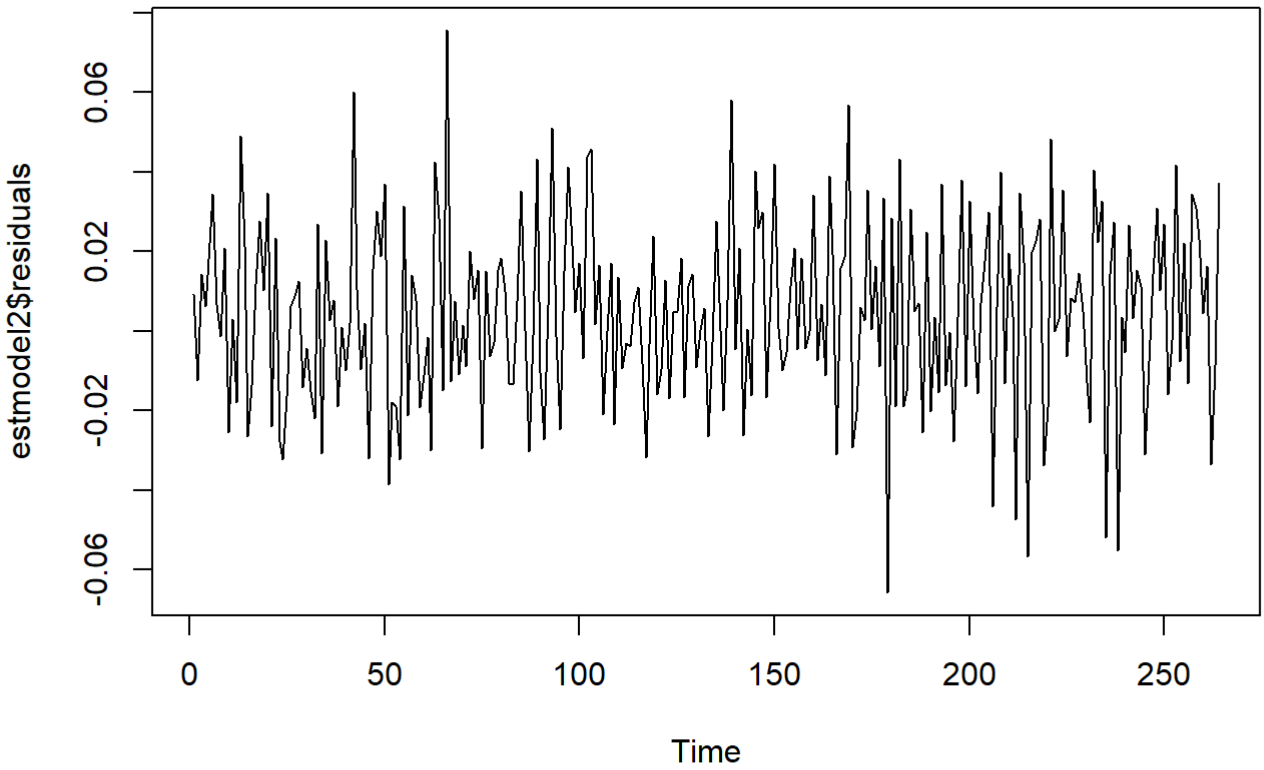
fitdf 沒設 -3

```
##
##  Box-Ljung test
##
## data:  estmodel3$residuals
## X-squared = 9.4906, df = 24, p-value = 0.9963
```

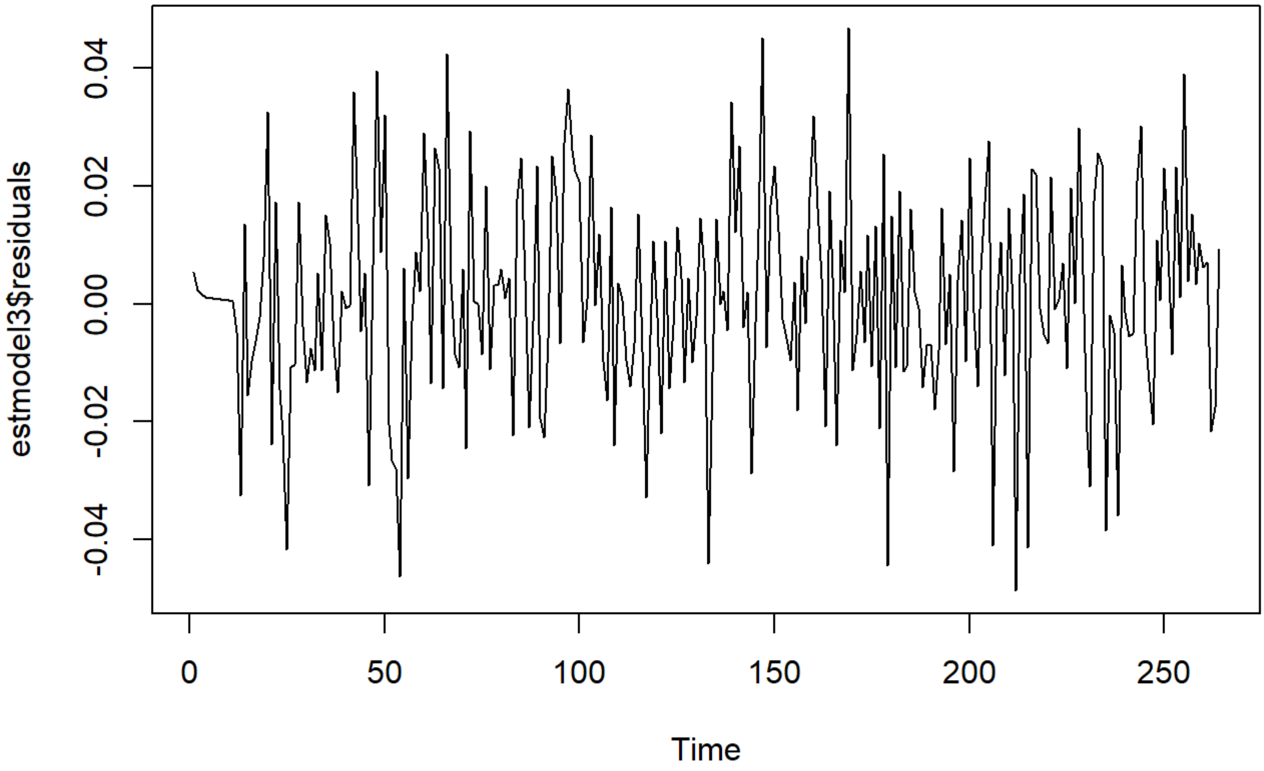
```
# par(mfrow=c(3,1),mar=c(4,4,4,1)) # 邊:下左上右
plot(estmodel1$residuals)
```



```
plot(estmodel2$residuals)
```




```
plot(estmodel3$residuals)
```




```
knitr::include_graphics("C:/Users/user/Desktop/time_series/seasonal_diff.jpg")
```

在 R 中，使用 `forecast::forecast()` 函數進行 ARIMA 模型的擬合，可以通過設置 `lambda=0` 來抑制季節性差分，從而獲得原始時間序列的預測值。如果要單獨獲取季節性差分後的序列，可以使用 `forecast::residuals()` 函數獲取 ARIMA 模型的殘差序列，殘差序列就是原始序列減去 ARIMA 模型的預測值所得到的序列。

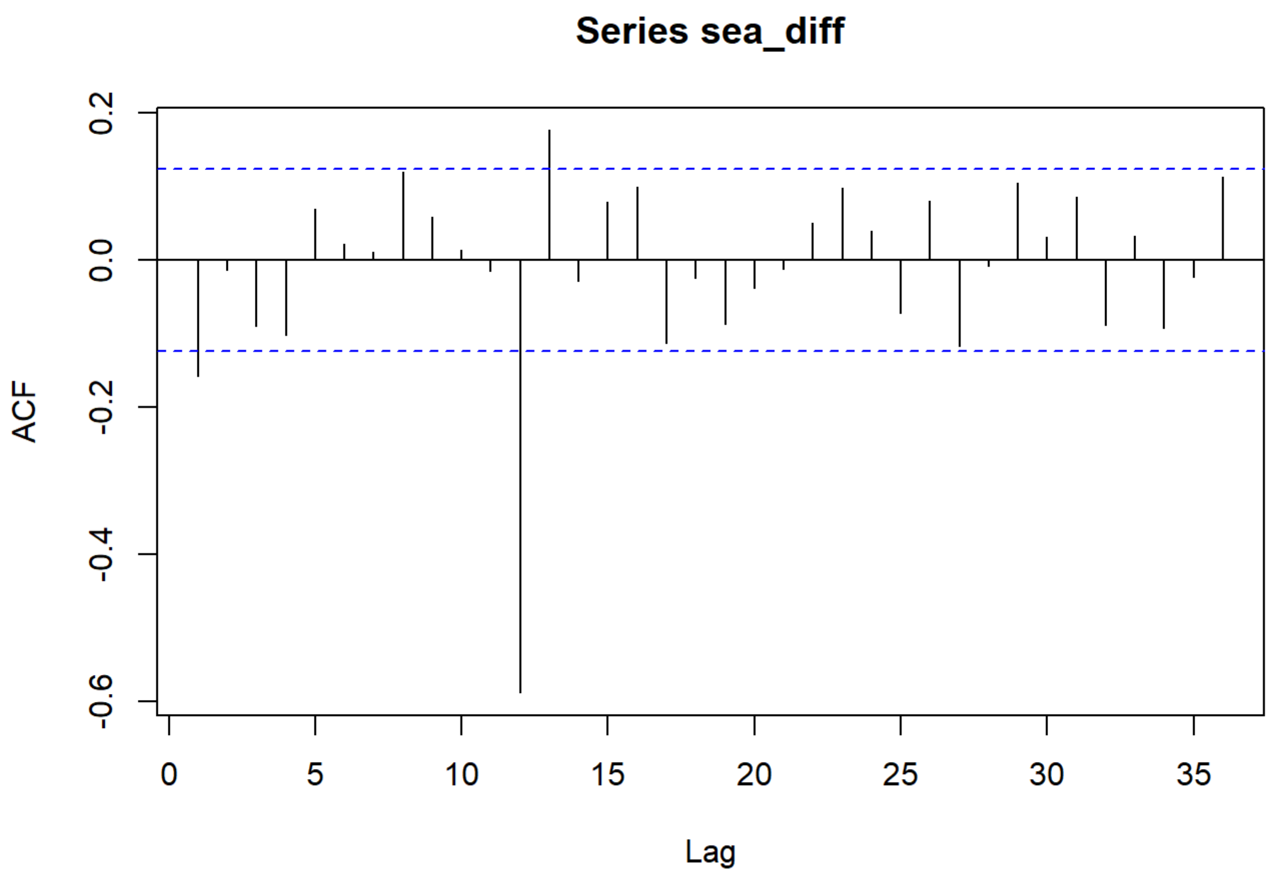
假設你已經擬合了一個 ARIMA(1,1,1)(0,1,1)12 模型，並且命名為 `model`，可以使用以下代碼獲取季節性差分後的序列：

