

Latihan Pertemuan 7 : Diagnostik Model & Peramalan - Adinda Shabrina Putri Salsabila

Packages

```
library(ggplot2)
library(tsibble)
library(tseries)
library(MASS)
library(forecast)
library(TSA)
library(TTR)
library(aTSA)
library(graphics)
```

Data

Digunakan data harga cabai rawit di wilayah Jakarta yang dalam hal ini hanya digunakan data 140 periode akhir

```
data <- read.csv("C:/Users/hp/Documents/datampdw6.csv")
datacabai.ts <- ts(data)
head(datacabai.ts)

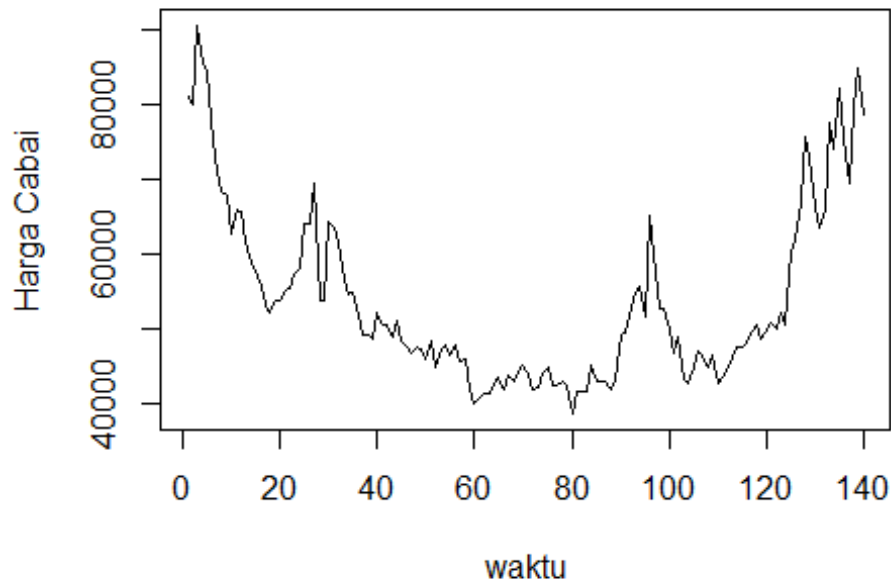
## Time Series:
## Start = 1
## End = 6
## Frequency = 1
##      harga
## [1,] 81090
## [2,] 80000
## [3,] 90670
## [4,] 85730
## [5,] 84340
## [6,] 75040
```

Eksplorasi Data

Plot Data Penuh

```
plot.ts(datacabai.ts, lty=1, xlab="waktu", ylab="Harga Cabai", main="Plot  
Data Harga Cabai")
```

Plot Data Harga Cabai

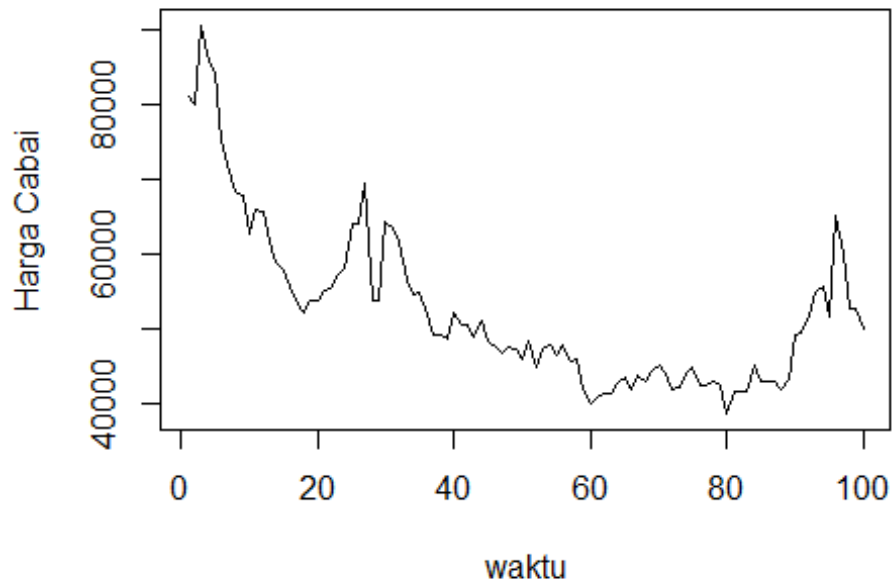


Berdasarkan plot data deret waktu, terlihat adanya tren jangka panjang yang fluktuatif. Data dimulai dengan harga tinggi, kemudian menurun secara umum hingga sekitar titik data ke-80, lalu mulai menunjukkan tren kenaikan kembali hingga akhir periode. Juga adanya indikasi pola siklis, ditunjukkan oleh fluktuasi yang berulang dalam jangka waktu yang lebih panjang. Terlihat beberapa siklus naik-turun sepanjang periode waktu yang ditampilkan. Berdasarkan pola data, pembagian data latih dan data uji ditetapkan dengan proporsi 71%:29%.

Plot Data Latih

```
cabaitrain<-datacabai.ts[1:100]
train.ts<-ts(cabaitrain)
plot.ts(train.ts, lty=1, xlab="waktu", ylab="Harga Cabai", main="Plot Harga Cabai Train")
```

Plot Harga Cabai Train

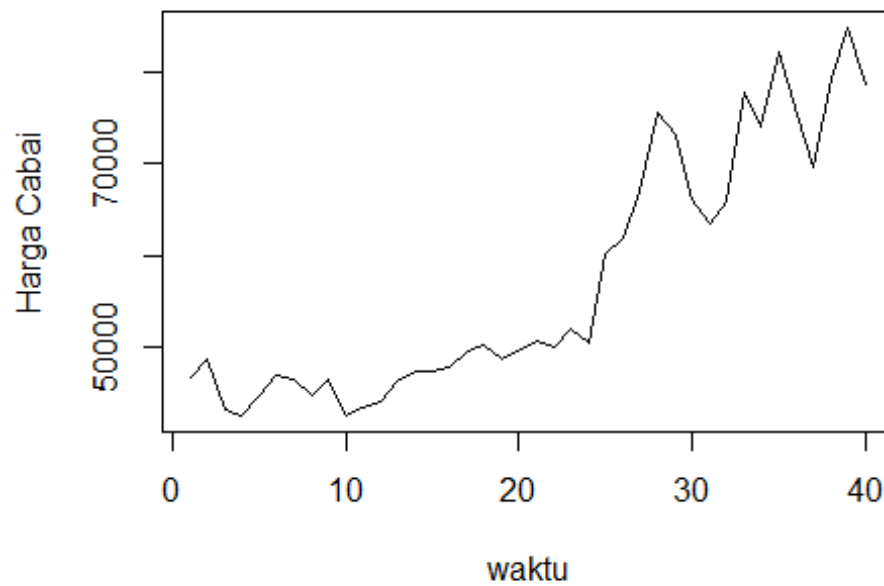


Berdasarkan plot data deret waktu pada data latih, terlihat bahwa data cenderung memiliki trend yang naik-turun dan cenderung tidak bergerak pada nilai tengah tertentu. Hal ini mengindikasikan bahwa data tidak stasioner dalam rata-rata.

