Mining Time Series Data

Hazim Fitri

2024-12-08

Contents

1. Convert any data type into "Date" data type ISO 8601

1. %Y = 4 digit year (2000)
2. %y = 2 digit year (00)
3. %m = 2 digit month (12)
4. %B = month (December)
5. %b = abbreviated month (Dec)
6. %d = 2 digit day (03)
7. %A = day (Sunday)
8. %a = abbreviated day (Sun)

```
Sys.Date()
```

```
## [1] "2025-01-01"
```

```
sample_data <- rnorm(60)
ts(sample_data, start=c(2020,2), frequency = 12)
```

```
##              Jan         Feb         Mar         Apr         May         Jun
## 2020              0.17016324  0.23670034 -0.42860799 -0.15129134  0.49403406
## 2021 -0.09925246 -0.16752979  0.12142728  1.07648395  1.06195594  0.99249071
## 2022  1.17536812 -1.09616870  0.54104719  0.51341970 -0.04718075  0.47194056
## 2023 -0.49869736  0.63820781 -2.68343915  0.26780544 -0.11159016  1.70165829
## 2024 -0.25772145 -0.62890975 -0.55113217  0.59556109 -0.66593191  0.96359685
## 2025  1.00213540
##              Jul         Aug         Sep         Oct         Nov         Dec
## 2020  0.74220417 -0.21661272 -0.09894000  0.18690987 -0.24635221 -1.09619793
## 2021  1.41485904  0.03675725 -0.38611118  0.27469286 -0.60197077 -0.31718849
## 2022  0.28554395 -1.76528020  0.49964157 -1.52364312  1.23291697  0.51600801
## 2023  0.24721166 -1.88468275  0.07077981  0.67299896  0.43106153  0.93749779
## 2024  1.63249710  0.82985680 -0.53640139  0.29061587  1.15029042 -0.86303796
## 2025
```

```r
dates_format <- read.csv('dates_formats3.csv', sep = ';')
head(dates_format)
```

```
##   Japanese_format US_format     CA_mix_format    SA_mix_format  NZ_format
## 1      20/01/2017 1/20/2017 January 20, 2017 20 January 2017 20/01/2017
## 2      21/01/2017 1/21/2017 January 21, 2017 21 January 2017 21/01/2017
## 3      22/01/2017 1/22/2017 January 22, 2017 22 January 2017 22/01/2017
## 4      23/01/2017 1/23/2017 January 23, 2017 23 January 2017 23/01/2017
## 5      24/01/2017 1/24/2017 January 24, 2017 24 January 2017 24/01/2017
## 6      25/01/2017 1/25/2017 January 25, 2017 25 January 2017 25/01/2017
```

```r
attach(dates_format)
class(Japanese_format)
```

```
## [1] "character"
```

```r
dates_format$Japanese_format <- as.Date(Japanese_format, format = "%d/%m/%Y")
dates_format
```

```
##    Japanese_format US_format     CA_mix_format    SA_mix_format  NZ_format
## 1       2017-01-20 1/20/2017  January 20, 2017  20 January 2017 20/01/2017
## 2       2017-01-21 1/21/2017  January 21, 2017  21 January 2017 21/01/2017
## 3       2017-01-22 1/22/2017  January 22, 2017  22 January 2017 22/01/2017
## 4       2017-01-23 1/23/2017  January 23, 2017  23 January 2017 23/01/2017
## 5       2017-01-24 1/24/2017  January 24, 2017  24 January 2017 24/01/2017
## 6       2017-01-25 1/25/2017  January 25, 2017  25 January 2017 25/01/2017
## 7       2017-01-26 1/26/2017  January 26, 2017  26 January 2017 26/01/2017
## 8       2017-01-27 1/27/2017  January 27, 2017  27 January 2017 27/01/2017
## 9       2017-01-28 1/28/2017  January 28, 2017  28 January 2017 28/01/2017
## 10      2017-01-29 1/29/2017  January 29, 2017  29 January 2017 29/01/2017
## 11      2017-01-30 1/30/2017  January 30, 2017  30 January 2017 30/01/2017
## 12      2017-01-31 1/31/2017  January 31, 2017  31 January 2017 31/01/2017
## 13      2017-02-01 2/01/2017  February 1, 2017 01 February 2017  1/02/2017
## 14      2017-02-02 2/02/2017  February 2, 2017 02 February 2017  2/02/2017
## 15      2017-02-03 2/03/2017  February 3, 2017 03 February 2017  3/02/2017
## 16      2017-02-04 2/04/2017  February 4, 2017 04 February 2017  4/02/2017
## 17      2017-02-05 2/05/2017  February 5, 2017 05 February 2017  5/02/2017
## 18      2017-02-06 2/06/2017  February 6, 2017 06 February 2017  6/02/2017
## 19      2017-02-07 2/07/2017  February 7, 2017 07 February 2017  7/02/2017
## 20      2017-02-08 2/08/2017  February 8, 2017 08 February 2017  8/02/2017
## 21      2017-02-09 2/09/2017  February 9, 2017 09 February 2017  9/02/2017
## 22      2017-02-10 2/10/2017 February 10, 2017 10 February 2017 10/02/2017
```

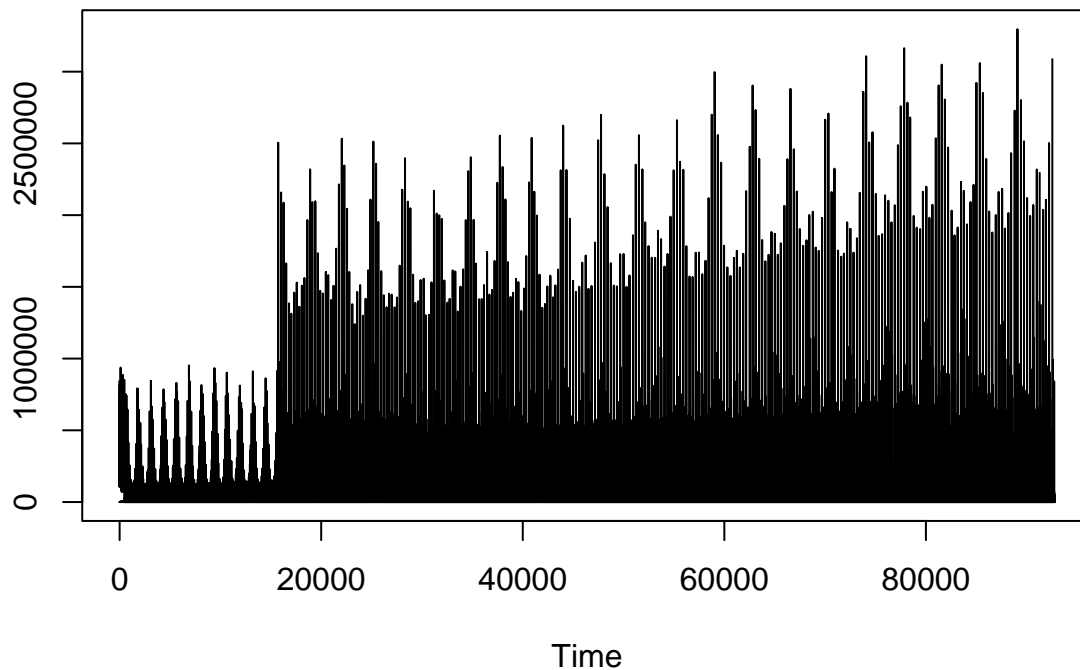## 2. Time Series Decomposition

Time series can be decomposed into 4 components (TSCR) Trend, Seasonal, Cyclical, Random