R　Copy code

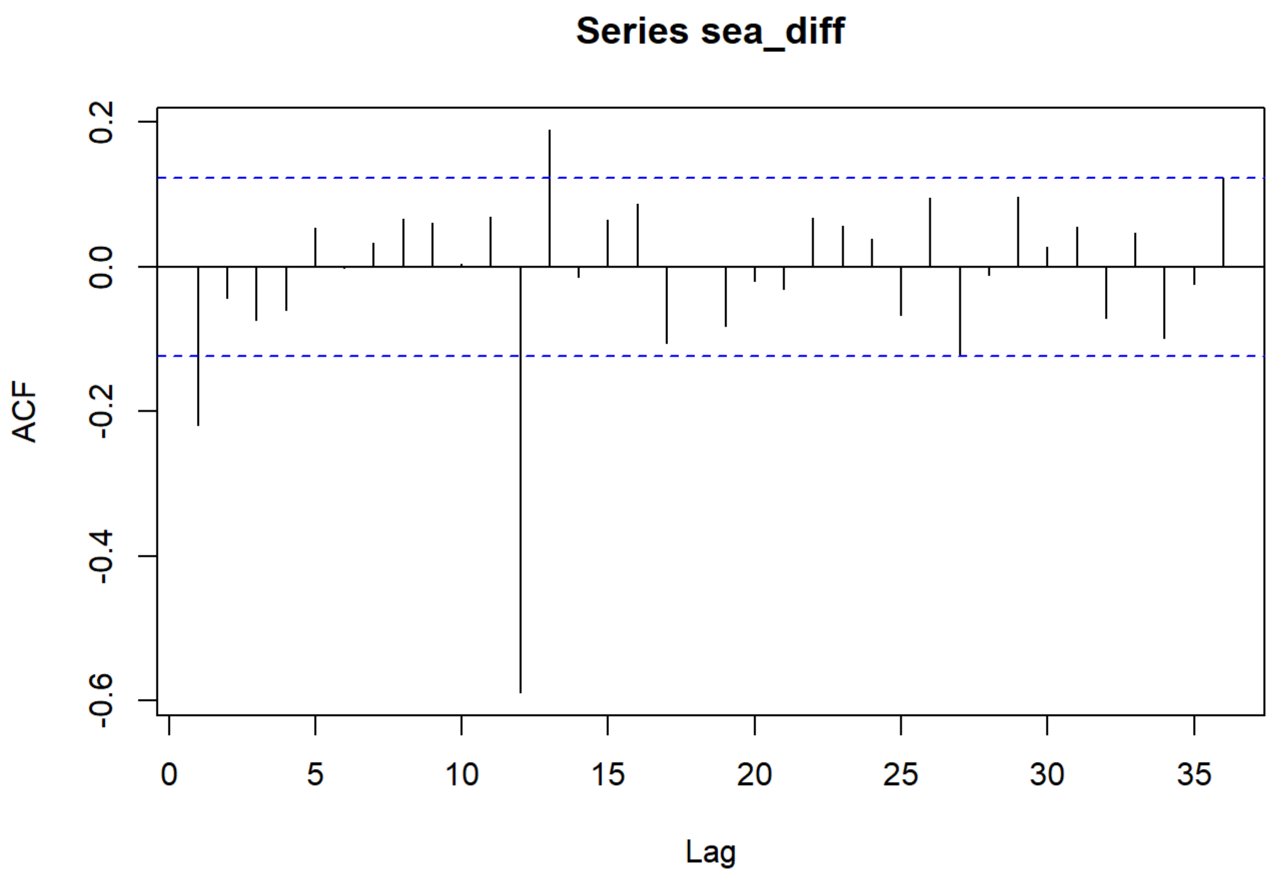
```r
# 獲取 ARIMA 模型的殘差序列
residuals <- residuals(model)

# 獲取季節性差分後的序列
seasonal_diff <- diff(residuals, lag = 12)
```

其中，`diff()` 函數的 `lag` 參數設置為 12，因為季節周期為 12。最終得到的 `seasonal_diff` 序列即為季節性差分後的序列。

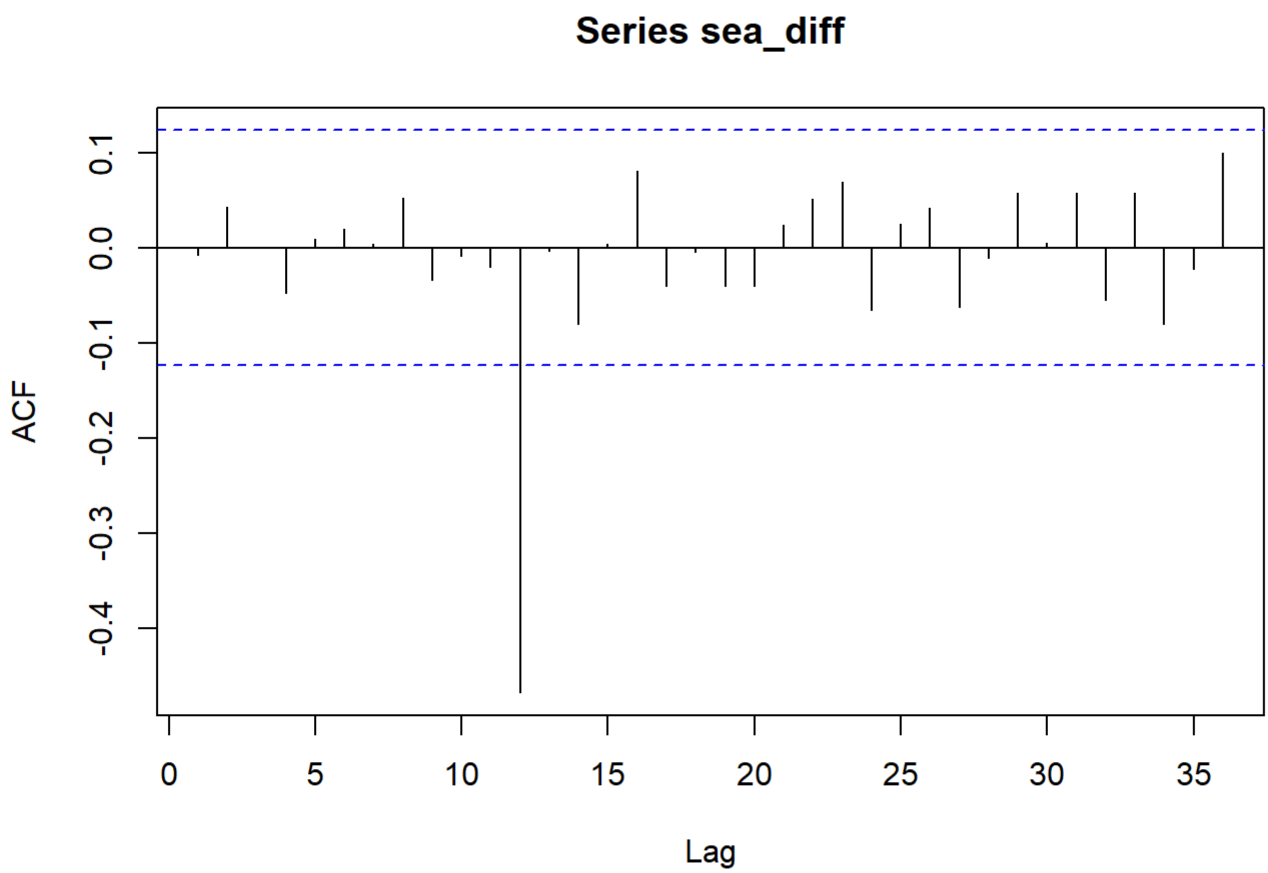
```
residual = estmodel1$residuals
sea_diff = diff(residual,lag=12)
acf(sea_diff, lag.max = 36)
```




```
residual = estmodel2$residuals
sea_diff = diff(residual,lag=12)
acf(sea_diff, lag.max = 36)
```




```
residual = estmodel3$residuals
sea_diff = diff(residual,lag=12)
acf(sea_diff, lag.max = 36)
```




```
forecast = predict(estmodel3,n.ahead=24)
forecast$pred
```

```
## Time Series:
## Start = 265
## End = 288
## Frequency = 1
##  [1]  9.885234  9.871566  9.878023  9.895135  9.923770 10.008822 10.047577
##  [8] 10.063985 10.074243 10.009724  9.953802  9.931109  9.914588  9.897898
## [15]  9.905695  9.925135  9.954578 10.039574 10.078259 10.095647 10.104600
## [22] 10.038657  9.982967  9.961430
```

5.

2.