Plot Data Uji

```
cabaitest<-datacabai.ts[101:140]  
test.ts<-ts(cabaitest)  
plot.ts(test.ts, lty=1, xlab="waktu", ylab="Harga Cabai", main="Plot Harga  
Cabai Test")
```

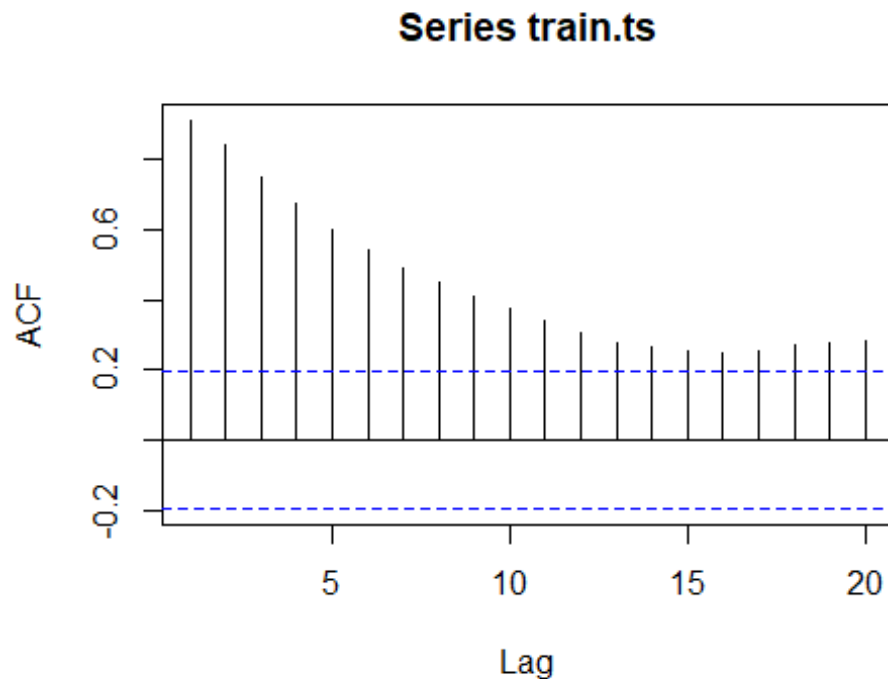
Plot Harga Cabai Test



Uji Stasioneritas Data

Plot ACF

```
acf(train.ts)
```



Berdasarkan plot ACF, terlihat bahwa plot ACF data menurun secara perlahan (*tails of slowly*). Hal ini juga menjadi indikasi bahwa data tidak stasioner dalam rataa

Uji ADF

```
tseries::adf.test(train.ts)

##
## Augmented Dickey-Fuller Test
##
## data: train.ts
## Dickey-Fuller = -3.1755, Lag order = 4, p-value = 0.09586
## alternative hypothesis: stationary
```

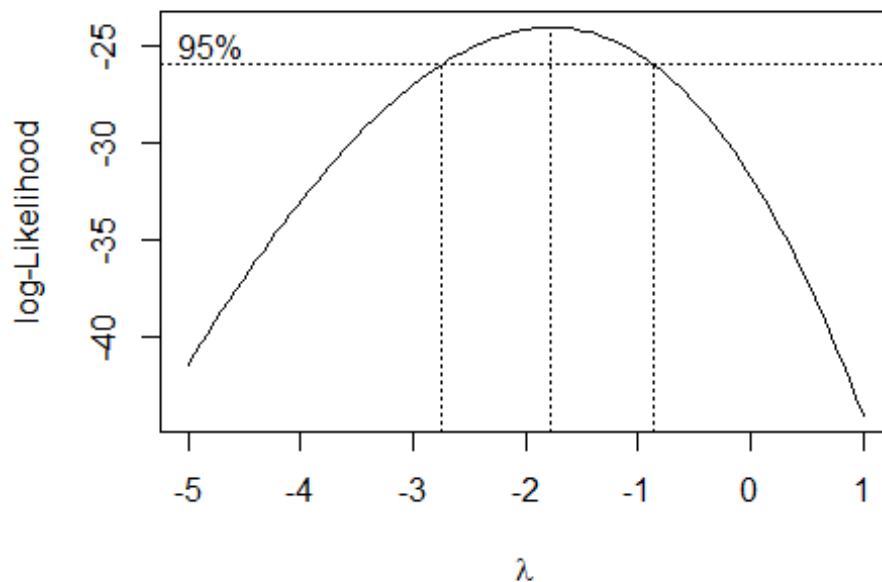
H_0 : Data tidak stasioner dalam rataa

H_1 : Data stasioner dalam rataa

Berdasarkan uji ADF tersebut, didapat *p-value* sebesar 0.09586 yang lebih besar dari taraf nyata 5% sehingga tak tolak H_0 dan menandakan bahwa data tidak stasioner dalam rataa. Hal ini sesuai dengan hasil eksplorasi menggunakan plot time series dan plot ACF, sehingga ketidakstasioneran model kedepannya harus ditangani

