Fix-final-project.R

Asus

2025-02-06

```r
#Import library
library(readxl)

library(forecast)

library(tseries)

library(TSA)

library(lmtest)

library(stargazer)

#Import Data
df<-read.csv("C:/Users/Asus/Downloads/mcsft.csv")
View(df)

#Mengcek kelas dan tipe data
class(df)

## [1] "data.frame"

str(df)

## 'data.frame':    435 obs. of  2 variables:
##  $ Date : chr  "2020-01-02" "2020-01-03" "2020-01-06" "2020-01-07" ...
##  $ Close: num  158 156 156 155 157 ...

summary(df)

##      Date               Close      
##  Length:435         Min.   :133.5  
##  Class :character   1st Qu.:185.5  
##  Mode  :character   Median :213.5  
##                     Mean   :218.8  
##                     3rd Qu.:246.4  
##                     Max.   :305.2  

#Mengubah tipe data menjadi time series
tsdata <- ts(df)
class(tsdata)

## [1] "mts"    "ts"     "matrix" "array"

str(tsdata)
```

```
##  Time-Series [1:435, 1:2] from 1 to 435: 1 2 3 4 5 6 7 8 9 10 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : NULL
##   ..$ : chr [1:2] "Date" "Close"
```

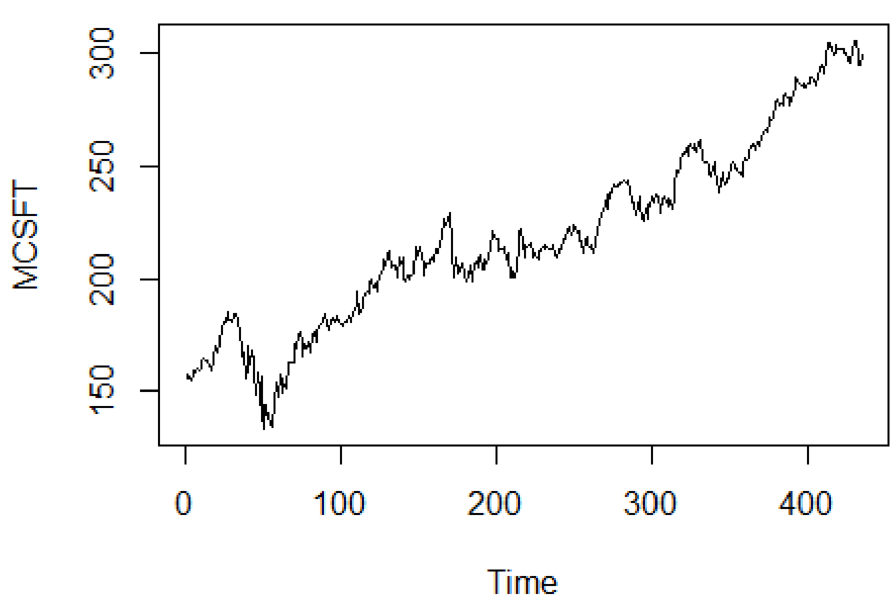
```
summary(tsdata)
```

```
##       Date           Close      
##  Min.   :  1.0   Min.   :133.5  
##  1st Qu.:109.5   1st Qu.:185.5  
##  Median :218.0   Median :213.5  
##  Mean   :218.0   Mean   :218.8  
##  3rd Qu.:326.5   3rd Qu.:246.4  
##  Max.   :435.0   Max.   :305.2
```