-2 weekly → monthly

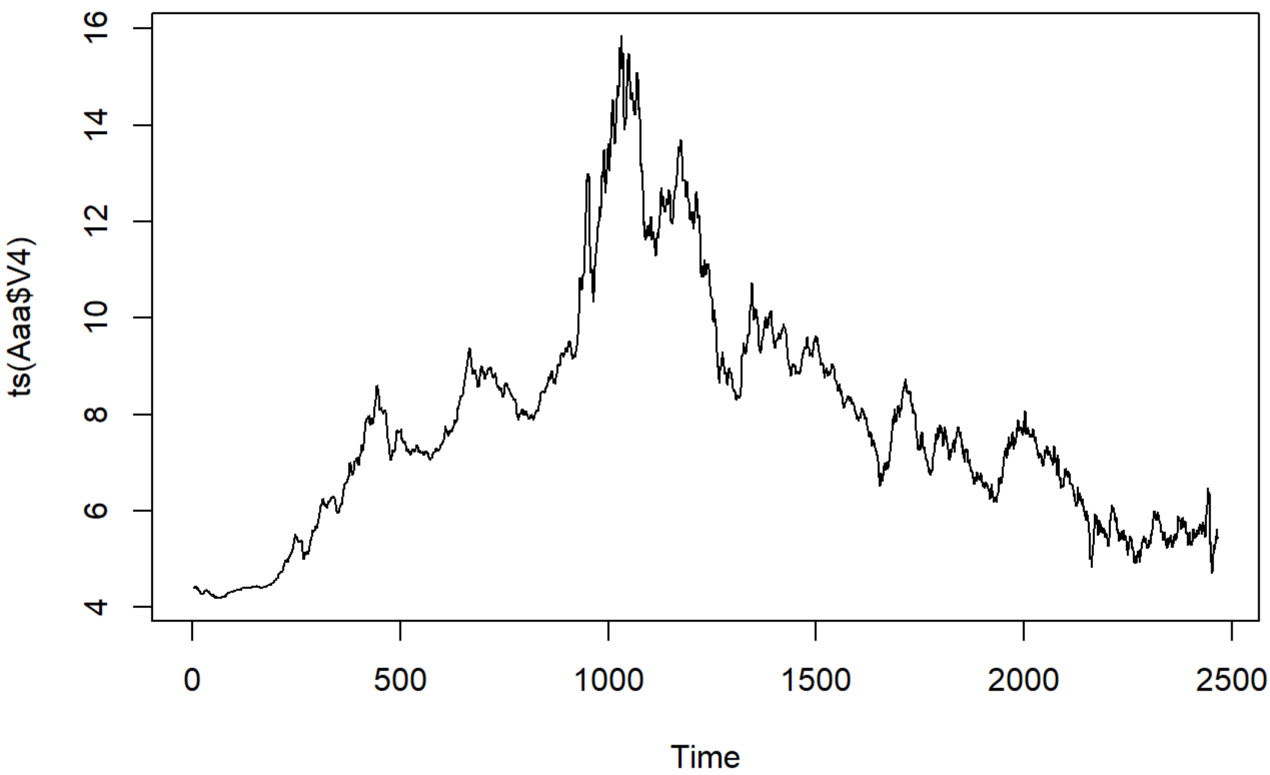
Consider the monthly Aaa bond yields of the prior problem(Assignment 8 (4)).
Build an ARIMA time series model for the series.

Aaa

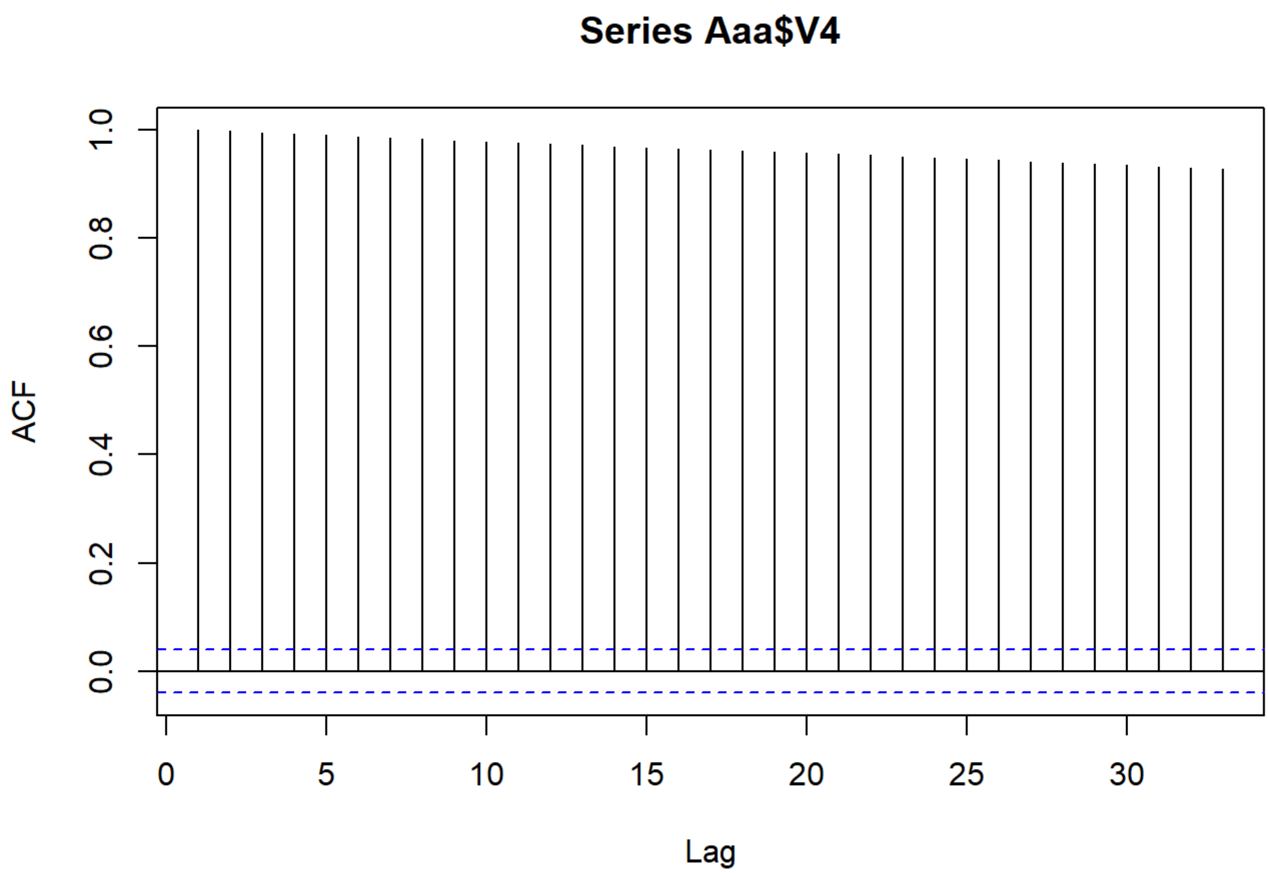
```
Aaa = read.table("C:/Users/user/Desktop/time_series/HW/w-Aaa.txt",header = F)
head(Aaa)
```

```
##       V1 V2 V3   V4
## 1 1962  1  5 4.43
## 2 1962  1 12 4.42
## 3 1962  1 19 4.42
## 4 1962  1 26 4.41
## 5 1962  2  2 4.42
## 6 1962  2  9 4.42
```

```
plot(ts(Aaa$V4))
```




```
acf(Aaa$V4)
```




```
adfTest(Aaa$V4,lags=24,type=c("c"))
```

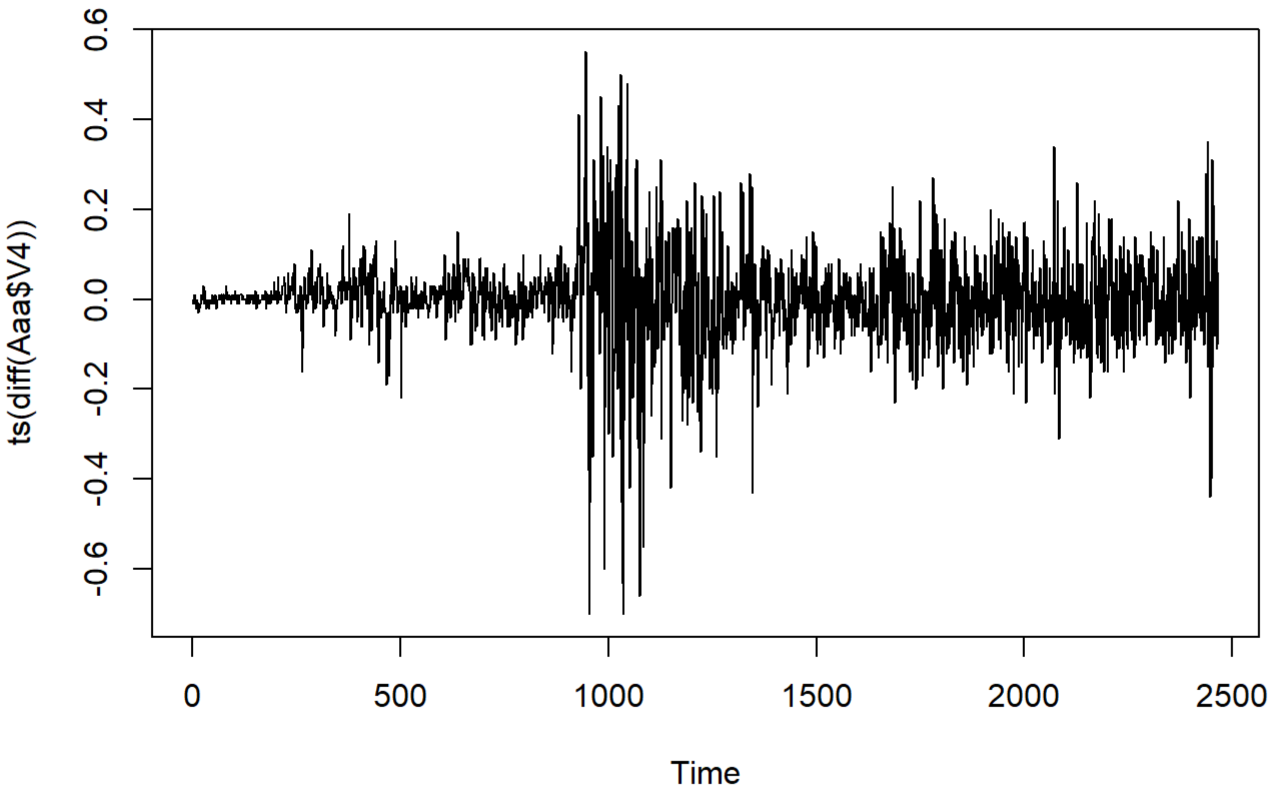
```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 24
##   STATISTIC:
##     Dickey-Fuller: -1.7138
##   P VALUE:
##     0.4201
##
## Description:
##  Fri Apr 28 09:51:57 2023 by user: user
```

```
adfTest(diff(Aaa$V4),lags=24,type=c("c"))
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 24
##   STATISTIC:
##     Dickey-Fuller: -8.9041
##   P VALUE:
##     0.01
##
## Description:
##  Fri Apr 28 09:51:57 2023 by user: user
```