Plot Box-Cox

```
index <- seq(1:100)
bc = boxcox(train.ts~index, lambda = seq(-5,by=1))
```



```
#Nilai Rounded Lambda
lambda <- bc$x[which.max(bc$y)]
lambda

## [1] -1.787879

#SK
bc$x[bc$y > max(bc$y) - 1/2 * qchisq(.95,1)]

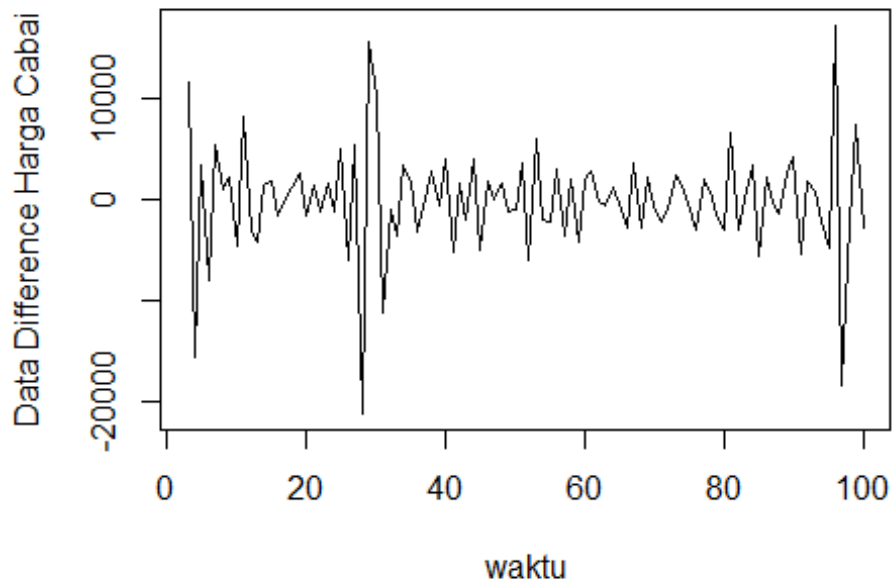
## [1] -2.6969697 -2.6363636 -2.5757576 -2.5151515 -2.4545455 -2.3939394
## [7] -2.3333333 -2.2727273 -2.2121212 -2.1515152 -2.0909091 -2.0303030
## [13] -1.9696970 -1.9090909 -1.8484848 -1.7878788 -1.7272727 -1.6666667
## [19] -1.6060606 -1.5454545 -1.4848485 -1.4242424 -1.3636364 -1.3030303
## [25] -1.2424242 -1.1818182 -1.1212121 -1.0606061 -1.0000000 -0.9393939
## [31] -0.8787879
```

Plot Boxcox menunjukkan nilai *rounded value* (λ) optimum sebesar **-1.787879** dan pada selang kepercayaan 95% nilai memiliki batas bawah **-2.6969697** dan batas atas **-0.8787879**. Selang tersebut tidak memuat nilai satu sehingga dapat dikatakan bahwa data tidak stasioner dalam ragam.

Penanganan Ketidakstasioneran Data

```
train.diff<-diff(train.ts,differences = 2)
plot.ts(train.diff, lty=1, xlab="waktu", ylab="Data Difference Harga Cabai",
main="Plot Difference Harga Cabai")
```

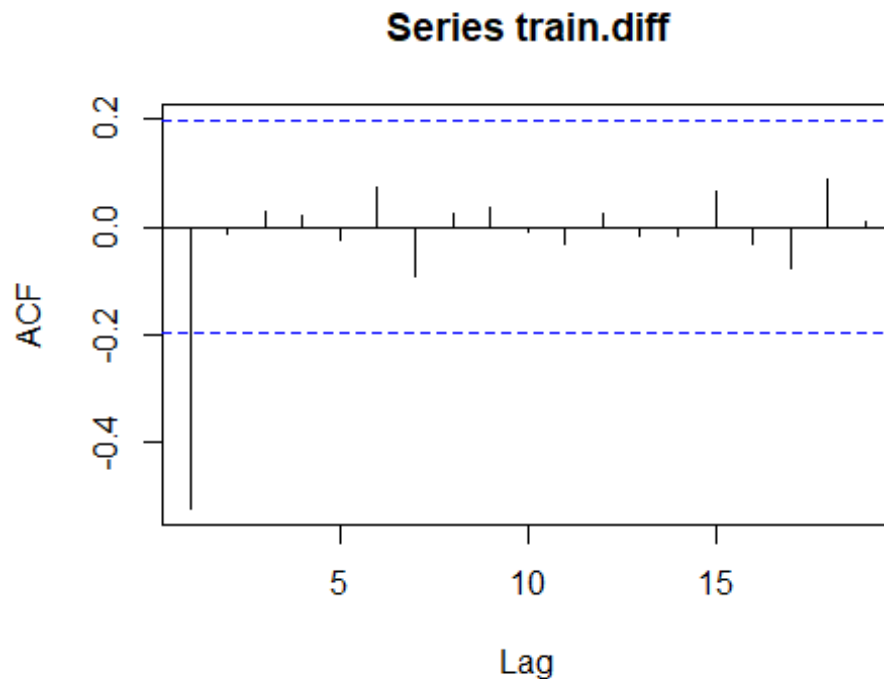
Plot Difference Harga Cabai



Berdasarkan plot data deret waktu, terlihat bahwa data sudah stasioner dalam rata-rata ditandai dengan data bergerak pada nilai tengah tertentu (tidak terdapat trend ataupun musiman pada data)

Plot ACF

```
acf(train.diff)
```



Berdasarkan plot tersebut, terlihat bahwa plot ACF cuts off pada lag ke 1. Hal ini menandakan data sudah stasioner dalam rata-rata dan ketidakstasioneran data telah berhasil tertangani.