2.1 Additive Decomposition

```r
library(USgas)
```

```
## Warning: package 'USgas' was built under R version 4.4.2
```

```r
data(package = 'USgas')
data(usgas)
ts.plot(usgas)
```

```r
data(USVsales)
```

```
## Warning in data(USVsales): data set 'USVsales' not found
```

2.2 Multiplicative Decomposition

2. Time Series Forecasting

```r
towel <- read.csv('towel.csv')
head(towel)
```

```
##         y
## 1 15.0000
## 2 14.4064
## 3 14.9383
## 4 16.0374
## 5 15.6320
## 6 14.3975
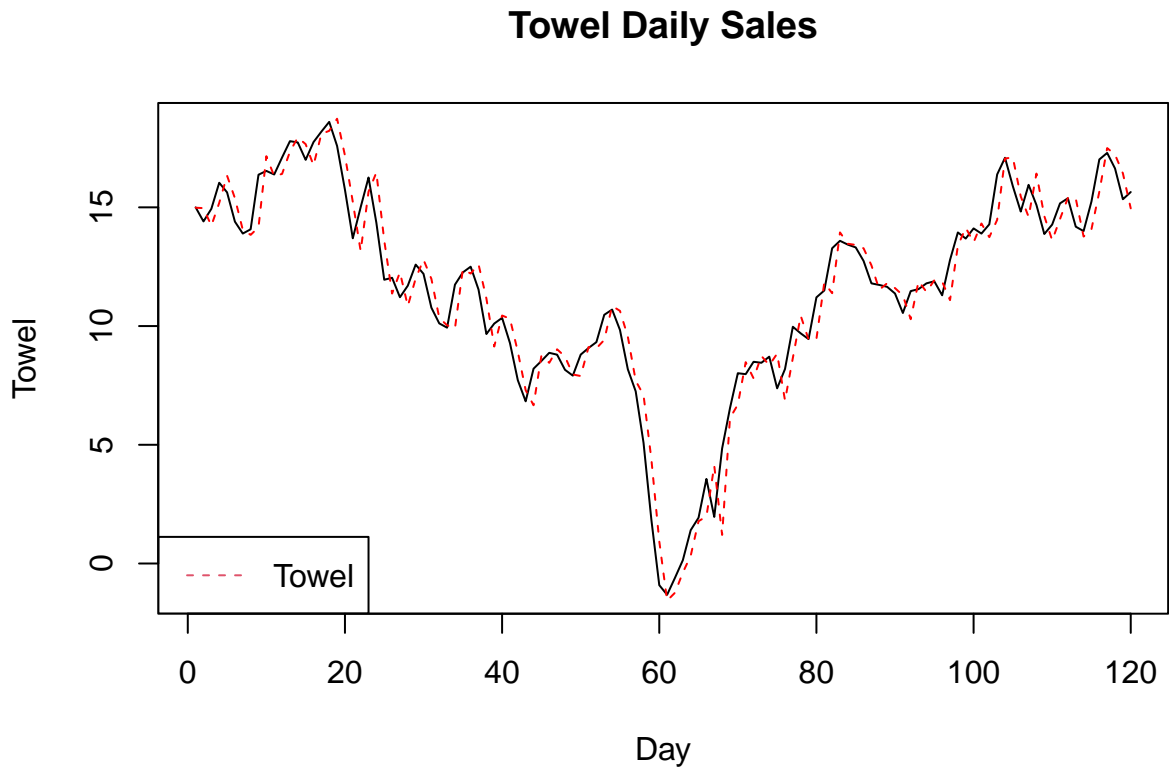```

```r
head(ts(towel))
```

```
##          y
## [1,] 15.0000
## [2,] 14.4064
## [3,] 14.9383
## [4,] 16.0374
## [5,] 15.6320
## [6,] 14.3975
```

```r
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 4.4.2
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
ts.plot(ts(towel), main = 'Towel Daily Sales', xlab = 'Day', ylab = 'Towel')
lines(fitted(auto.arima(ts(towel))), col = 'red', lty = 2)
legend('bottomleft', 'Towel', col = 2, lty = 2)
```

Towel Daily Sales

We can see that the data is not stationary (pegun). How to determine whether a data is stationary or not?

A stationary data satisfy these conditions:

1. Mean is constant over time
2. Variance is constant over time

Then, what should we do if the data is not stationary, we can do differencing (pembezaan) on the data manually or can do it automatically

```
auto.arima(ts(towel))
```

```
## Series: ts(towel)
## ARIMA(0,1,1)
##
## Coefficients:
##          ma1
##       0.3518
## s.e.  0.0800
##
## sigma^2 = 1.08:  log likelihood = -172.99
## AIC=349.98   AICc=350.08   BIC=355.53
```

after we find the ARIMA model,

```
model <- arima(ts(towel), order = c(0,1,1))
summary(model)
```

```
##
## Call:
## arima(x = ts(towel), order = c(0, 1, 1))
##
## Coefficients:
##          ma1
##       0.3518
## s.e.  0.0800
##
## sigma^2 estimated as 1.071:  log likelihood = -172.99,  aic = 349.98
##
## Training set error measures:
##                        ME     RMSE       MAE      MPE    MAPE      MASE
## Training set 0.005711291 1.030456 0.8053392 3.003594 15.5468 0.9337157
##                      ACF1
## Training set 0.007198121
```

```
f1 <- forecast(model)
f1
```

```
##     Point Forecast    Lo 80    Hi 80     Lo 95    Hi 95
## 121       15.88729 14.56117 17.21341 13.859170 17.91542
## 122       15.88729 13.65741 18.11717 12.476983 19.29760
## 123       15.88729 13.02615 18.74844 11.511549 20.26304
## 124       15.88729 12.51091 19.26367 10.723567 21.05102
## 125       15.88729 12.06450 19.71008 10.040841 21.73375
## 126       15.88729 11.66503 20.10956  9.429900 22.34469
## 127       15.88729 11.30022 20.47437  8.871963 22.90262
## 128       15.88729 10.96235 20.81224  8.355243 23.41934
## 129       15.88729 10.64622 21.12837  7.871764 23.90282
## 130       15.88729 10.34810 21.42648  7.415833 24.35875
```

```
f2 <- forecast(auto.arima(ts(towel)))
f2
```

```
##     Point Forecast    Lo 80    Hi 80     Lo 95    Hi 95
## 121       15.88729 14.55557 17.21902 13.850593 17.92399
## 122       15.88729 13.64798 18.12661 12.462560 19.31203
## 123       15.88729 13.01405 18.76054 11.493042 20.28154
## 124       15.88729 12.49663 19.27795 10.701728 21.07286
## 125       15.88729 12.04834 19.72625 10.016115 21.75847
## 126       15.88729 11.64717 20.12741  9.402590 22.37200
## 127       15.88729 11.28082 20.49377  8.842294 22.93229
## 128       15.88729 10.94152 20.83307  8.323388 23.45120
## 129       15.88729 10.62405 21.15053  7.837864 23.93672
## 130       15.88729 10.32468 21.44991  7.380005 24.39458
```

```
predict(auto.arima(ts(towel)), n.ahead = 10)
```

```
## $pred
## Time Series:
## Start = 121
## End = 130
## Frequency = 1
##  [1] 15.88729 15.88729 15.88729 15.88729 15.88729 15.88729 15.88729 15.88729
##  [9] 15.88729 15.88729
##
## $se
```

```
## Time Series:
## Start = 121
## End = 130
## Frequency = 1
##  [1] 1.039152 1.747345 2.242006 2.645745 2.995554 3.308583 3.594454 3.859206
##  [9] 4.106927 4.340533
```

str(f1)