ARIMA

```
plot(ts(diff(Aaa$V4)))
```




```
a = eacf(diff(Aaa$V4))
```

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x x x x x o x o x x x  o  o  x
## 1 x x o o x o x o o o o  o  o  x
## 2 x x o o x o x o o o o  o  o  x
## 3 x x o x x o x o o o o  o  o  x
## 4 x x x x x o x x x o o  o  o  o
## 5 x x x x x o x o x o o  o  o  o
## 6 x x x x x x o o x o o  o  o  o
## 7 x x x o x x o x x o o  o  o  o
```

```
a
```

```
## $eacf
##             [,1]        [,2]         [,3]         [,4]        [,5]
## [1,]  0.36830361  0.11495649  0.107969164  0.074318904  0.05352157
## [2,]  0.06514535 -0.19406276  0.029364973 -0.003343535  0.05780652
## [3,]  0.26271641 -0.22364181  0.009953875  0.001000876  0.05128305
## [4,] -0.11719638 -0.21185782  0.004190754  0.069282223  0.04843773
## [5,] -0.38560183 -0.06161516  0.378964231 -0.074014333  0.14129937
## [6,]  0.33806950  0.22966920  0.302995489  0.181773137  0.14133976
## [7,] -0.49586146  0.35602644 -0.342035294  0.291594860 -0.15825085
## [8,]  0.10527436 -0.27052100 -0.393258313  0.028367896  0.12733931
##               [,6]         [,7]         [,8]        [,9]        [,10]
## [1,] -0.0141862323 -0.058519989 -0.029718673 -0.06437679 -0.063559212
## [2,] -0.0158037846 -0.058003278  0.023896947 -0.02837114 -0.013168229
## [3,]  0.0092868769 -0.060140991  0.010479294 -0.03758126  0.008771952
## [4,]  0.0070697964 -0.061991866 -0.019428127 -0.02621373 -0.014345186
## [5,] -0.0007458337 -0.053556778  0.042288181 -0.04908788 -0.018185854
## [6,] -0.0011489045 -0.041097235  0.015799865 -0.05917183  0.019356304
## [7,]  0.1866005542 -0.009716541 -0.007489158 -0.04094584 -0.008919339
## [8,]  0.1681724949 -0.012189082 -0.076812641 -0.04139245 -0.008130541
##            [,11]        [,12]         [,13]        [,14]
## [1,] -0.04946062 -0.017771972 -0.0326487001 -0.060979363
## [2,] -0.03096522  0.024039563 -0.0006896259 -0.061034951
## [3,] -0.01829121  0.024597802 -0.0017926450 -0.049420554
## [4,] -0.02352728  0.006978665 -0.0023662409 -0.043543251
## [5,] -0.02432684  0.007322114 -0.0005561746 -0.035160293
## [6,] -0.02567578  0.001340920 -0.0067586649 -0.033349142
## [7,]  0.02039475  0.025633123 -0.0186565182  0.002741181
## [8,]  0.01530369  0.038975242 -0.0108496407 -0.009569732
## 
## $ar.max
## [1] 8
## 
## $ma.ma
## [1] 14
## 
## $symbol
##   0   1   2   3   4   5   6   7   8   9   10  11  12  13 
## 0 "x" "x" "x" "x" "x" "o" "x" "o" "x" "x" "x" "o" "o" "x"
## 1 "x" "x" "o" "o" "x" "o" "x" "o" "o" "o" "o" "o" "o" "x"
## 2 "x" "x" "o" "o" "x" "o" "x" "o" "o" "o" "o" "o" "o" "x"
## 3 "x" "x" "o" "x" "x" "o" "x" "o" "o" "o" "o" "o" "o" "x"
## 4 "x" "x" "x" "x" "x" "o" "x" "x" "x" "o" "o" "o" "o" "o"
## 5 "x" "x" "x" "x" "x" "o" "x" "o" "x" "o" "o" "o" "o" "o"
## 6 "x" "x" "x" "x" "x" "x" "o" "o" "x" "o" "o" "o" "o" "o"
## 7 "x" "x" "x" "o" "x" "x" "o" "x" "x" "o" "o" "o" "o" "o"
```

取 AR(1) MA(7)

```
model = arima(Aaa$V4,order=c(1,1,7))
model
```

```
##
## Call:
## arima(x = Aaa$V4, order = c(1, 1, 7))
##
## Coefficients:
##           ar1     ma1     ma2     ma3     ma4     ma5     ma6      ma7
##       -0.2431  0.6225  0.1786  0.1175  0.0927  0.0907  0.0431  -0.0408
## s.e.   0.4215  0.4215  0.1640  0.0468  0.0513  0.0407  0.0432   0.0338
##
## sigma^2 estimated as 0.007955:  log likelihood = 2461.1,  aic = -4906.2
```

```
Box.test(model$residuals, lag=12, type="Ljung")
```

```
##
##  Box-Ljung test
##
## data:  model$residuals
## X-squared = 8.6257, df = 12, p-value = 0.7345
```

```
rbind(model$coef-2*sqrt(diag(model$var.coef)),model$coef+2*sqrt(diag(model$var.coef)))
```

```
##             ar1        ma1        ma2       ma3          ma4         ma5
## [1,] -1.0860705 -0.2205847 -0.1494678 0.0239469 -0.009779349 0.009390211
## [2,]  0.5998079  1.4655524  0.5066483 0.2110654  0.195228919 0.171999108
##              ma6         ma7
## [1,] -0.04325282 -0.10846481
## [2,]  0.12947292  0.02686175
```

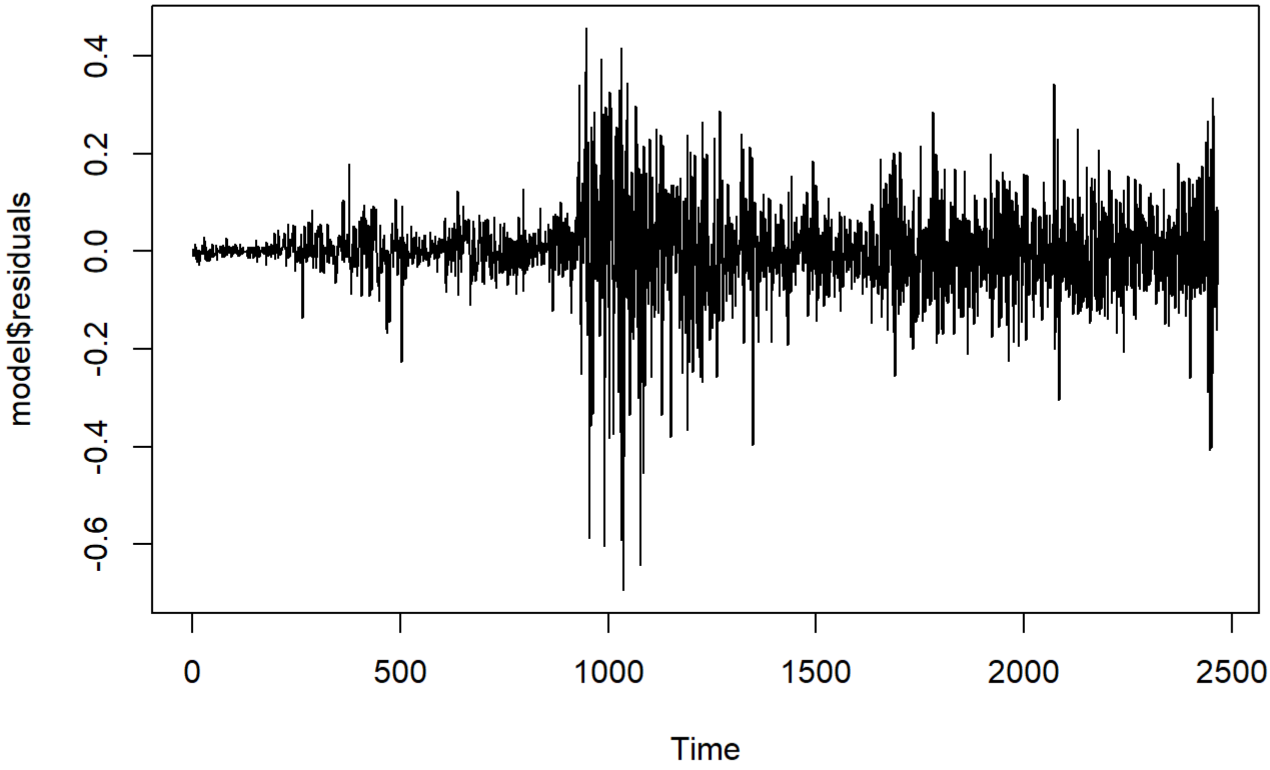
```
model2 = arima(Aaa$V4,order=c(1,1,7),
               fixed=c(0,0,0,NA,0,NA,0,0),
               transform.pars = FALSE)
model2
```