Uji ADF

```
tseries::adf.test(train.diff)

## Warning in tseries::adf.test(train.diff): p-value smaller than printed p-
value

##
## Augmented Dickey-Fuller Test
##
## data: train.diff
## Dickey-Fuller = -8.4919, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

H_0 : Data tidak stasioner dalam rata-rata

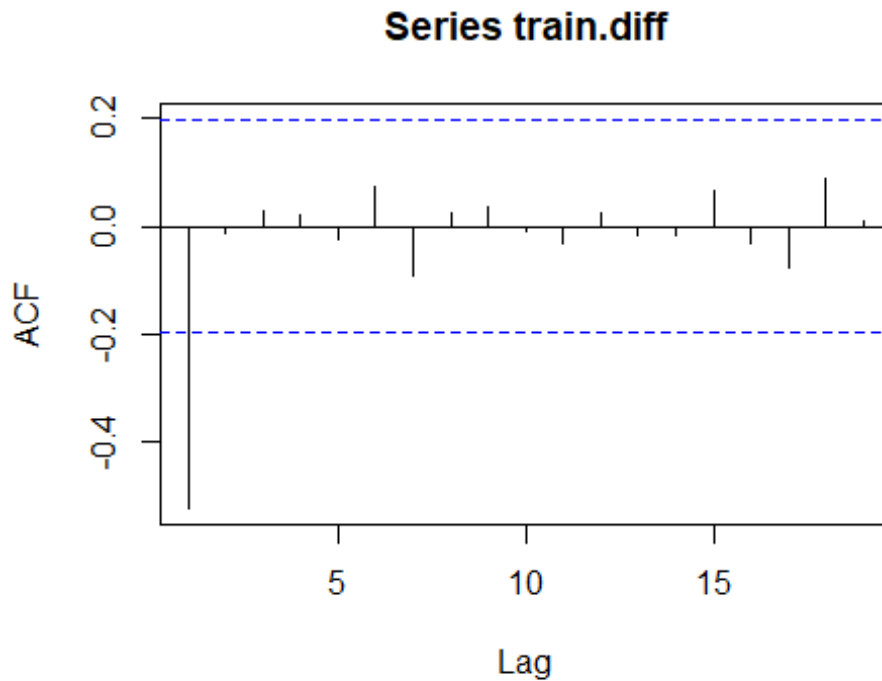
H_1 : Data stasioner dalam rata-rata

Berdasarkan uji ADF tersebut, didapat *p-value* sebesar 0.01 yang lebih kecil dari taraf nyata 5% sehingga tolak H_0 atau data stasioner dalam rata-rata. Hal ini sesuai dengan hasil eksplorasi menggunakan plot time series dan plot ACF, sehingga dalam hal ini ketidakstasioneran data sudah berhasil ditangani dan dapat dilanjutkan ke pemodelan

Identifikasi Model

Plot ACF

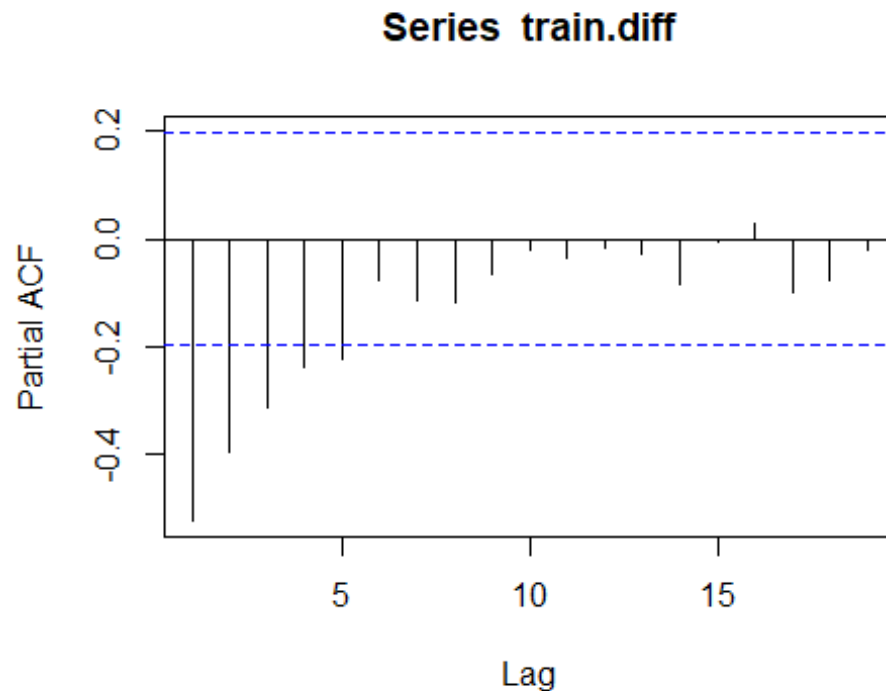
```
acf(train.diff)
```



Berdasarkan plot tersebut, terlihat bahwa plot ACF cenderung *cuts off* pada lag ke 1, sehingga jika plot PACF dianggap *tails of*, maka model tentatifnya adalah ARIMA(0,2,1)

Plot PACF

```
pacf(train.diff)
```



Berdasarkan plot tersebut, terlihat bahwa plot PACF cenderung *cuts off* pada lag ke 5, sehingga jika plot ACF dianggap *tails of*, maka model tentatifnya adalah ARIMA(5,2,0).

Jika baik plot ACF maupun plot PACF keduanya dianggap *tails of*, maka model yang terbentuk adalah ARIMA(5,2,1)

Plot EACF

```
eacf(train.diff)
```

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

Identifikasi model menggunakan plot EACF dilakukan dengan melihat ujung segitiga pada pola segitiga nol. Dalam hal ini model tentatif yang terbentuk adalah ARIMA(0,2,1), ARIMA(0,2,2), ARIMA(1,2,2), ARIMA(2,2,3), ARIMA(4,2,1)

Pendugaan Parameter Model Tentatif

ARIMA(0,2,1)

```
model1.da=Arima(train.diff, order=c(0,0,1),method="ML")
summary(model1.da) #AIC=1893.61

## Series: train.diff
## ARIMA(0,0,1) with non-zero mean
##
## Coefficients:
##          ma1      mean
##        -1.0000  16.1206
## s.e.    0.0265  12.6682
##
## sigma^2 = 13219384: log likelihood = -943.81
## AIC=1893.61  AICc=1893.87  BIC=1901.37
##
## Training set error measures:
##              ME      RMSE      MAE  MPE  MAPE      MASE      ACF1
## Training set -30.86232 3598.555 2373.288 -Inf  Inf  0.364166 -0.1434246

lmtest::coefTest(model1.da) #seluruh parameter signifikan

##
## z test of coefficients:
##
##          Estimate Std. Error z value Pr(>|z|)
## ma1          -1.00000    0.02646 -37.7932  <2e-16 ***
## intercept  16.12064    12.66817   1.2725   0.2032
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ARIMA(5,2,0)

```
model2.da=Arima(train.diff, order=c(5,0,0),method="ML")
summary(model2.da) #AIC=1908.64

## Series: train.diff
## ARIMA(5,0,0) with non-zero mean
##
## Coefficients:
##          ar1      ar2      ar3      ar4      ar5      mean
##        -1.0057 -0.9562 -0.7802 -0.5505 -0.2824 -11.6436
## s.e.    0.0989  0.1430  0.1618  0.1582  0.1192  85.6004
##
## sigma^2 = 15261584: log likelihood = -947.32
## AIC=1908.64  AICc=1909.89  BIC=1926.74
##
## Training set error measures:
##              ME      RMSE      MAE  MPE  MAPE      MASE      ACF1
## Training set -58.56193 3785.129 2622.455 Inf  Inf  0.402399 -0.007891049
```

```
lmtest::coeftest(model2.da) #seluruh parameter signifikan
```

```
##
## z test of coefficients:
##
##           Estimate Std. Error  z value  Pr(>|z|)
## ar1         -1.005687    0.098853 -10.1736 < 2.2e-16 ***
## ar2         -0.956161    0.143007  -6.6861 2.291e-11 ***
## ar3         -0.780199    0.161814  -4.8216 1.424e-06 ***
## ar4         -0.550459    0.158162  -3.4803 0.0005008 ***
## ar5         -0.282426    0.119166  -2.3700 0.0177874 *
## intercept -11.643643   85.600405  -0.1360 0.8918029
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ARIMA(5,2,1)