```
## List of 10
##  $ method   : chr "ARIMA(0,1,1)"
##  $ model    :List of 15
##   ..$ coef     : Named num 0.352
##   .. ..- attr(*, "names")= chr "ma1"
##   ..$ sigma2   : num 1.07
##   ..$ var.coef : num [1, 1] 0.00641
##   .. ..- attr(*, "dimnames")=List of 2
##   .. .. ..$ : chr "ma1"
##   .. .. ..$ : chr "ma1"
##   ..$ mask     : logi TRUE
##   ..$ loglik   : num -173
##   ..$ aic      : num 350
##   ..$ arma     : int [1:7] 0 1 0 0 1 1 0
##   ..$ residuals: Time-Series [1:120] from 1 to 120: 0.015 -0.56 0.713 0.849 -0.704 ...
##   ..$ call     : language arima(x = ts(towel), order = c(0, 1, 1))
##   ..$ series   : chr "ts(towel)"
##   ..$ code     : int 0
##   ..$ n.cond   : int 0
##   ..$ nobs     : int 119
##   ..$ model    :List of 10
##   .. ..$ phi  : num(0)
##   .. ..$ theta: num 0.352
##   .. ..$ Delta: num 1
##   .. ..$ Z    : num [1:3] 1 0 1
##   .. ..$ a    : num [1:3] 0.304 0.242 15.341
##   .. ..$ P    : num [1:3, 1:3] 0.00 0.00 2.17e-21 0.00 0.00 ...
##   .. ..$ T    : num [1:3, 1:3] 0 0 1 1 0 0 0 0 1
##   .. ..$ V    : num [1:3, 1:3] 1 0.352 0 0.352 0.124 ...
##   .. ..$ h    : num 0
##   .. ..$ Pn   : num [1:3, 1:3] 1.00 3.52e-01 -1.18e-21 3.52e-01 1.24e-01 ...
##   ..$ x        : Time-Series [1:120, 1] from 1 to 120: 15 14.4 14.9 16 15.6 ...
##   .. ..- attr(*, "dimnames")=List of 2
##   .. .. ..$ : NULL
##   .. .. ..$ : chr "y"
##   ..- attr(*, "class")= chr "Arima"
##  $ level    : num [1:2] 80 95
##  $ mean     : Time-Series [1:10] from 121 to 130: 15.9 15.9 15.9 15.9 15.9 ...
##  $ lower    : Time-Series [1:10, 1:2] from 121 to 130: 14.6 13.7 13 12.5 12.1 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:2] "80%" "95%"
##  $ upper    : Time-Series [1:10, 1:2] from 121 to 130: 17.2 18.1 18.7 19.3 19.7 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:2] "80%" "95%"
##  $ x        : Time-Series [1:120, 1] from 1 to 120: 15 14.4 14.9 16 15.6 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr "y"
##  $ series   : chr "ts(towel)"
##  $ fitted   : Time-Series [1:120, 1] from 1 to 120: 15 15 14.2 15.2 16.3 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
```

```
##   .. ..$ : chr "x"
##  $ residuals: Time-Series [1:120] from 1 to 120: 0.015 -0.56 0.713 0.849 -0.704 ...
##  - attr(*, "class")= chr "forecast"
```

```
f1$lower[,1]
```

```
## Time Series:
## Start = 121
## End = 130
## Frequency = 1
##  [1] 14.56117 13.65741 13.02615 12.51091 12.06450 11.66503 11.30022 10.96235
##  [9] 10.64622 10.34810
```

```
identical(f1, f2)
```

```
## [1] FALSE
```

3. Time Series Clustering

```
load('sample2.RData')
str(sample2)
```

```
## 'data.frame':     60 obs. of  60 variables:
##  $ V1 : num   28.2 29.2 27.2 24.1 24 ...
##  $ V2 : num   30.8 31.5 25.8 33 32.9 ...
##  $ V3 : num   33.9 33.2 28.7 35.5 33.9 ...
##  $ V4 : num   26.7 35.6 31.4 28.6 33.3 ...
##  $ V5 : num   27.5 31.6 31.6 28.8 30.3 ...
##  $ V6 : num   28.1 35.1 30.2 32.5 28.8 ...
##  $ V7 : num   32.5 28.6 31.6 30 25.6 ...
##  $ V8 : num   33.7 34.3 32.2 24.3 24.3 ...
##  $ V9 : num   26 33.1 30.9 24.4 24.2 ...
##  $ V10: num   32.7 35.7 30.7 34.1 33.2 ...
##  $ V11: num   34.3 29.7 34.8 30.4 34 ...
##  $ V12: num   30.3 28.2 31.9 33.7 33.8 ...
##  $ V13: num   29.8 30.5 29.4 29.9 28.3 ...
##  $ V14: num   33 24.8 27.9 33.2 31.1 ...
##  $ V15: num   35.6 26.4 27.1 33.2 34.5 ...
##  $ V16: num   27.7 28 27.3 29.4 24.5 ...
##  $ V17: num   32.9 33.6 30.9 30 24.2 ...
##  $ V18: num   28.4 34 31 25.3 30.3 ...
##  $ V19: num   28.5 33.8 31.8 27.8 31.8 ...
##  $ V20: num   25.6 24.5 29.1 24.6 24.1 ...
##  $ V21: num   30.6 26.1 34.1 34.9 25.3 ...
##  $ V22: num   31.9 32.9 31.9 27.4 30.6 ...
##  $ V23: num   29.9 27.3 31.2 31 33.5 ...
##  $ V24: num   28.3 35.6 29.9 25 31.7 ...
##  $ V25: num   32 31.4 31.6 34.9 35.9 ...
##  $ V26: num   29.8 26.9 27.8 29 33.1 ...
##  $ V27: num   35.3 24.2 32 30.1 35.1 ...
##  $ V28: num   34.2 25.9 32.6 25.3 33.5 ...
##  $ V29: num   32.3 30.7 34.8 35.3 28.4 ...
##  $ V30: num   34.4 32.7 24.7 27.4 33.5 ...
##  $ V31: num   32 24.3 32.6 24 30.4 ...
##  $ V32: num   28.6 29.6 36 28 28 ...
##  $ V33: num   24.4 27.6 35.3 26.3 25.9 ...
##  $ V34: num   33.5 26.7 30.4 29.5 32.1 ...
##  $ V35: num   30.4 24.4 32.6 29.7 30.6 ...
##  $ V36: num   28.4 34.2 26.3 30.1 35.5 ...
##  $ V37: num   35.2 27.6 31.3 24.3 27.1 ...
```

```
##  $ V38: num   33.5 24.3 29.6 31 28 ...
##  $ V39: num   32.7 32.6 25.3 27.7 34.6 ...
##  $ V40: num   28.1 25.1 29.1 25.7 27.3 ...
##  $ V41: num   35.2 35.3 29.5 28.3 28.4 ...
##  $ V42: num   25.1 36 33.7 34.6 31.9 ...
##  $ V43: num   29.6 34.3 31.6 28.1 30 ...
##  $ V44: num   29.4 26.9 29.6 32.6 26.2 ...
##  $ V45: num   33.5 31.7 33.9 28.1 35.5 ...
##  $ V46: num   30.8 35.5 27.5 33.8 26.5 ...
##  $ V47: num   27.5 28.3 30.2 24.6 26.4 ...
##  $ V48: num   26.2 35.6 32.2 32.5 25.4 ...
##  $ V49: num   25.8 31.8 35.4 26.3 32.2 ...
##  $ V50: num   27.2 33.8 29.3 25.2 31.3 ...
##  $ V51: num   30.1 35.7 33.6 31 33.8 ...
##  $ V52: num   32.1 25.5 31.9 26.4 25.5 ...
##  $ V53: num   35.4 30.6 29.3 32.3 27.1 ...
##  $ V54: num   34.6 33.3 24.3 35.3 31.5 ...
##  $ V55: num   24.1 33 24 27.2 34.8 ...
##  $ V56: num   34.7 24 34.7 34.7 28.8 ...
##  $ V57: num   35.3 32.8 25.8 25.7 24.4 ...
##  $ V58: num   34.1 29.2 34.9 32 35.4 ...
##  $ V59: num   24.8 35.5 32.2 33.9 31.7 ...
##  $ V60: num   33.7 25.9 32.5 26 26.2 ...
```