```
##
## Call:
## arima(x = Aaa$V4, order = c(1, 1, 7), transform.pars = FALSE, fixed = c(0, 0,
##     0, NA, 0, NA, 0, 0))
##
## Coefficients:
##       ar1  ma1  ma2     ma3  ma4     ma5  ma6  ma7
##         0    0    0  0.1101    0  0.0566    0    0
## s.e.    0    0    0  0.0202    0  0.0209    0    0
##
## sigma^2 estimated as 0.009196:  log likelihood = 2282.38,  aic = -4560.76
```

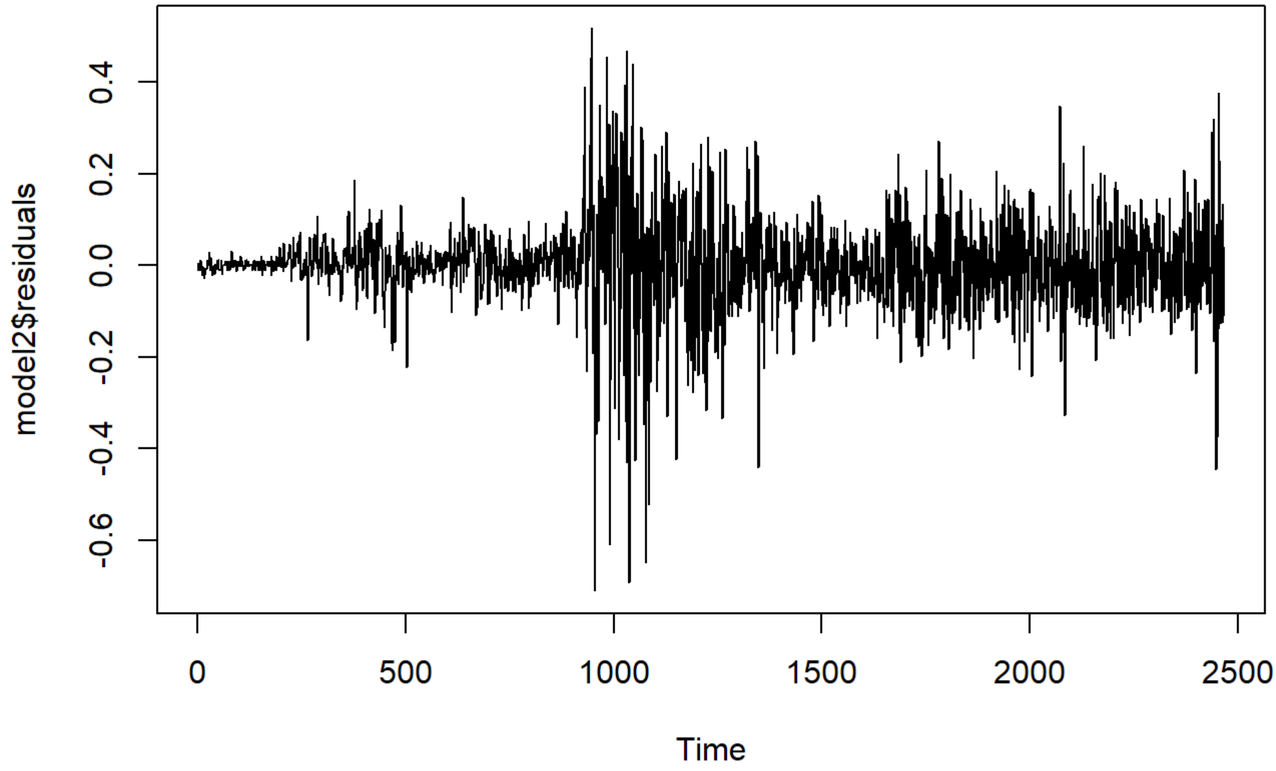
```
Box.test(model2$residuals, lag=12, type="Ljung",fitdf=8-6)
```

```
##
##  Box-Ljung test
##
## data:  model2$residuals
## X-squared = 358.16, df = 10, p-value < 2.2e-16
```

```
plot(model$residuals)
```




```
plot(model2$residuals)
```



10.

- 還是取 ARIMA(1,1,7) model

3.

The quarterly gross domestic product implicit price deflator is often (季度國內生產總值隱含物價平減指數常被用作衡量通貨膨脹的指標。) used as a measure of inflation. The file q-gdpdef.txt contains the data for the United States from the first quarter of 1947 to the last quarter of 2008. Data format is year, month, day, and deflator. The data are seasonally adjusted and equal to 100 for year 2000. Build an ARIMA model for the series and check the validity of the fitted model. Use the fitted model to predict the inflation for each quarter of 2009. The data are obtained from the Federal Reserve Bank of St Louis.

季度GDP隱含物價平減指數

```
gdpdata = read.table("https://faculty.chicagobooth.edu/-/media/faculty/ruey-s-tsay/teaching/fts3/q-gdpdef.txt", header=T)
head(gdpdata)
```

```
##   year mom day gdpdef
## 1 1947   1   1 15.105
## 2 1947   4   1 15.329
## 3 1947   7   1 15.597
## 4 1947  10   1 15.989
## 5 1948   1   1 16.111
## 6 1948   4   1 16.254
```