```
model3.da=Arima(train.diff, order=c(5,0,1),method="ML")
```

```
summary(model3.da) #AIC=1899.01
```

```
## Series: train.diff
## ARIMA(5,0,1) with non-zero mean
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5      ma1      mean
##          -0.1933  -0.1651  -0.0943  -0.0639  0.0102  -1.000  18.8908
## s.e.      0.1026   0.1107   0.1160   0.1169   0.1240   0.029   8.5315
##
## sigma^2 = 13183762: log likelihood = -941.51
## AIC=1899.01  AICc=1900.63  BIC=1919.69
##
## Training set error measures:
##           ME      RMSE      MAE  MPE MAPE      MASE      ACF1
## Training set 4.992139 3498.866 2416.909 -Inf  Inf  0.3708592 0.005667689
```

```
lmtest::coeftest(model3.da) #terdapat parameter tidak signifikan
```

```
##
## z test of coefficients:
##
##           Estimate Std. Error  z value  Pr(>|z|)
## ar1         -0.193255    0.102628  -1.8831 0.05969 .
## ar2         -0.165103    0.110728  -1.4911 0.13594
## ar3         -0.094303    0.115999  -0.8130 0.41624
## ar4         -0.063925    0.116907  -0.5468 0.58452
## ar5          0.010216    0.123967   0.0824 0.93432
## ma1         -0.999986    0.028964 -34.5250 < 2e-16 ***
## intercept 18.890834    8.531463   2.2143 0.02681 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ARIMA(0,2,2)

```
model4.da=Arima(train.diff, order=c(0,0,2),method="ML")
summary(model4.da) #AIC=1892.73

## Series: train.diff
## ARIMA(0,0,2) with non-zero mean
##
## Coefficients:
##          ma1      ma2      mean
##      -1.1994  0.1994  16.8496
## s.e.   0.1201  0.1170  10.0711
##
## sigma^2 = 12907690: log likelihood = -942.36
## AIC=1892.73  AICc=1893.16  BIC=1903.07
##
## Training set error measures:
##              ME      RMSE      MAE  MPE MAPE      MASE      ACF1
## Training set -43.16505 3537.309 2437.761 -Inf  Inf  0.3740588 0.03292367

lmtest::coeftest(model4.da) #seluruh parameter signifikan

##
## z test of coefficients:
##
##      Estimate Std. Error z value Pr(>|z|)
## ma1      -1.19939    0.12007  -9.9891  < 2e-16 ***
## ma2       0.19940    0.11700   1.7043  0.08833 .
## intercept 16.84957   10.07107   1.6731  0.09431 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ARIMA(1,2,2)

```
model5.da=Arima(train.diff, order=c(1,0,2),method="ML")
summary(model5.da) #AIC=1890.3

## Series: train.diff
## ARIMA(1,0,2) with non-zero mean
##
## Coefficients:
##          ar1      ma1      ma2      mean
##      0.7783 -1.9989  1.0000  16.3161
## s.e.   0.0765  0.0595  0.0595   3.0042
##
## sigma^2 = 11892664: log likelihood = -940.15
## AIC=1890.3  AICc=1890.96  BIC=1903.23
##
## Training set error measures:
##              ME      RMSE      MAE  MPE MAPE      MASE      ACF1
## Training set 81.33425 3377.462 2283.513 -Inf  Inf  0.3503905 -0.03784076
```

```
lmtest::coeftest(model5.da) #seluruh parameter signifikan
```

```
##
## z test of coefficients:
##
##           Estimate Std. Error  z value  Pr(>|z|)
## ar1           0.778303   0.076545  10.1679 < 2.2e-16 ***
## ma1          -1.998881   0.059523 -33.5815 < 2.2e-16 ***
## ma2           0.999957   0.059537  16.7957 < 2.2e-16 ***
## intercept 16.316128   3.004177   5.4311 5.599e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ARIMA(4,2,1)

```
model6.da=Arima(train.diff, order=c(4,0,1),method="ML")
summary(model6.da) #AIC=1897.02
```

```
## Series: train.diff
## ARIMA(4,0,1) with non-zero mean
##
## Coefficients:
##           ar1          ar2          ar3          ar4          ma1          mean
##          -0.1933   -0.1642   -0.0949   -0.0660   -1.000   18.9301
## s.e.    0.1026    0.1107    0.1148    0.1147    0.029    8.4628
##
## sigma^2 = 13039218: log likelihood = -941.51
## AIC=1897.02  AICc=1898.26  BIC=1915.11
##
## Training set error measures:
##              ME      RMSE      MAE  MPE  MAPE      MASE      ACF1
## Training set 5.188913 3498.699 2414.611 -Inf  Inf  0.3705067 0.006381861
```

```
lmtest::coeftest(model6.da) #terdapat parameter tak signifikan
```

```
##
## z test of coefficients:
##
##           Estimate Std. Error  z value  Pr(>|z|)
## ar1          -0.193303   0.102623  -1.8836  0.05961 .
## ar2          -0.164153   0.110714  -1.4827  0.13816
## ar3          -0.094906   0.114764  -0.8270  0.40826
## ar4          -0.066019   0.114668  -0.5757  0.56479
## ma1          -0.999997   0.029023 -34.4555 < 2e-16 ***
## intercept 18.930097   8.462832   2.2369  0.02530 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Penentuan Model Terbaik

```
perbandingan <- matrix(c("1893.61", "semua_signifikan",
                        "1908.64", "semua_signifikan",
                        "1899.01", "4_parameter_tidak_signifikan",
                        "1892.73", "semua_signifikan",
                        "1890.3", "Semua_signifikan",
                        "1897.02", "3_parameter_tidak_signifikan"), nrow=6, ncol=2, byrow = T)
colnames(perbandingan) <- c("AIC", "Parameter")
row.names(perbandingan) <-
c("ARIMA(0,2,1)", "ARIMA(5,2,0)", "ARIMA(5,2,1)", "ARIMA(0,2,2)", "ARIMA(1,2,2)",
  "ARIMA(4,2,1)")
perbandingan