```r
library(dtw)
```

```
## Warning: package 'dtw' was built under R version 4.4.2

## Loading required package: proxy

## Warning: package 'proxy' was built under R version 4.4.2

##
## Attaching package: 'proxy'

## The following objects are masked from 'package:stats':
##
##     as.dist, dist

## The following object is masked from 'package:base':
##
##     as.matrix

## Loaded dtw v1.23-1. See ?dtw for help, citation("dtw") for use in publication.
```

```r
dist(sample2, method = 'DTW')
```

```
##              16        10        58        15        82        91        11
## 10    175.1280
## 58    144.1657  171.9889
## 15    166.9458  163.0294  196.0601
## 82    170.7334  165.2784  192.9108  161.3009
## 91    163.6098  172.9859  166.7570  184.2107  168.7043
## 11    195.9865  182.3403  204.9765  190.9235  187.0962  200.3000
## 9     183.9797  174.1057  190.2381  174.8510  187.2452  193.2259  174.6517
## 57    182.9158  176.2096  179.0649  160.2957  165.0189  173.8225  172.1541
## 49    163.1556  182.0432  161.1173  171.6772  187.3219  177.0821  178.5544
## 116   329.2900  331.0366  347.1432  345.3147  351.3869  340.2702  375.7339
## 110   412.4996  389.3421  431.4444  392.5875  411.7251  406.7473  386.8013
## 158   306.7922  338.4695  319.0572  316.3249  318.0288  315.0440  306.1883
## 115   314.6060  301.5108  309.3024  318.5051  330.4084  297.9523  323.8169
```

```
## 182   391.6906   360.7782   379.5559   380.0575   391.5457   386.0444   358.2902
## 191   457.1948   422.9261   455.5199   447.3367   436.1005   432.1660   452.7524
## 111   457.5511   482.9348   465.5950   491.5149   508.8061   462.9060   468.8489
## 109   375.7010   378.9904   398.2781   396.3763   374.5768   379.9558   388.9362
## 157   463.2064   480.4719   464.1849   509.7655   516.7934   474.7728   509.2513
## 149   446.7099   437.0666   450.0722   454.6354   444.7990   455.3046   416.1305
## 216   826.4922   800.3383   804.5198   878.7758   831.0711   757.4787   884.9138
## 210   740.6852   719.9976   744.3946   788.5271   744.6802   688.3748   799.1393
## 258   511.5507   500.6385   488.5145   548.2394   514.5929   469.2607   584.2304
## 215   563.7351   551.2358   554.8192   607.5049   585.5175   530.9445   632.6973
## 282   418.3971   422.6791   432.1538   462.0719   446.1832   399.4993   495.5733
## 291   479.8153   486.2968   463.1286   535.5405   501.3543   442.9482   551.7522
## 211   394.2444   406.8931   392.6001   442.4436   426.2136   374.5052   464.4990
## 209   737.2810   715.3253   726.4019   787.6761   757.0085   680.9282   820.6890
## 257   608.3304   590.2800   610.3613   655.7898   620.8541   558.5648   672.2675
## 249   672.1137   658.3494   642.9390   720.7345   684.7243   612.2686   758.1122
## 316   782.3056   815.9420   813.9956   818.4009   806.9538   848.5733   907.5346
## 310   674.1048   707.6508   689.2203   702.2251   695.7891   728.7550   775.4822
## 358   759.5318   784.7909   797.9506   781.0134   782.3751   821.6977   865.3701
## 315   689.7036   705.7095   718.9107   709.2578   696.3422   750.6292   778.4101
## 382   647.0202   686.7918   659.5400   681.4913   671.3781   710.2219   756.8518
## 391   756.6014   798.7839   795.8240   790.7428   789.2344   815.0340   881.6153
## 311   794.9235   819.5027   822.4442   822.9499   817.2233   862.2246   904.9188
## 309   431.9679   453.1353   454.3822   459.3160   423.5933   471.7541   476.5173
## 357   296.8420   312.9019   306.5631   319.9087   304.2201   335.3904   317.8638
## 349   379.9298   398.0071   372.8486   426.9845   397.8384   431.0043   400.7940
## 416   427.3556   424.1760   425.1307   457.8307   422.6760   394.1854   477.5533
## 410   416.7125   433.9305   394.3301   473.7980   445.8931   393.1413   492.0611
## 458   573.4030   585.8512   551.8523   627.9127   582.5162   548.9773   628.5763
## 415   459.4040   476.9384   461.7285   512.7724   486.3578   436.7941   531.4936
## 482   473.2471   473.0897   471.3512   514.5563   471.8153   444.8954   523.3485
## 491   274.9795   279.9997   267.9129   307.2508   297.0211   288.5079   316.6242
## 411   286.7039   279.8211   289.0663   303.0023   289.4373   280.2153   337.0956
## 409   514.3473   514.0479   494.7017   551.1865   517.6003   481.6217   563.1958
## 457   627.5547   604.3573   618.0067   663.9866   649.1852   589.5756   688.5234
## 449   603.8714   608.4002   588.0258   647.4232   611.7859   564.4548   655.9725
## 516   490.3972   498.2236   477.1504   525.3784   497.6744   530.9871   562.7056
## 510   298.1177   327.1161   299.6660   362.4249   320.4068   328.5151   340.9911
## 558   244.3351   247.3566   246.0886   244.9287   220.8859   263.4180   230.6287
## 515   458.4887   476.3788   497.8260   494.8522   484.7094   518.5549   551.7643
## 582   558.4922   585.4574   568.0021   631.9905   594.1992   620.1393   643.4169
## 591   391.3356   411.6931   432.5499   416.2209   405.6877   444.7050   486.9401
## 511   393.2345   384.5817   387.8119   436.5029   400.4979   429.6109   441.2595
## 509   416.6172   426.6450   461.7612   459.5775   434.6470   474.3889   467.1983
## 557   331.2201   329.0783   307.0269   377.7614   349.9872   361.6908   366.7179
## 549   587.0573   608.3581   618.6086   615.0500   603.0659   626.7392   670.9693
##               9         57         49        116        110        158        115
## 10
## 58
## 15
## 82
## 91
## 11
## 9
## 57   164.4125
## 49   153.0165   184.3468
## 116  344.8343   330.3620   345.6073
## 110  403.3874   416.9524   406.8755   295.5500
## 158  304.7006   320.7581   288.2042   336.7648   259.9603
## 115  311.4715   318.6337   306.1945   246.8503   332.2817   269.1133
## 182  385.7393   360.4200   380.9130   255.8917   231.3441   285.5706   278.2683
## 191  465.4668   447.7683   456.6107   289.3515   343.3710   339.6877   367.1080
## 111  456.7924   473.6344   456.5273   308.4226   258.1221   285.0677   311.8032
## 109  379.6993   387.0614   388.2885   358.1955   321.7303   313.6220   326.2104
```