```
adfTest(gdpdata$gdpdef,lags=12,type=c("c"))
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 12
##   STATISTIC:
##     Dickey-Fuller: 0.38
##   P VALUE:
##     0.9805
##
## Description:
##  Fri Apr 28 09:51:59 2023 by user: user
```

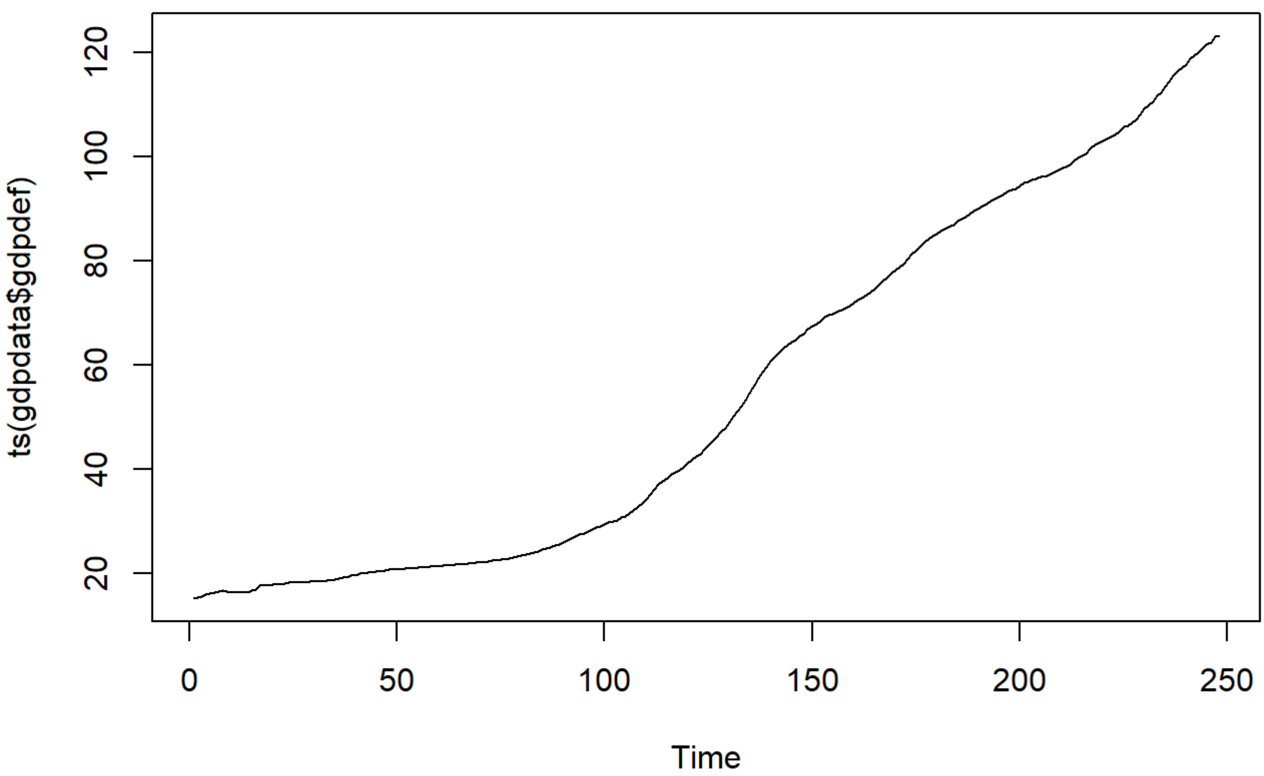
```
adfTest(diff(gdpdata$gdpdef),lags=12,type=c("c"))
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 12
##   STATISTIC:
##     Dickey-Fuller: -1.923
##   P VALUE:
##     0.3408
##
## Description:
##  Fri Apr 28 09:51:59 2023 by user: user
```

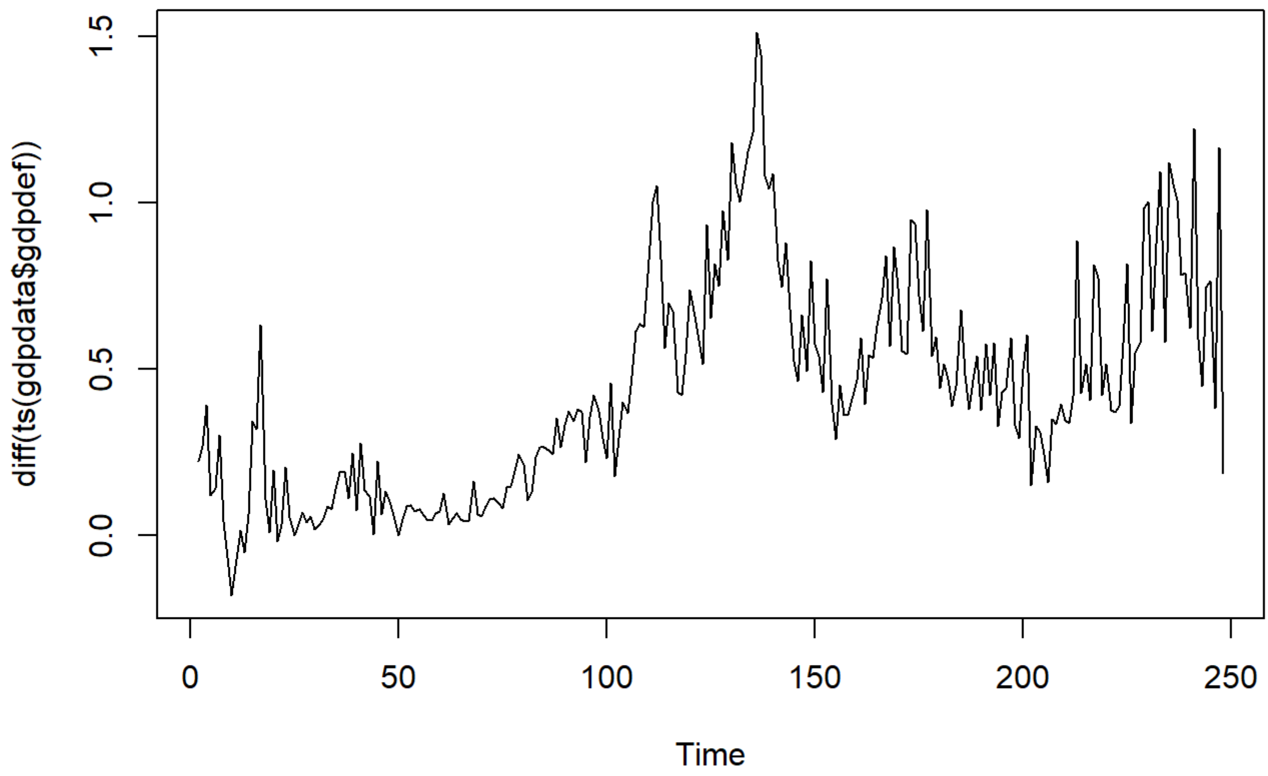
```
adfTest(diff(diff(gdpdata$gdpdef)),lags=12,type=c("c"))
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 12
##   STATISTIC:
##     Dickey-Fuller: -4.8349
##   P VALUE:
##     0.01
##
## Description:
##  Fri Apr 28 09:51:59 2023 by user: user
```

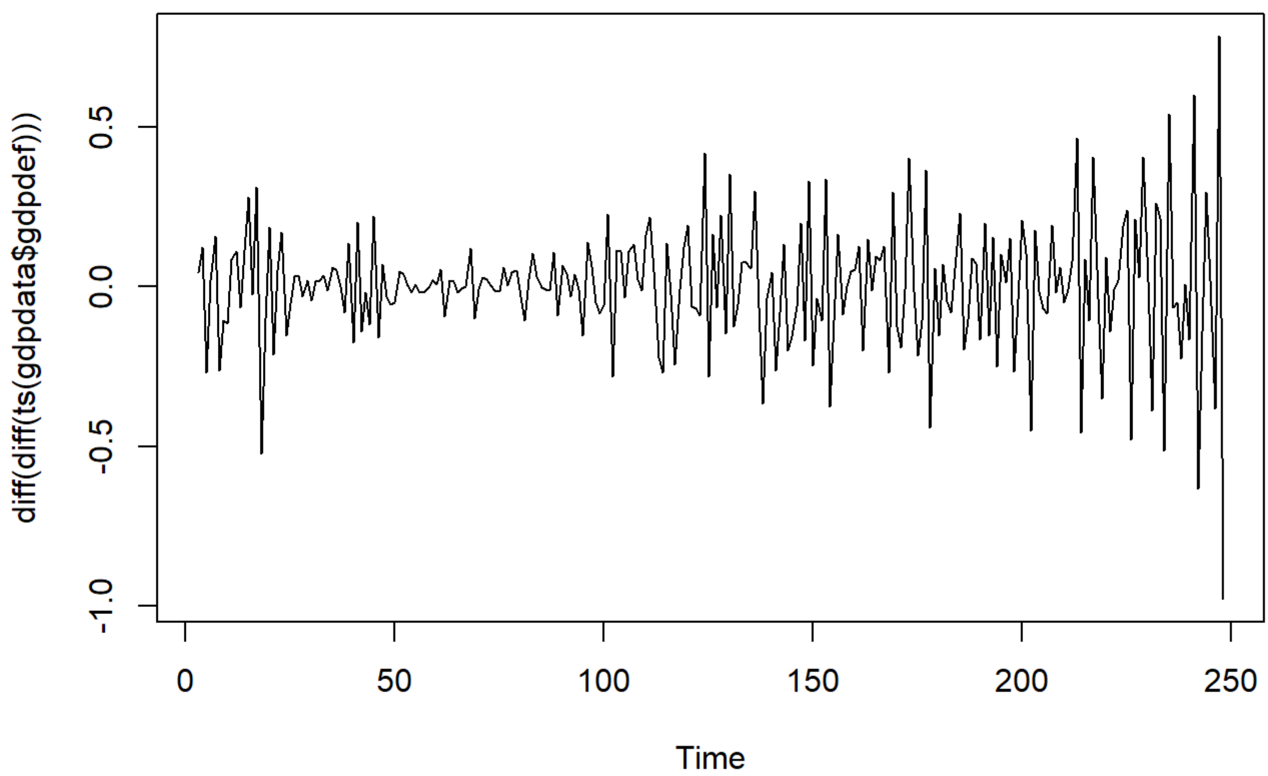
```
plot(ts(gdpdata$gdpdef))
```




```
plot(diff(ts(gdpdata$gdpdef)))
```




```
plot(diff(diff(ts(gdpdata$gdpdef))))
```



構建 ARIMA 系列的模型並檢查擬合模型的有效性

```
eacf(diff(diff(gdpdata$gdpdef)),ar.max=15,ma.max=15)
```

```
## AR/MA
##    0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
## 0  x o o x x x x x o o o  x  x  o  o  o
## 1  x x o x o o o o o o o  x  o  x  o  o
## 2  x o o x o o o o o o o  x  o  x  o  o
## 3  x o x o o o o o o o o  o  o  o  o  o
## 4  x o x o x o o o o o o  o  o  o  o  o
## 5  x x x o x o o o o x o  o  o  o  o  o
## 6  x x x o o o o o o o o  o  o  x  o  o
## 7  x x x o x o o x x o o  o  o  x  o  o
## 8  x x o o x x o x o o o  o  o  o  o  o
## 9  x x o o x x o o x o o  o  o  o  o  o
## 10 x x x x x x x x x o x  o  o  o  o  o
## 11 x x x x x x o o o x x  o  o  o  o  o
## 12 o x o x x o x o o o x  x  x  o  o  o
## 13 x o x o x o o o o o x  x  o  o  o  o
## 14 x o x o x x o o o o o  x  o  o  o  o
## 15 x x x x o x x o o o o  o  o  o  o  o
```

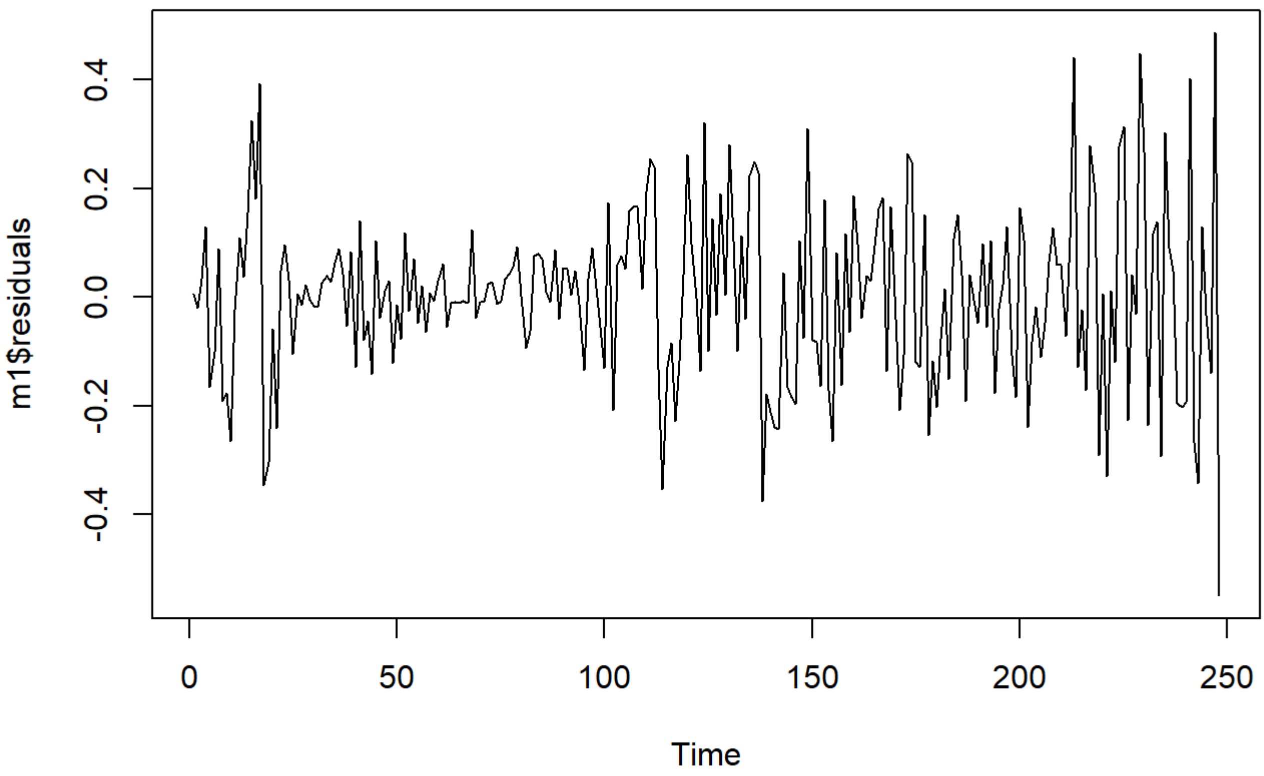
```
m1 = arima(gdpdata$gdpdef, order=c(2,2,4))
m1
```