##           AIC      Parameter
## ARIMA(0,2,1) "1893.61" "semua_signifikan"
## ARIMA(5,2,0) "1908.64" "semua_signifikan"
## ARIMA(5,2,1) "1899.01" "4_parameter_tidak_signifikan"
## ARIMA(0,2,2) "1892.73" "semua_signifikan"
## ARIMA(1,2,2) "1890.3"  "Semua_signifikan"
## ARIMA(4,2,1) "1897.02" "3_parameter_tidak_signifikan"
```

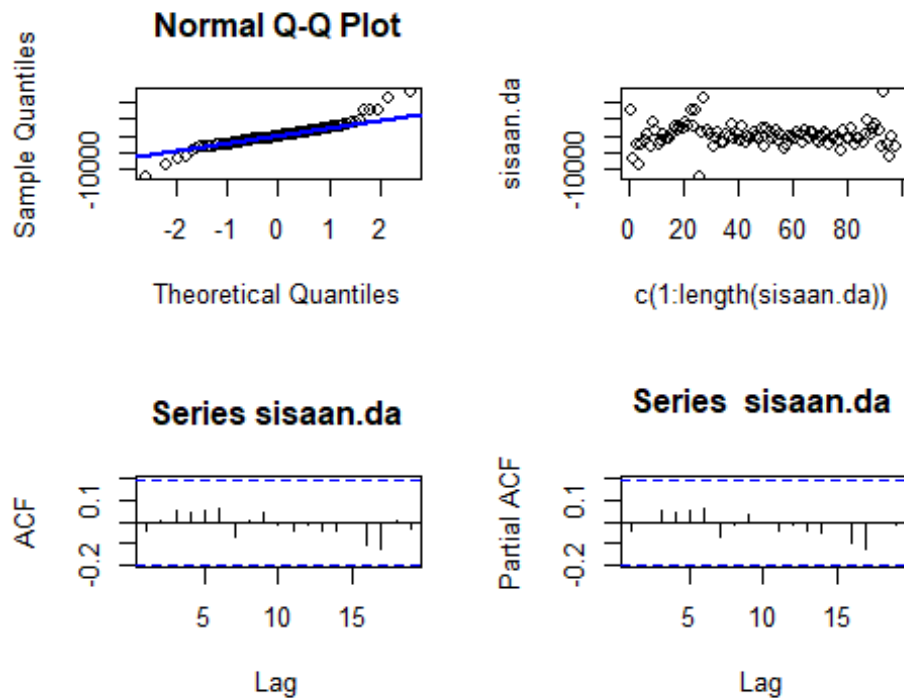
Berdasarkan pendugaan parameter di atas dengan differencing sebanyak dua kali, nilai AIC terkecil dimiliki oleh model ARIMA(1,2,2) dan parameter model ARIMA(1,2,2) juga seluruhnya signifikan sehingga model yang dipilih adalah model ARIMA(1,2,2).

Analisis Sisaan

Model terbaik hasil identifikasi kemudian dicek asumsi sisaannya. Sisaan model ARIMA harus memenuhi asumsi normalitas, kebebasan sisaan, dan kehomogenan ragam. Diagnostik model dilakukan secara eksplorasi dan uji formal.

Eksplorasi Sisaan

```
#Eksplorasi
sisaan.da <- model5.da$residuals
par(mfrow=c(2,2))
qqnorm(sisaan.da)
qqline(sisaan.da, col = "blue", lwd = 2)
plot(c(1:length(sisaan.da)),sisaan.da)
acf(sisaan.da)
pacf(sisaan.da)
```



```
par(mfrow = c(1,1))
```

Berdasarkan plot kuantil-kuantil normal, secara eksplorasi ditunjukkan sisaan tidak menyebar normal ditandai dengan titik titik yang cenderung tidak mengikuti garis 45°. Kemudian dapat dilihat juga lebar pita sisaan yang cenderung tidak sama menandakan bahwa sisaan memiliki ragam yang heterogen. Plot ACF dan PACF sisaan ARIMA(1,2,2) juga tidak signifikan pada 20 lag awal yang menandakan saling bebas. Kondisi ini akan diuji lebih lanjut dengan uji formal.

Uji Formal

#1) Sisaan Menyebar Normal

```
ks.test(sisaan.da,"pnorm") #tak tolak H0 > sisaan menyebar normal
```

```
##
## Exact one-sample Kolmogorov-Smirnov test
##
## data:  sisaan.da
## D = 0.53061, p-value = 4.441e-16
## alternative hypothesis: two-sided
```

Selain dengan eksplorasi, asumsi tersebut dapat diuji menggunakan uji formal. Pada tahapan ini uji formal yang digunakan untuk normalitas adalah uji Kolmogorov-Smirnov (KS). Hipotesis pada uji KS adalah sebagai berikut.

H_0 : Sisaan menyebar normal

H_1 : Sisaan tidak menyebar normal

Berdasarkan uji KS tersebut, didapat p -value sebesar $4.441e-16$ yang kurang dari taraf nyata 5% sehingga tolak H_0 dan menandakan bahwa sisaan tidak menyebar normal. Hal ini sesuai dengan hasil eksplorasi menggunakan plot kuantil-kuantil normal.

```
#2) Sisaan saling bebas/tidak ada autokorelasi
Box.test(sisaan.da, type = "Ljung") #tak tolak  $H_0$  > sisaan saling bebas

##
## Box-Ljung test
##
## data:  sisaan.da
## X-squared = 0.14467, df = 1, p-value = 0.7037
```

Selanjutnya akan dilakukan uji formal untuk kebebasan sisaan menggunakan uji Ljung-Box. Hipotesis yang digunakan adalah sebagai berikut.

H_0 : Sisaan saling bebas

H_1 : Sisaan tidak saling bebas

Berdasarkan uji Ljung-Box tersebut, didapat p -value sebesar 0.7037 yang lebih besar dari taraf nyata 5% sehingga tak tolak H_0 dan menandakan bahwa sisaan saling bebas.

```
#3) Sisaan homogen
Box.test((sisaan.da)^2, type = "Ljung") #tak tolak  $H_0$  > sisaan homogen

##
## Box-Ljung test
##
## data:  (sisaan.da)^2
## X-squared = 0.69256, df = 1, p-value = 0.4053
```

Hipotesis yang digunakan untuk uji kehomogenan ragam adalah sebagai berikut.

H_0 : Ragam sisaan homogen

H_1 : Ragam sisaan tidak homogen

Berdasarkan uji Ljung-Box terhadap sisaan kuadrat tersebut, didapat p -value sebesar 0.4053 yang lebih dari taraf nyata 5% sehingga tolak H_0 dan menandakan bahwa ragam sisaan tidak homogen. Hal ini sesuai dengan eksplorasi

```
#4) Nilai tengah sisaan sama dengan nol
t.test(sisaan.da, mu = 0, conf.level = 0.95) #tak tolak  $H_0$  > nilai tengah
sisaan sama dengan 0