```
## 157   488.1908   524.2834   477.3408   353.2030   289.1832   316.8130   312.5230
## 149   472.4503   454.4991   464.5975   334.0386   336.8638   314.4886   388.1866
## 216   899.9221   939.6868   737.8263   875.0836   866.1553   621.1048   733.7122
## 210   793.5789   846.6302   649.5476   784.8242   790.2005   566.8792   620.9641
## 258   582.9039   587.1045   452.8861   584.1407   590.6502   409.7886   478.9324
## 215   617.6892   658.4277   489.7379   704.2768   705.4041   488.7204   554.9464
## 282   486.5476   507.3609   370.0488   560.9853   582.4969   360.4161   400.0814
## 291   542.3131   564.3754   435.2316   590.8589   628.8956   443.8754   461.8727
## 211   451.0164   469.2712   354.7309   543.9814   579.7249   385.8108   424.3972
## 209   815.4536   855.0778   648.5208   807.0390   826.8892   587.6550   665.1433
## 257   671.0913   709.3697   542.0269   659.7261   690.0414   483.7147   538.8532
## 249   765.9814   783.2014   594.5434   763.3026   772.8675   516.0699   618.6798
## 316   881.4033   826.0864   815.1398   588.4384   791.0603   976.8035   937.3255
## 310   758.7420   693.1954   696.8148   540.0836   737.2655   870.0883   841.4203
## 358   860.3303   800.9359   783.1251   566.9694   794.0769   941.6001   940.3617
## 315   749.2224   709.0763   690.6125   496.3731   678.4012   817.0195   819.0900
## 382   739.7234   682.0211   671.7484   527.3991   657.4242   800.9801   821.8595
## 391   872.0570   796.9382   789.7751   587.7537   802.3301   983.8889   942.7474
## 311   892.3242   837.8666   829.8585   527.9664   769.3356   927.3580   946.3951
## 309   471.4559   433.0463   443.1499   420.8000   521.6431   598.7788   569.7374
## 357   324.0263   298.5065   300.3700   384.7282   471.4751   481.2695   463.0333
## 349   402.8837   387.8868   395.6313   364.6725   438.3759   473.6616   500.2464
## 416   456.8217   474.8644   384.1068   473.4160   482.2014   339.6419   417.8467
## 410   487.0535   500.9988   397.5624   618.8456   658.1216   411.3575   464.0841
## 458   639.9155   657.0336   533.4181   631.9233   623.2834   421.7111   535.4592
## 415   517.6733   549.4892   421.1514   550.3138   557.8984   403.1910   443.9433
## 482   541.7588   560.3967   435.4080   546.2614   550.3333   356.3866   462.2375
## 491   303.6780   297.5684   284.9972   434.1536   454.6514   314.9782   367.8393
## 411   309.0959   332.7755   263.3596   415.1493   446.1599   333.1699   326.0159
## 409   538.1757   569.7415   451.8028   527.6136   541.1377   376.7483   454.2626
## 457   677.6342   723.2457   563.0500   696.0687   672.6708   430.4637   560.9914
## 449   660.3213   697.8517   552.5136   586.8539   604.7479   391.8755   528.1111
## 516   546.1916   501.4544   517.9954   350.1469   503.2757   601.6501   580.0385
## 510   335.8130   310.1975   339.6613   362.2085   459.8880   435.2844   427.1881
## 558   224.2427   203.8213   242.6285   317.0965   426.2371   359.6406   367.1334
## 515   512.2972   466.8071   489.7389   370.9560   458.4230   537.5934   562.3735
## 582   629.0548   581.4827   609.6208   419.9534   534.3978   610.6819   598.2454
## 591   453.6550   424.0183   426.2505   396.3721   484.5519   566.2147   576.3478
## 511   424.4846   397.5772   431.2305   370.4580   454.1659   488.5243   481.1578
## 509   472.6665   433.1884   458.1481   353.4361   486.5024   551.8799   538.3884
## 557   353.5782   340.1391   374.0903   340.6480   399.8583   410.9724   444.8728
## 549   653.8127   617.0528   621.5486   405.7764   593.2543   701.7157   718.2135
##             182        191        111        109        157        149        216
## 10
## 58
## 15
## 82
## 91
## 11
## 9
## 57
## 49
## 116
## 110
## 158
## 115
## 182
## 191   317.0607
## 111   298.6713   358.5672
## 109   309.0362   288.3550   329.0478
## 157   328.9365   393.1221   269.5958   353.7915
## 149   325.9122   274.3806   367.1752   343.7351   389.4240
## 216   805.1517   858.6777   637.6681   836.5676   703.3033   757.1477
## 210   766.3368   806.0262   586.8305   745.9485   667.7633   707.0131   176.7453
```

```
## 258   587.3715   601.0278   545.1220   541.0853   550.0343   594.5751   238.0743
## 215   686.2869   735.4474   585.2514   646.3527   595.4377   711.3758   193.3705
## 282   587.5652   603.4359   526.4884   490.3294   514.4958   587.0745   267.0942
## 291   610.4768   660.6785   546.1680   576.3543   579.8801   630.4615   203.9899
## 211   547.3040   606.6419   521.6474   530.0703   543.0441   578.3284   317.4725
## 209   774.8533   815.6385   590.6338   760.7838   672.9532   728.0638   186.3797
## 257   642.7841   685.8969   525.6841   645.0844   592.5565   627.4209   187.4107
## 249   734.6865   770.1092   552.4348   726.2985   625.0091   664.9010   187.6744
## 316   701.2434   710.9590   864.6781   904.9058   841.1203   665.8931 2867.1367
## 310   649.3816   640.1871   823.4902   797.0925   811.9173   590.4126 2542.4639
## 358   668.7750   681.9107   881.6839   881.3672   829.6033   646.3389 2768.7468
## 315   578.4629   611.9814   755.3418   759.4720   728.0598   591.5309 2531.3495
## 382   615.3425   599.7273   754.5138   742.2208   735.9651   596.3133 2477.9494
## 391   726.3452   699.6076   875.3323   903.3866   826.5083   648.4218 2820.4087
## 311   679.2866   642.9511   841.7553   859.9165   780.5019   610.3545 2824.9890
## 309   463.3757   524.8751   600.9928   592.3491   639.8039   483.0316 1944.4676
## 357   460.7856   481.1587   573.2666   483.5174   603.3828   456.7659 1563.6178
## 349   414.5594   485.2992   542.1793   508.2119   539.3892   468.3561 1630.6319
## 416   484.8637   535.9122   522.4786   483.3615   508.2753   565.0660   367.7129
## 410   606.1491   661.6488   502.6564   564.3592   536.3398   581.6965   251.2541
## 458   570.5515   630.9203   557.2632   547.3573   563.1127   559.4955   302.5337
## 415   550.6911   629.4016   560.0213   522.6191   556.1025   640.5049   297.1813
## 482   545.5687   572.3504   541.3730   531.5401   519.5424   558.7746   360.6703
## 491   464.6655   516.0576   518.4839   441.5788   521.5269   496.9733   565.3128
## 411   438.1433   478.5134   505.8696   415.1446   503.7343   495.9307   452.2819
## 409   541.6940   590.0964   565.1622   533.1911   523.5025   599.1315   359.5486
## 457   624.7726   681.0840   501.0955   614.2271   561.9972   588.3617   285.5049
## 449   585.9369   624.4602   542.0483   572.8904   563.3500   592.9815   292.5477
## 516   449.6273   467.1835   554.3017   538.5000   555.3603   448.9096 1938.2141
## 510   430.8964   476.4940   512.4479   475.1514   530.7926   460.0181 1445.9496
## 558   397.0740   433.6814   487.3893   409.3047   518.4816   452.0862 1281.4766
## 515   413.1531   487.9575   527.9795   533.7974   593.5587   435.0354 1982.9833
## 582   508.1844   549.3079   603.2473   586.0947   620.4711   495.2484 2026.8166
## 591   451.2492   486.5355   567.1190   530.4514   599.0338   439.1375 1770.3262
## 511   427.4373   471.3820   519.3928   488.2188   560.0169   454.7943 1716.5551
## 509   439.0724   458.8486   558.7986   543.1909   549.8173   436.6915 1821.2556
## 557   420.6317   457.6400   494.2704   461.4041   536.4781   468.1237 1511.1746
## 549   558.2255   480.5482   659.5020   656.0190   591.5777   488.0520 2060.6935
##              210        258        215        282        291        211        209
## 10
## 58
## 15
## 82
## 91
## 11
## 9
## 57
## 49
## 116
## 110
## 158
## 115
## 182
## 191
## 111
## 109
## 157
## 149
## 216
## 210
## 258   233.4554
## 215   192.2041   191.6450
## 282   263.7349   165.5706   212.2624
## 291   196.5622   181.4513   183.6643   198.9183
```