```
##
## Call:
## arima(x = gdpdata$gdpdef, order = c(2, 2, 4))
##
## Coefficients:
##           ar1      ar2     ma1      ma2      ma3     ma4
##       -1.2800  -0.3022  0.7295  -0.4817  -0.0822  0.2654
## s.e.   0.2462   0.2520  0.2370   0.1381   0.1642  0.0804
##
## sigma^2 estimated as 0.02561:  log likelihood = 100.98,  aic = -189.96
```

```
Box.test(m1$residuals,lag=12,type="Ljung")
```

```
##
##  Box-Ljung test
##
## data:  m1$residuals
## X-squared = 10.485, df = 12, p-value = 0.5735
```

```
plot(m1$residuals)
```




```
rbind(m1$coef-2*sqrt(diag(m1$var.coef)),m1$coef+2*sqrt(diag(m1$var.coef)))
```

```
##               ar1        ar2       ma1        ma2        ma3       ma4
## [1,] -1.7724991 -0.8061430 0.2554901 -0.7578341 -0.4105619 0.1046217
## [2,] -0.7875669  0.2017778 1.2034689 -0.2055692  0.2462177 0.4261385
```

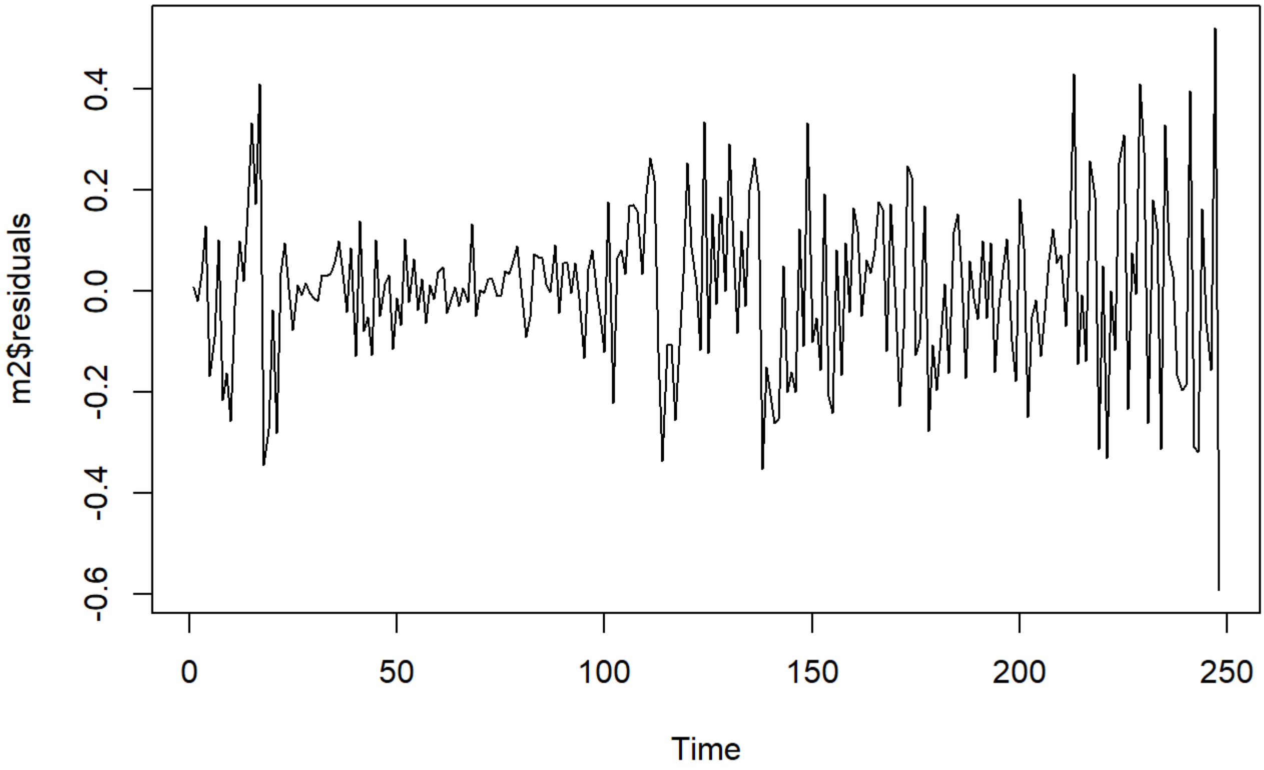
```
m2 = arima(gdpdata$gdpdef,
           order=c(2,2,4),
           fixed=c(NA,0,NA,NA,0,NA),
           transform.pars=F)
m2
```

```
##
## Call:
## arima(x = gdpdata$gdpdef, order = c(2, 2, 4), transform.pars = F, fixed = c(NA,
##     0, NA, NA, 0, NA))
##
## Coefficients:
##           ar1  ar2     ma1      ma2  ma3     ma4
##       -0.9847    0  0.4793  -0.6281    0  0.2435
## s.e.   0.0200    0  0.0582   0.0652    0  0.0699
##
## sigma^2 estimated as 0.02584:  log likelihood = 99.84,  aic = -191.68
```

```
Box.test(m2$residuals,lag=12,type="Ljung",fitdf=6-2)
```

```
##
##  Box-Ljung test
##
## data:  m2$residuals
## X-squared = 11.966, df = 8, p-value = 0.1527
```

```
plot(m2$residuals)
```



- 雖然 p-value 下降，但還是大於 0.05
- m2 AIC 小，選m2

擬合模型來預測 2009 年每個季度的通貨膨脹

```
forecast = predict(m2,n.ahead=4)
forecast$pred
```

```
## Time Series:
## Start = 249
## End = 252
## Frequency = 1
## [1] 123.7689 124.2963 124.9477 125.3327
```

4. 10 ·

Consider the daily simple returns of IBM stock, CRSP value-weighted index, CRSP equal-weighted index, and the S&P composite index from January 1980 to December 2008. The index returns include dividend distributions. The data file is d-ibm3dxwkdays8008.txt, which has 12 columns. The columns are (year, month, day, IBM, VW, EW, SP, M, T, W, H, F), where M, T, W, R, and F denotes indicator variables for Monday to Friday, respectively. Use a regression model to study the effects of trading days on the equal-weighted index returns. What is the fitted model? Are the weekday effects significant in the returns at the 5% level?

the daily simple returns

```
data4 = read.table("https://faculty.chicagobooth.edu/-/media/faculty/ruey-s-tsay/teaching/fts
3/d-ibm3dxwkdays8008.txt", header=T)
head(data4)
```

```
##   year mom day       ibm        vw        ew        sp M T W R F
## 1 1980   1   2 -0.029126 -0.020089 -0.011686 -0.020196 0 0 1 0 0
## 2 1980   1   3  0.016000 -0.006510 -0.011628 -0.005106 0 0 0 1 0
## 3 1980   1   4 -0.001969  0.013735  0.015809  0.012355 0 0 0 0 1
## 4 1980   1   7 -0.003945  0.004368  0.007013  0.002722 1 0 0 0 0
## 5 1980   1   8  0.067327  0.019340  0.014152  0.020036 0 1 0 0 0
## 6 1980   1   9 -0.029685  0.001714  0.007452  0.000918 0 0 1 0 0
```

What is the fitted model?

```
ml = lm(ew ~ M+T+W+R+F, data=data4)
summary(ml)
```

```
##
## Call:
## lm(formula = ew ~ M + T + W + R + F, data = data4)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.102962 -0.003094  0.000533  0.003795  0.108319
##
## Coefficients: (1 not defined because of singularities)
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0022386  0.0002155  10.389  < 2e-16 ***
## M           -0.0031734  0.0003085 -10.286  < 2e-16 ***
## T           -0.0019778  0.0003028  -6.532 6.94e-11 ***
## W           -0.0010185  0.0003027  -3.365 0.000770 ***
## R           -0.0010294  0.0003042  -3.384 0.000719 ***
## F                   NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.008234 on 7314 degrees of freedom
## Multiple R-squared:  0.01618,    Adjusted R-squared:  0.01564
## F-statistic: 30.06 on 4 and 7314 DF,  p-value: < 2.2e-16
```

```
step(ml)
```

```
## Start:  AIC=-70250.08
## ew ~ M + T + W + R + F
##
##
## Step:  AIC=-70250.08
## ew ~ M + T + W + R
##
##        Df Sum of Sq     RSS    AIC
## <none>              0.49586 -70250
## - W     1 0.0007675 0.49663 -70241
## - R     1 0.0007762 0.49664 -70241
## - T     1 0.0028923 0.49876 -70210
## - M     1 0.0071735 0.50304 -70147
```

```
##
## Call:
## lm(formula = ew ~ M + T + W + R, data = data4)
##
## Coefficients:
## (Intercept)            M            T            W            R
##    0.002239    -0.003173    -0.001978    -0.001019    -0.001029
```

```
ml2 = lm(ew ~ M+T+W+F+R, data=data4)
summary(ml2)
```

```
##
## Call:
## lm(formula = ew ~ M + T + W + F + R, data = data4)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.102962 -0.003094  0.000533  0.003795  0.108319
##
## Coefficients: (1 not defined because of singularities)
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0012092  0.0002148   5.631 1.86e-08 ***
## M           -0.0021440  0.0003080  -6.961 3.67e-12 ***
## T           -0.0009484  0.0003023  -3.137 0.001711 **
## W            0.0000109  0.0003022   0.036 0.971239
## F            0.0010294  0.0003042   3.384 0.000719 ***
## R                   NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.008234 on 7314 degrees of freedom
## Multiple R-squared:  0.01618,    Adjusted R-squared:  0.01564
## F-statistic: 30.06 on 4 and 7314 DF,  p-value: < 2.2e-16
```