##
## One Sample t-test
##
## data:  sisaan.da
## t = 0.23724, df = 97, p-value = 0.813
## alternative hypothesis: true mean is not equal to 0
```

```
## 95 percent confidence interval:
## -599.0881 761.7566
## sample estimates:
## mean of x
## 81.33425
```

Terakhir, dengan uji-t, akan dicek apakah nilai tengah sisaan sama dengan nol. Hipotesis yang diujikan sebagai berikut.

H_0 : nilai tengah sisaan sama dengan 0

H_1 : nilai tengah sisaan tidak sama dengan 0

Berdasarkan uji-t tersebut, didapat *p-value* sebesar 0.813 yang lebih besar dari taraf nyata 5% sehingga tak tolak H_0 dan menandakan bahwa nilai tengah sisaan sama dengan nol.

Overfitting

Tahapan selanjutnya adalah overfitting dilakukan dengan menaikkan orde AR(p) dan MA(q) dari model ARIMA (1,2,2) untuk melihat apakah terdapat model lain yang lebih baik dari model saat ini. Kandidat model overfitting adalah ARIMA (2,2,2) dan ARIMA (1,2,3)

```
model5a.ma2=Arima(cabaitrain, order=c(2,2,2),method="ML")
summary(model5a.ma2) #1896.25

## Series: cabaitrain
## ARIMA(2,2,2)
##
## Coefficients:
##          ar1          ar2          ma1          ma2
##          0.1605   -0.0729   -1.2928    0.3174
## s.e.    0.6605    0.1569    0.6597    0.6440
##
## sigma^2 = 13496114: log likelihood = -943.13
## AIC=1896.25  AICc=1896.91  BIC=1909.18
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
ACF1
## Training set 370.2043 3561.79 2432.255 0.674579 4.430264 1.033856 -
0.007322373

lmtest::coeftest(model5a.ma2) #hanya ma1 yang signifikan

##
## z test of coefficients:
##
##      Estimate Std. Error z value Pr(>|z|)
## ar1  0.160510   0.660455  0.2430  0.80798
## ar2 -0.072943   0.156924 -0.4648  0.64205
```

```
## ma1 -1.292832    0.659726 -1.9596    0.05004 .
## ma2  0.317357    0.643961  0.4928    0.62214
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

model5b.ma2=Arima(cabaitrain, order=c(1,2,3),method="ML")
summary(model5b.ma2) #1896.28

## Series: cabaitrain
## ARIMA(1,2,3)
##
## Coefficients:
##          ar1          ma1          ma2          ma3
##          0.0026    -1.1349    0.0678    0.0938
## s.e.    0.9658    0.9613    1.1087    0.1945
##
## sigma^2 = 13499633: log likelihood = -943.14
## AIC=1896.28   AICc=1896.93   BIC=1909.2
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
ACF1
## Training set 370.8077 3562.254 2432.43 0.6756461 4.429495 1.03393 -
0.007072647

lmtest::coeftest(model5b.ma2) #seluruh parameter tidak signifikan

##
## z test of coefficients:
##
##      Estimate Std. Error z value Pr(>|z|)
## ar1  0.0025776  0.9658242  0.0027  0.9979
## ma1 -1.1348629  0.9613269 -1.1805  0.2378
## ma2  0.0678177  1.1086812  0.0612  0.9512
## ma3  0.0938431  0.1945001  0.4825  0.6295

#model yang dipilih adalah model awal, yaitu ARIMA(1,2,2)
```

Peramalan

Peramalan dilakukan menggunakan fungsi `forecast()`. Contoh peramalan berikut ini dilakukan untuk 40 hari ke depan.

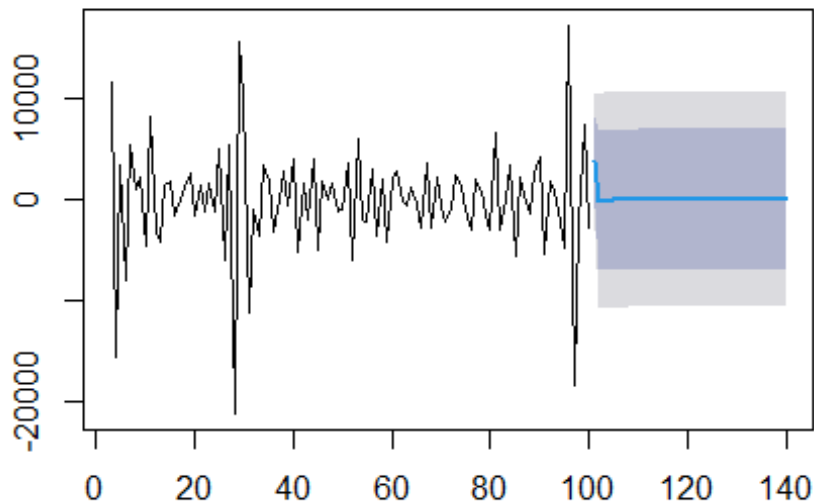
```
#---FORECAST---#
ramalan.da <- forecast::forecast(model5.da, h = 40)
ramalan.da