```
## 211  280.7295  184.2833  210.6769  183.4964  185.2353
## 209  172.7391  196.5368  192.2933  251.2537  194.5812  272.9075
## 257  191.1697  196.4908  189.0324  182.5498  188.3735  210.8263  167.8796
## 249  203.0775  211.0436  195.9927  217.5148  205.3168  217.4082  182.3841
## 316 2662.1948 2095.6749 2230.3193 1909.5114 2011.4366 1780.1801 2748.0673
## 310 2469.0025 1957.4453 2083.2299 1759.0281 1881.6865 1677.6902 2488.1384
## 358 2593.2628 2025.1200 2180.1026 1864.3254 1954.6782 1729.9162 2653.0465
## 315 2434.4271 1984.9806 2083.7607 1794.4110 1870.9583 1676.3839 2489.4399
## 382 2405.8013 1917.3027 2041.5942 1715.0221 1854.5633 1652.1416 2450.7030
## 391 2646.6788 2128.3875 2289.7396 1945.8929 2070.8337 1847.1186 2715.6278
## 311 2670.9274 2111.5726 2251.4975 1920.0435 2052.2794 1817.8091 2713.5613
## 309 1874.2111 1568.8040 1681.8678 1403.3633 1524.1811 1364.7753 1909.7532
## 357 1487.7610 1209.7465 1307.7841 1094.5116 1212.3541 1085.1260 1503.4783
## 349 1602.1631 1297.5790 1399.3466 1210.7219 1353.7889 1227.6786 1617.8542
## 416  343.4806  221.6644  285.6033  214.0072  247.5515  206.0767  340.0201
## 410  274.5103  246.6985  250.7805  245.6002  239.6597  246.6074  249.1719
## 458  319.4254  277.0722  299.5206  275.0139  276.8185  281.5167  304.2282
## 415  296.0241  221.1846  237.8159  202.7355  226.2059  227.9480  261.0948
## 482  345.7744  241.9995  291.1835  231.4226  271.4426  250.4224  332.8638
## 491  482.6849  305.3460  369.3351  261.4331  320.2692  252.7834  499.5432
## 411  411.6199  253.7007  312.6596  233.0855  269.8167  224.6179  403.1346
## 409  340.5504  242.5912  254.4872  225.1278  258.6181  245.1796  325.4817
## 457  292.6711  291.7202  263.1180  289.2843  266.3026  276.2195  281.0331
## 449  304.1345  278.0441  280.7931  256.3285  261.5138  280.7864  297.2433
## 516 1810.1794 1378.6157 1478.4398 1232.1371 1321.1018 1162.2621 1843.0342
## 510 1385.5363 1110.1571 1210.3616 1005.5114 1084.3360  958.7676 1399.7819
## 558 1194.1854  957.8059 1030.6501  838.5865  934.9923  816.3089 1213.3024
## 515 1806.5010 1391.0757 1497.9142 1233.0408 1317.4276 1155.5595 1865.2255
## 582 1957.0384 1633.4936 1715.1654 1434.6928 1528.7747 1378.8262 1989.8740
## 591 1607.8478 1198.7075 1337.7079 1118.4929 1197.5484 1009.6076 1655.6850
## 511 1627.6880 1298.6707 1368.2469 1133.7947 1210.4423 1062.3666 1653.5981
## 509 1639.4492 1269.7960 1371.4627 1130.5483 1214.5430 1040.9591 1714.8136
## 557 1479.7426 1146.0047 1270.1629 1062.1124 1226.5762 1098.4799 1481.9519
## 549 1888.0496 1451.8259 1607.9668 1353.6647 1441.3101 1239.1352 1949.4793
##            257       249       316       310       358       315       382
## 10
## 58
## 15
## 82
## 91
## 11
## 9
## 57
## 49
## 116
## 110
## 158
## 115
## 182
## 191
## 111
## 109
## 157
## 149
## 216
## 210
## 258
## 215
## 282
## 291
## 211
## 209
## 257
## 249   186.6203
```

```
## 316 2401.7380 2612.3993
## 310 2247.9064 2370.5147  186.4061
## 358 2325.9322 2540.0938  169.4218  189.2728
## 315 2251.8321 2372.8624  185.5520  193.8454  190.3676
## 382 2225.4213 2309.8813  198.8966  201.1588  216.2667  180.9743
## 391 2426.4880 2583.1068  156.3046  174.6910  173.6105  195.3691  198.9028
## 311 2399.7135 2597.7707  187.8287  192.8147  176.6124  198.7769  208.5011
## 309 1719.8777 1806.0652  241.0633  204.0337  251.9073  211.2741  233.0564
## 357 1356.3945 1404.4864  367.9550  274.1337  352.4757  281.7031  268.8086
## 349 1462.4936 1513.5269  303.8218  272.1534  308.0630  224.8090  241.3437
## 416  255.7374  307.1740 1767.2225 1628.7882 1707.1044 1672.0597 1595.2665
## 410  236.4893  240.7972 1699.1602 1507.1122 1678.9767 1546.5208 1465.2624
## 458  274.6402  280.2280 2069.4549 1811.6985 1990.1648 1809.3364 1765.6519
## 415  237.6159  243.3973 1862.5001 1728.6196 1819.2683 1778.4143 1683.5221
## 482  288.6771  287.0789 1939.1338 1709.1882 1867.8551 1700.8454 1648.9072
## 491  389.4303  424.5677 1291.5400 1197.7365 1252.8502 1202.7189 1191.2611
## 411  298.0642  371.0819 1437.9718 1332.4802 1379.6890 1362.1360 1318.1805
## 409  262.3540  291.0509 1857.1196 1757.4864 1799.6455 1764.8963 1711.9882
## 457  265.1150  259.2030 2269.0156 2013.6411 2208.7066 2003.9452 1946.9169
## 449  260.7436  283.1933 2133.6876 1994.2284 2063.2260 1981.4581 1946.2202
## 516 1598.4254 1739.8372  336.9541  302.4338  333.1101  306.5733  300.3013
## 510 1251.6997 1304.5152  428.8973  360.0278  437.5025  375.8806  364.6204
## 558 1066.1918 1123.3298  459.5980  385.9800  467.2663  382.7908  370.0220
## 515 1598.6731 1768.6542  395.9807  338.3058  365.1944  304.5471  336.3361
## 582 1812.3080 1873.1539  338.7678  318.0229  345.8794  288.0557  271.3779
## 591 1426.0531 1567.0481  355.0474  309.7662  340.7427  283.4909  283.3660
## 511 1476.4259 1560.8842  360.3773  338.2268  367.6958  293.2159  301.0897
## 509 1466.9181 1609.6692  324.4792  307.9878  313.7574  286.1064  285.4719
## 557 1348.9734 1346.4313  439.5722  365.1693  456.9742  358.7957  338.4169
## 549 1708.1607 1837.5243  354.3086  343.4559  360.5399  339.4429  355.7887
##              391       311       309       357       349       416       410
## 10
## 58
## 15
## 82
## 91
## 11
## 9
## 57
## 49
## 116
## 110
## 158
## 115
## 182
## 191
## 111
## 109
## 157
## 149
## 216
## 210
## 258
## 215
## 282
## 291
## 211
## 209
## 257
## 249
## 316
## 310
## 358
## 315
```