```
step(ml2)
```

```
## Start:  AIC=-70250.08
## ew ~ M + T + W + F + R
##
##
## Step:  AIC=-70250.08
## ew ~ M + T + W + F
##
##        Df Sum of Sq     RSS    AIC
## - W     1 0.0000001 0.49586 -70252
## <none>              0.49586 -70250
## - T     1 0.0006673 0.49653 -70242
## - F     1 0.0007762 0.49664 -70241
## - M     1 0.0032852 0.49915 -70204
##
## Step:  AIC=-70252.08
## ew ~ M + T + F
##
##        Df Sum of Sq     RSS    AIC
## <none>              0.49586 -70252
## - T     1 0.0009061 0.49677 -70241
## - F     1 0.0010262 0.49689 -70239
## - M     1 0.0043768 0.50024 -70190
```

```
##
## Call:
## lm(formula = ew ~ M + T + F, data = data4)
##
## Coefficients:
## (Intercept)            M            T            F
##   0.0012147   -0.0021495   -0.0009539    0.0010239
```

```
ml3 = lm(ew ~ M+T+F+R+W, data=data4)
summary(ml3)
```

```
##
## Call:
## lm(formula = ew ~ M + T + F + R + W, data = data4)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.102962 -0.003094  0.000533  0.003795  0.108319
##
## Coefficients: (1 not defined because of singularities)
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0012201  0.0002126   5.739 9.91e-09 ***
## M           -0.0021549  0.0003065  -7.031 2.24e-12 ***
## T           -0.0009593  0.0003008  -3.190  0.00143 **
## F            0.0010185  0.0003027   3.365  0.00077 ***
## R           -0.0000109  0.0003022  -0.036  0.97124
## W                   NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.008234 on 7314 degrees of freedom
## Multiple R-squared:  0.01618,    Adjusted R-squared:  0.01564
## F-statistic: 30.06 on 4 and 7314 DF,  p-value: < 2.2e-16
```

```
step(ml3)
```

```
## Start:  AIC=-70250.08
## ew ~ M + T + F + R + W
##
##
## Step:  AIC=-70250.08
## ew ~ M + T + F + R
##
##        Df Sum of Sq     RSS    AIC
## - R     1 0.0000001 0.49586 -70252
## <none>              0.49586 -70250
## - T     1 0.0006897 0.49655 -70242
## - F     1 0.0007675 0.49663 -70241
## - M     1 0.0033513 0.49922 -70203
##
## Step:  AIC=-70252.08
## ew ~ M + T + F
##
##        Df Sum of Sq     RSS    AIC
## <none>              0.49586 -70252
## - T     1 0.0009061 0.49677 -70241
## - F     1 0.0010262 0.49689 -70239
## - M     1 0.0043768 0.50024 -70190
```

```
##
## Call:
## lm(formula = ew ~ M + T + F, data = data4)
##
## Coefficients:
## (Intercept)            M            T            F
##   0.0012147   -0.0021495   -0.0009539    0.0010239
```

```
ml4 = lm(ew ~ M+W+F+R+T, data=data4)
summary(ml4)
```

```
##
## Call:
## lm(formula = ew ~ M + W + F + R + T, data = data4)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.102962 -0.003094  0.000533  0.003795  0.108319
##
## Coefficients: (1 not defined because of singularities)
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0002608  0.0002127   1.226  0.22028
## M           -0.0011956  0.0003066  -3.900 9.72e-05 ***
## W            0.0009593  0.0003008   3.190  0.00143 **
## F            0.0019778  0.0003028   6.532 6.94e-11 ***
## R            0.0009484  0.0003023   3.137  0.00171 **
## T                   NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.008234 on 7314 degrees of freedom
## Multiple R-squared:  0.01618,    Adjusted R-squared:  0.01564
## F-statistic: 30.06 on 4 and 7314 DF,  p-value: < 2.2e-16
```

```
step(ml4)
```

```
## Start:  AIC=-70250.08
## ew ~ M + W + F + R + T
##
##
## Step:  AIC=-70250.08
## ew ~ M + W + F + R
##
##        Df  Sum of Sq     RSS    AIC
## <none>               0.49586 -70250
## - R     1 0.00066734 0.49653 -70242
## - W     1 0.00068972 0.49655 -70242
## - M     1 0.00103096 0.49689 -70237
## - F     1 0.00289230 0.49876 -70210
```

```
##
## Call:
## lm(formula = ew ~ M + W + F + R, data = data4)
##
## Coefficients:
## (Intercept)            M            W            F            R
##   0.0002608   -0.0011956    0.0009593    0.0019778    0.0009484
```

```
ml5 = lm(ew ~ T+W+F+R+M, data=data4)
summary(ml5)
```

```
##
## Call:
## lm(formula = ew ~ T + W + F + R + M, data = data4)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.102962 -0.003094  0.000533  0.003795  0.108319
##
## Coefficients: (1 not defined because of singularities)
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0009348  0.0002208  -4.234 2.32e-05 ***
## T            0.0011956  0.0003066   3.900 9.72e-05 ***
## W            0.0021549  0.0003065   7.031 2.24e-12 ***
## F            0.0031734  0.0003085  10.286  < 2e-16 ***
## R            0.0021440  0.0003080   6.961 3.67e-12 ***
## M                   NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.008234 on 7314 degrees of freedom
## Multiple R-squared:  0.01618,    Adjusted R-squared:  0.01564
## F-statistic: 30.06 on 4 and 7314 DF,  p-value: < 2.2e-16
```

```
step(ml5)
```

```
## Start:  AIC=-70250.08
## ew ~ T + W + F + R + M
##
##
## Step:  AIC=-70250.08
## ew ~ T + W + F + R
##
##        Df Sum of Sq     RSS    AIC
## <none>               0.49586 -70250
## - T     1 0.0010310 0.49689 -70237
## - R     1 0.0032852 0.49915 -70204
## - W     1 0.0033513 0.49922 -70203
## - F     1 0.0071735 0.50304 -70147
```

```
##
## Call:
## lm(formula = ew ~ T + W + F + R, data = data4)
##
## Coefficients:
## (Intercept)            T            W            F            R
##  -0.0009348    0.0011956    0.0021549    0.0031734    0.0021440
```

- AIC 最低的模型:lm(formula = ew ~ M + T + F, data = data4)

```
best = lm(formula = ew ~ M + T + F, data = data4)
summary(best)
```

```
##
## Call:
## lm(formula = ew ~ M + T + F, data = data4)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.102962 -0.003094  0.000533  0.003792  0.108319
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0012147  0.0001511   8.040 1.04e-15 ***
## M           -0.0021495  0.0002675  -8.035 1.08e-15 ***
## T           -0.0009539  0.0002609  -3.656 0.000258 ***
## F            0.0010239  0.0002632   3.891 0.000101 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.008233 on 7315 degrees of freedom
## Multiple R-squared:  0.01617,    Adjusted R-squared:  0.01577
## F-statistic: 40.09 on 3 and 7315 DF,  p-value: < 2.2e-16
```

Are the weekday effects significant in the returns at the 5% level?

- 星期一、二、五在上面每個模型中，p-value 均小於0.05，故這幾天顯著影響報酬率
- 但其實大部分的模型每天對報酬率都有顯著影響，AIC間也變化不大

5. 6.

Now consider similar questions of the previous exercise for the IBM stock returns.(d-ibm3dxwkdays8008.txt)
(a) Is there any weekday effect on the daily simple returns of IBM stock? Estimate your model and test the hypothesis that there is no Friday effect. Draw your conclusion.

```
ibm = lm(ibm ~ M + T + W + R + F ,data=data4)
summary(ibm)
```

```
##
## Call:
## lm(formula = ibm ~ M + T + W + R + F, data = data4)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.231629 -0.009290 -0.000036  0.008840  0.131619
##
## Coefficients: (1 not defined because of singularities)
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0005902  0.0004671  -1.264 0.206382
## M            0.0025896  0.0006687   3.873 0.000109 ***
## T            0.0020296  0.0006563   3.092 0.001992 **
## W            0.0002289  0.0006561   0.349 0.727217
## R            0.0006073  0.0006594   0.921 0.357085
## F                   NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01785 on 7314 degrees of freedom
## Multiple R-squared:  0.003269,	Adjusted R-squared:  0.002723
## F-statistic: 5.996 on 4 and 7314 DF,  p-value: 8.178e-05
```

- model:$R_t = -0.0005902 + 0.0025896M + 0.0020296T + 0.0002289W + 0.0006073R$
- 星期一和星期二在不考慮星期五影響下 p-value < 0.05，顯著影響報酬率

X

```
c = step(ibm)
```

就平均而言，週一與週二顯著異於週五

-2

```
## Start:  AIC=-58927.08
## ibm ~ M + T + W + R + F
##
##
## Step:  AIC=-58927.08
## ibm ~ M + T + W + R
##
##          Df Sum of Sq    RSS    AIC
## - W      1 0.0000388 2.3295 -58929
## - R      1 0.0002701 2.3297 -58928
## <none>               2.3294 -58927
## - T      1 0.0030458 2.3325 -58920
## - M      1 0.0047770 2.3342 -58914
##
## Step:  AIC=-58928.96
## ibm ~ M + T + R
##
##          Df Sum of Sq    RSS    AIC
## - R      1 0.0002371 2.3297 -58930
## <none>               2.3295 -58929
## - T      1 0.0036424 2.3331 -58920
## - M      1 0.0057903 2.3352 -58913
##
## Step:  AIC=-58930.22
## ibm ~ M + T
##
##          Df Sum of Sq    RSS    AIC
## <none>               2.3297 -58930
## - T      1 0.0034308 2.3331 -58921
## - M      1 0.0056518 2.3353 -58914
```

```
c
```

```
##
## Call:
## lm(formula = ibm ~ M + T, data = data4)
##
## Coefficients:
## (Intercept)            M            T
##  -0.0003112    0.0023106    0.0017506
```

- best model: $R_t = -0.0003112 + 0.0023106M + 0.0017506T$

b. Are there serial correlations in the residuals?
Use Q(12) to perform the test.Draw your conclusion

```
Box.test(c$residuals,lag=12,type='Ljung')
```

fitdf - 2

```
##
##  Box-Ljung test
##
## data:  c$residuals
## X-squared = 16.936, df = 12, p-value = 0.152
```

- NO，因為 p-value > 0.05，不拒絕 H0 (無序列相關)