##      Point Forecast      Lo 80      Hi 80      Lo 95      Hi 95
## 101    3797.019696   -660.0864  8254.126   -3019.536  10613.58
## 102     -94.982290  -7069.7950  6879.830  -10762.038  10572.07
```

## 103	-70.307784	-7048.1499	6907.534	-10741.996	10601.38
## 104	-51.103537	-7030.7801	6928.573	-10725.598	10623.39
## 105	-36.156811	-7016.9444	6944.631	-10712.350	10640.04
## 106	-24.523727	-7005.9842	6956.937	-10701.746	10652.70
## 107	-15.469661	-6997.3378	6966.398	-10693.315	10662.38
## 108	-8.422852	-6990.5378	6973.692	-10686.646	10669.80
## 109	-2.938299	-6985.2028	6979.326	-10681.390	10675.51
## 110	1.330346	-6981.0248	6983.685	-10677.260	10679.92
## 111	4.652647	-6977.7574	6987.063	-10674.022	10683.33
## 112	7.238403	-6975.2048	6989.682	-10671.487	10685.96
## 113	9.250906	-6973.2125	6991.714	-10669.505	10688.01
## 114	10.817243	-6971.6583	6993.293	-10667.958	10689.59
## 115	12.036329	-6970.4466	6994.519	-10666.750	10690.82
## 116	12.985146	-6969.5023	6995.473	-10665.808	10691.78
## 117	13.723614	-6968.7665	6996.214	-10665.074	10692.52
## 118	14.298366	-6968.1934	6996.790	-10664.501	10693.10
## 119	14.745698	-6967.7471	6997.238	-10664.055	10693.55
## 120	15.093857	-6967.3995	6997.587	-10663.708	10693.90
## 121	15.364831	-6967.1289	6997.859	-10663.438	10694.17
## 122	15.575730	-6966.9182	6998.070	-10663.227	10694.38
## 123	15.739874	-6966.7542	6998.234	-10663.063	10694.54
## 124	15.867628	-6966.6266	6998.362	-10662.936	10694.67
## 125	15.967059	-6966.5272	6998.461	-10662.836	10694.77
## 126	16.044446	-6966.4498	6998.539	-10662.759	10694.85
## 127	16.104677	-6966.3896	6998.599	-10662.699	10694.91
## 128	16.151555	-6966.3427	6998.646	-10662.652	10694.96
## 129	16.188040	-6966.3063	6998.682	-10662.615	10694.99
## 130	16.216437	-6966.2779	6998.711	-10662.587	10695.02
## 131	16.238538	-6966.2558	6998.733	-10662.565	10695.04
## 132	16.255739	-6966.2386	6998.750	-10662.548	10695.06
## 133	16.269127	-6966.2252	6998.763	-10662.534	10695.07
## 134	16.279547	-6966.2148	6998.774	-10662.524	10695.08
## 135	16.287657	-6966.2067	6998.782	-10662.516	10695.09
## 136	16.293969	-6966.2003	6998.788	-10662.510	10695.10
## 137	16.298881	-6966.1954	6998.793	-10662.505	10695.10
## 138	16.302705	-6966.1916	6998.797	-10662.501	10695.11
## 139	16.305681	-6966.1886	6998.800	-10662.498	10695.11
## 140	16.307997	-6966.1863	6998.802	-10662.496	10695.11

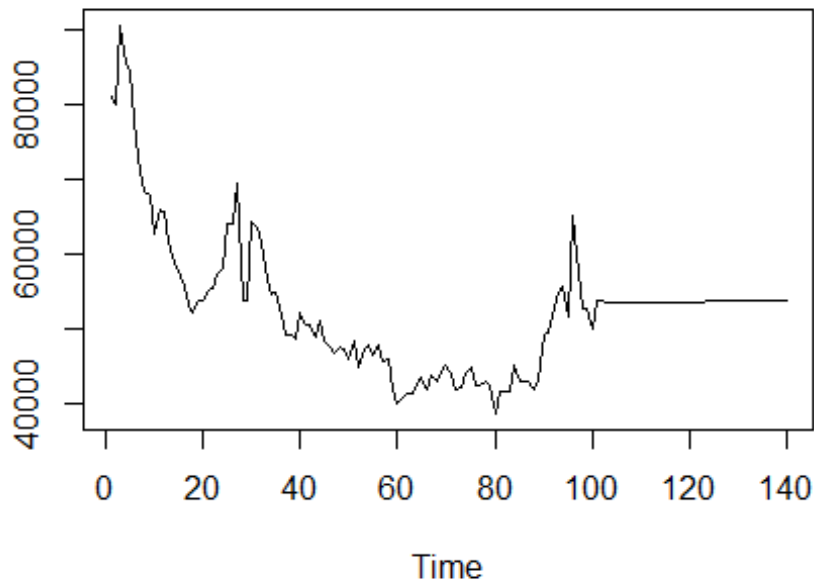
```
data.ramalan.da <- ramalan.da$mean
plot(ramalan.da)
```

Forecasts from ARIMA(1,0,2) with non-zero mean



Berdasarkan hasil plot ramalan di atas, dapat dilihat bahwa ramalan ARIMA(1,2,2) cenderung stabil hingga akhir periode. Selanjutnya, dapat dicari nilai akurasi antara hasil ramalan dengan data uji sebagai berikut.

```
pt_1 <- train.ts[100] #nilai akhir data latih
hasil.forc.Diff <- data.ramalan.da
hasil <- diffinv(hasil.forc.Diff, differences = 1) + pt_1
#has.1 sama hasilnta dengan: cumsum(c(pt_1,hasil.forc.Diff))
ts.plot(train.ts,hasil)
```



```
perbandingan.da<-matrix(data=c(head(test.ts, n=40), hasil[-1]),
                          nrow = 40, ncol = 2)
colnames(perbandingan.da)<-c("Aktual", "Hasil Forecast")
perbandingan.da
```

##		Aktual	Hasil Forecast
##	[1,]	46640	53717.02
##	[2,]	48810	53622.04
##	[3,]	43260	53551.73
##	[4,]	42570	53500.63
##	[5,]	44760	53464.47
##	[6,]	46910	53439.95
##	[7,]	46420	53424.48
##	[8,]	44660	53416.05
##	[9,]	46420	53413.11
##	[10,]	42630	53414.45
##	[11,]	43430	53419.10
##	[12,]	44090	53426.34
##	[13,]	46210	53435.59
##	[14,]	47430	53446.40
##	[15,]	47430	53458.44
##	[16,]	47940	53471.43
##	[17,]	49510	53485.15
##	[18,]	50360	53499.45
##	[19,]	48680	53514.19
##	[20,]	49640	53529.29
##	[21,]	50610	53544.65

```
## [22,] 49980      53560.23
## [23,] 52010      53575.97
## [24,] 50490      53591.84
## [25,] 60250      53607.80
## [26,] 61840      53623.85
## [27,] 67040      53639.95
## [28,] 75660      53656.10
## [29,] 73100      53672.29
## [30,] 66200      53688.51
## [31,] 63570      53704.75
## [32,] 65980      53721.00
## [33,] 77720      53737.27
## [34,] 74050      53753.55
## [35,] 82210      53769.84
## [36,] 75460      53786.13
## [37,] 69540      53802.43
## [38,] 79400      53818.73
## [39,] 84850      53835.04
## [40,] 78630      53851.35
```

```
accuracy(ts(hasil[-1]), head(test.ts, n=40))
```

```
##              ME      RMSE      MAE      MPE      MAPE      ACF1 Theil's U
## Test set 3569.986 13646.04 11221.58 1.590991 18.28176 0.90054 2.900061
```

Hasil peramalan menunjukkan nilai MAPE sebesar 18.28176% yang berarti model ARIMA (1,2,2) termasuk dalam kategori baik. Ini berarti model tersebut cukup akurat dalam memprediksi, tetapi masih ada ruang untuk peningkatan. Model ini dapat digunakan untuk peramalan, meskipun hasilnya mungkin tidak sempurna. Jika diperlukan akurasi yang lebih tinggi, model lain atau pendekatan yang lebih kompleks dapat dipertimbangkan.