```
## 382
## 391
## 311  180.9067
## 309  239.2966  225.1823
## 357  311.4097  360.8554  193.2415
## 349  298.7071  300.5772  184.4269  179.8640
## 416 1773.9031 1769.6936 1264.6615  956.4412 1057.2574
## 410 1698.7998 1709.4247 1153.4108  886.5442  968.8306  238.8364
## 458 2018.9767 2037.7941 1367.4596 1075.1067 1147.1902  228.4807  184.2529
## 415 1885.2824 1877.0779 1344.9249 1023.7177 1107.1961  183.9756  224.5038
## 482 1884.5055 1919.4286 1260.2230  969.7392 1063.4223  184.9524  196.6786
## 491 1360.6643 1324.9295  899.3018  734.9139  869.5993  282.6441  369.2314
## 411 1471.9809 1468.8738 1038.2352  791.4835  891.0855  191.7580  294.8490
## 409 1907.7182 1892.3444 1378.5712 1082.5053 1157.5906  179.9284  200.7810
## 457 2262.8463 2247.7302 1527.5828 1202.6683 1256.8377  263.4201  171.9886
## 449 2137.2675 2144.1763 1502.0909 1188.0568 1261.5096  223.7616  184.8163
## 516  321.5376  321.9363  256.5804  295.0870  251.9803 1156.5693 1099.2251
## 510  417.1986  427.9620  233.9889  222.8659  217.9916  889.6169  810.2666
## 558  423.0234  455.0035  230.9871  184.8549  213.5916  738.7346  686.6547
## 515  359.6968  369.3802  260.9424  276.4925  250.0588 1137.8400 1129.6823
## 582  327.8226  314.1053  268.8427  314.6790  260.6628 1357.6906 1225.6823
## 591  325.0655  332.8270  216.6676  236.5685  233.1903  976.7311 1108.2934
## 511  362.0831  364.4719  228.5629  245.0669  200.1102 1054.6648  994.4526
## 509  317.1559  299.2519  236.6151  260.2224  236.6937 1022.6318 1053.5476
## 557  411.7381  436.5378  252.2899  209.3507  205.1107  914.8850  892.4761
## 549  346.4923  329.0408  353.1266  414.8373  367.8390 1227.0337 1289.0398
##            458       415       482       491       411       409       457
## 10
## 58
## 15
## 82
## 91
## 11
## 9
## 57
## 49
## 116
## 110
## 158
## 115
## 182
## 191
## 111
## 109
## 157
## 149
## 216
## 210
## 258
## 215
## 282
## 291
## 211
## 209
## 257
## 249
## 316
## 310
## 358
## 315
## 382
## 391
## 311
## 309
```

```
## 357
## 349
## 416
## 410
## 458
## 415  223.9141
## 482  195.4737  188.4165
## 491  350.5414  314.5157  319.6894
## 411  297.3975  208.5018  252.4904  199.8641
## 409  205.6861  165.9211  199.2893  308.4925  228.6487
## 457  226.6942  249.2783  214.9593  430.3002  332.6068  225.8847
## 449  180.0448  196.3687  175.9837  368.5786  290.1385  193.5615  199.1473
## 516 1365.7275 1217.7682 1258.2175  834.5801  907.7636 1233.8374 1498.0758
## 510  993.3991  951.9124  899.7473  669.4285  720.0009 1003.1836 1105.4048
## 558  890.2236  807.9556  774.5303  521.9306  572.4532  870.6864  963.7964
## 515 1396.7561 1210.4896 1273.0072  811.4351  879.1442 1231.5203 1537.3679
## 582 1446.2340 1427.7705 1331.3975  989.7157 1103.1860 1469.4486 1557.7589
## 591 1210.7005 1074.8905 1084.8345  718.2944  752.1709 1088.4293 1420.1635
## 511 1189.3564 1122.1569 1079.5742  751.2923  801.3940 1153.2178 1311.7027
## 509 1271.6178 1124.0013 1139.8045  747.4206  799.8171 1142.0112 1412.8123
## 557 1042.0080  999.4399  949.7769  771.7427  765.5957 1037.8850 1135.6725
## 549 1467.8950 1316.7982 1342.4728  918.3700  984.8253 1329.5944 1661.4419
##                 449       516       510       558       515       582       591
## 10
## 58
## 15
## 82
## 91
## 11
## 9
## 57
## 49
## 116
## 110
## 158
## 115
## 182
## 191
## 111
## 109
## 157
## 149
## 216
## 210
## 258
## 215
## 282
## 291
## 211
## 209
## 257
## 249
## 316
## 310
## 358
## 315
## 382
## 391
## 311
## 309
## 357
## 349
## 416
## 410
```

```
## 458
## 415
## 482
## 491
## 411
## 409
## 457
## 449
## 516 1443.2426
## 510 1094.6853   239.8513
## 558  969.2235   295.3539   214.8268
## 515 1427.7966   184.9936   220.9261   259.4805
## 582 1576.2604   178.0676   282.8236   322.0392   186.9727
## 591 1261.3511   191.9173   244.3763   254.0292   197.7058   214.1429
## 511 1290.9843   209.9871   209.5292   237.6061   182.6784   214.3165   182.7114
## 509 1309.5127   185.7424   212.0275   257.4121   181.2420   207.8011   203.4438
## 557 1137.2228   268.4579   196.5879   211.1711   256.3025   281.0986   255.7545
## 549 1532.0960   211.7532   347.3877   392.3697   244.3941   250.8419   234.8534
##              511        509        557
## 10
## 58
## 15
## 82
## 91
## 11
## 9
## 57
## 49
## 116
## 110
## 158
## 115
## 182
## 191
## 111
## 109
## 157
## 149
## 216
## 210
## 258
## 215
## 282
## 291
## 211
## 209
## 257
## 249
## 316
## 310
## 358
## 315
## 382
## 391
## 311
## 309
## 357
## 349
## 416
## 410
## 458
## 415
## 482
## 491
```
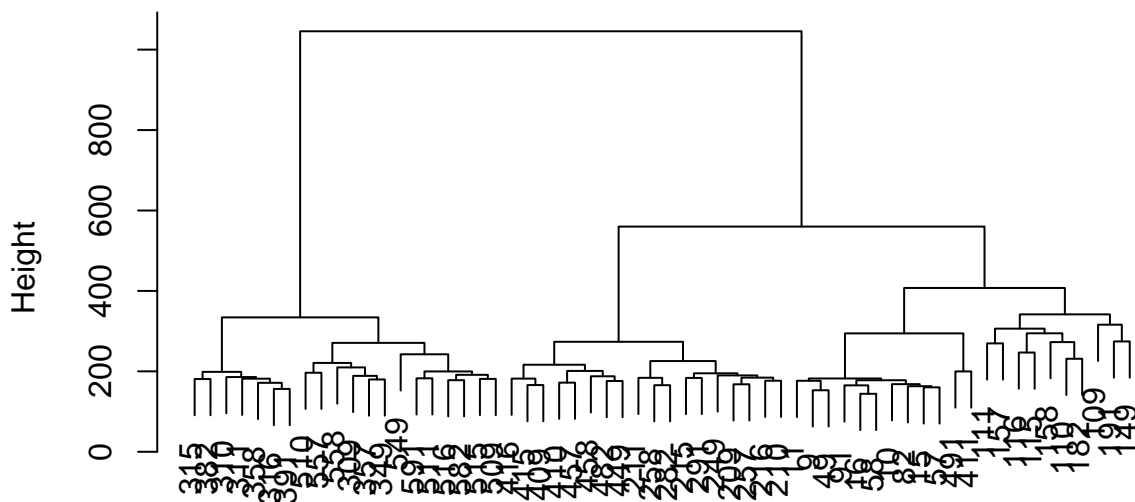
```
## 411
## 409
## 457
## 449
## 516
## 510
## 558
## 515
## 582
## 591
## 511
## 509   184.0914
## 557   208.9702   266.4586
## 549   284.5245   228.3851   399.2930
```

```r
hclust(dist(sample2, method = 'DTW'), method = 'average')
```

```
##
## Call:
## hclust(d = dist(sample2, method = "DTW"), method = "average")
##
## Cluster method   : average
## Distance         : DTW
## Number of objects: 60
```

```r
plot(hclust(dist(sample2, method = 'DTW'), method = 'average'))
```


Cluster Dendrogram

dist(sample2, method = "DTW")
hclust (*, "average")

4. Time Series Classification

Exercise

1. Import economic_data.csv ke dalam R.
2. Takrifkan data kepada format siri masa iaitu ianya adalah data bulanan bermula Januari 2000.

3. Plotkan siri masa tersebut.
4. Kenalpasti dan suaikan model ARIMA yang sesuai terhadap data.
5. Jalan peramalan terhadap data untuk 24 bulan seterusnya.
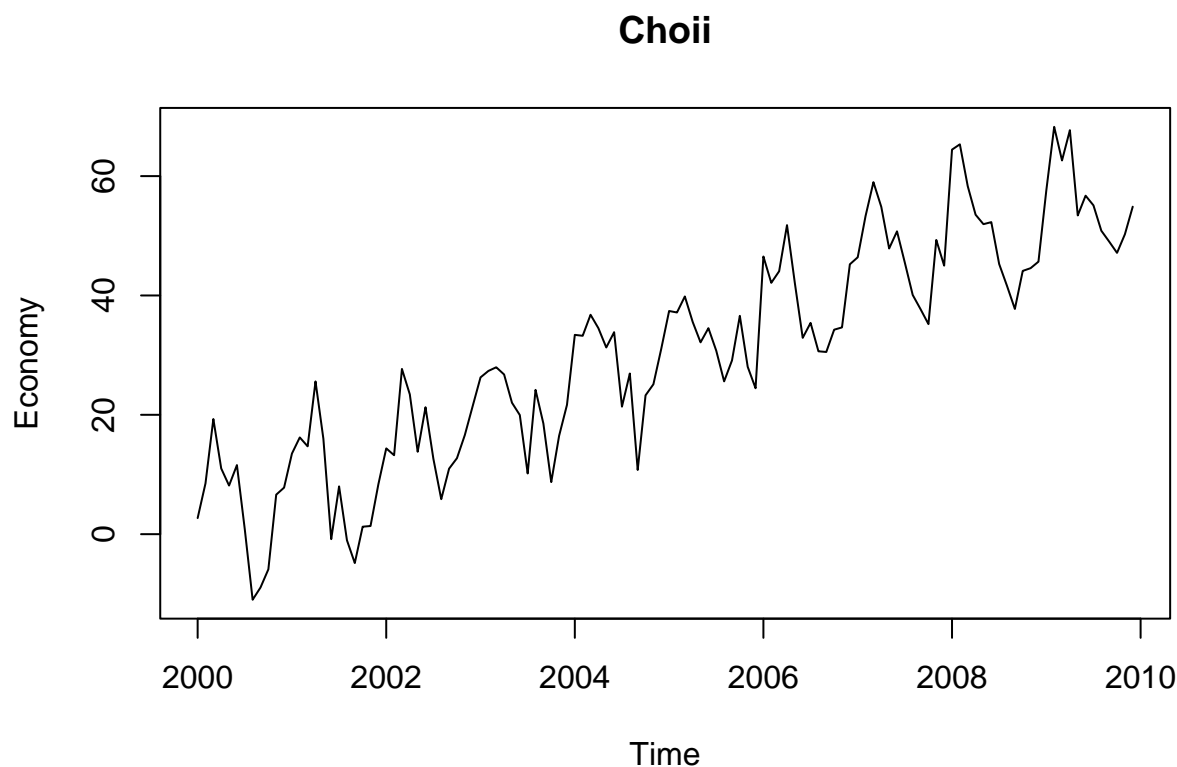6. Plotkan peramalan bersama selang keyakinan.

```r
eco_data <- read.csv('economic_data.csv', sep = ';')
head(eco_data)
```

```
##   Time Economic_Data.x
## 1    1        2.697622
## 2    2        8.509367
## 3    3       19.293542
## 4    4       11.012796
## 5    5        8.146439
## 6    6       11.575325
```

```r
eco_ts <- ts(eco_data[,2], start = c(2000,1), frequency = 12)
head(eco_ts)
```

```
##             Jan       Feb       Mar       Apr       May       Jun
## 2000   2.697622  8.509367 19.293542 11.012796  8.146439 11.575325
```

```r
plot.ts(eco_ts, main = 'Choii', xlab = 'Time', ylab = 'Economy')
```


Choii

```r
eco_model <- auto.arima(eco_ts)
summary(eco_model)
```

```
## Series: eco_ts
## ARIMA(0,0,0)(0,1,2)[12] with drift
##
## Coefficients:
##          sma1    sma2   drift
##       -1.0488  0.1765  0.4944
```

```
## s.e.    0.1606  0.1148  0.0109
##
## sigma^2 = 21.76:  log likelihood = -327.72
## AIC=663.44    AICc=663.82    BIC=674.16
##
## Training set error measures:
##                        ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.01724821 4.363728 3.269357 5.841319 32.55449 0.4613457
##                      ACF1
## Training set 0.04087961
```

```
fcst <- forecast(eco_model)
fcst
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Jan 2010       67.94274 61.93164 73.95384 58.74956 77.13592
## Feb 2010       68.73219 62.72109 74.74329 59.53900 77.92537
## Mar 2010       71.89923 65.88814 77.91033 62.70605 81.09242
## Apr 2010       70.53671 64.52561 76.54780 61.34352 79.72989
## May 2010       65.11032 59.09922 71.12142 55.91714 74.30350
## Jun 2010       64.04317 58.03208 70.05427 54.84999 73.23636
## Jul 2010       58.94314 52.93204 64.95424 49.74996 68.13632
## Aug 2010       56.22022 50.20913 62.23132 47.02704 65.41341
## Sep 2010       53.67621 47.66511 59.68731 44.48303 62.86939
## Oct 2010       57.05715 51.04605 63.06825 47.86397 66.25033
## Nov 2010       60.44689 54.43579 66.45799 51.25371 69.64007
## Dec 2010       63.16441 57.15332 69.17551 53.97123 72.35760
## Jan 2011       73.41010 67.40418 79.41602 64.22484 82.59536
## Feb 2011       75.87952 69.87360 81.88544 66.69426 85.06478
## Mar 2011       77.24475 71.23883 83.25067 68.05949 86.43001
## Apr 2011       76.82497 70.81906 82.83089 67.63971 86.01023
## May 2011       70.04293 64.03701 76.04885 60.85767 79.22819
## Jun 2011       69.77046 63.76454 75.77638 60.58520 78.95572
## Jul 2011       65.19121 59.18529 71.19712 56.00595 74.37647
## Aug 2011       62.15852 56.15260 68.16444 52.97326 71.34378
## Sep 2011       59.69520 53.68928 65.70112 50.50994 68.88046
## Oct 2011       62.26348 56.25756 68.26940 53.07822 71.44874
## Nov 2011       65.57543 59.56951 71.58135 56.39017 74.76069
## Dec 2011       68.51200 62.50608 74.51792 59.32674 77.69726
```

```
is.null(eco_model)
```

```
## [1] FALSE
```

```
#eco_U <- fcst$pred+0.69*fcst$se
#eco_L <- fcst$pred-0.69*fcst$se
```

```
#plot(eco_ts, fcst$mean)
```