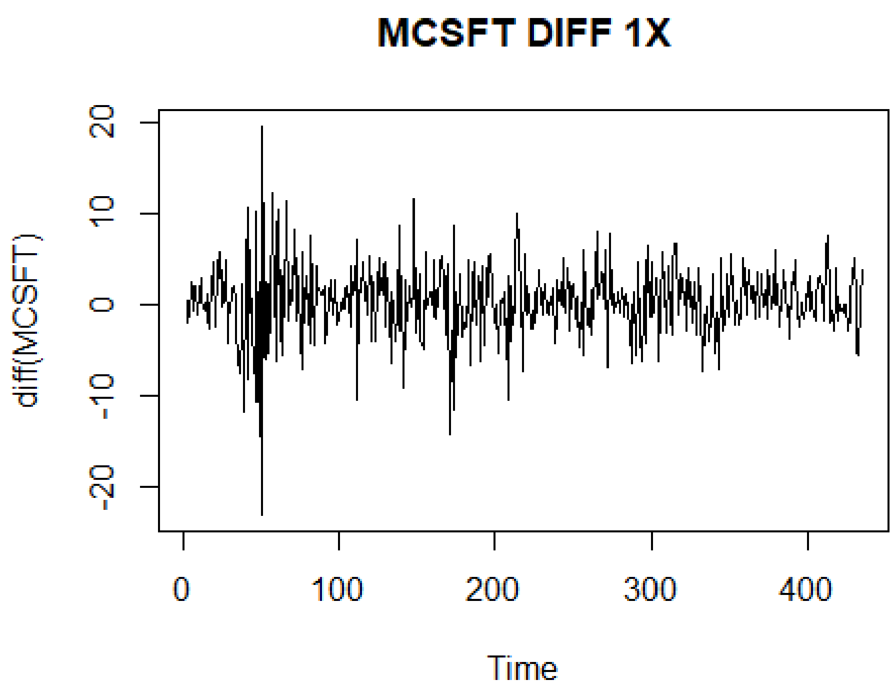
```
# Memeriksa apakah ada nilai yang hilang
has_na <- anyNA(tsdata)
print(has_na)
```

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

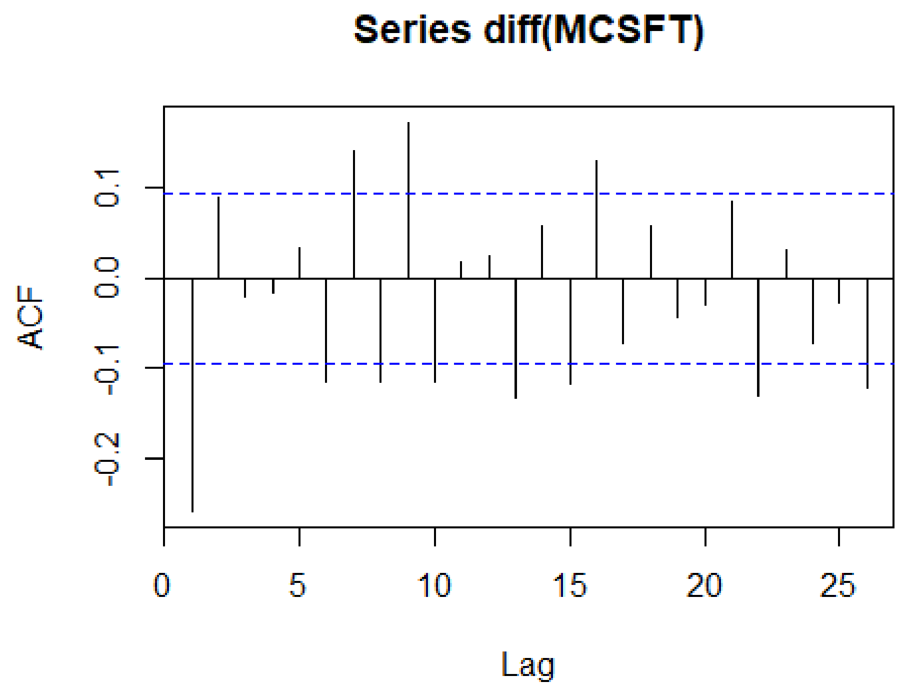
```
#Mengambil nilai dari kolom kedua data time series
MCSFT <- tsdata[,2]
View(MCSFT)

#PLOT DATA MCSFT
plot(MCSFT)
```



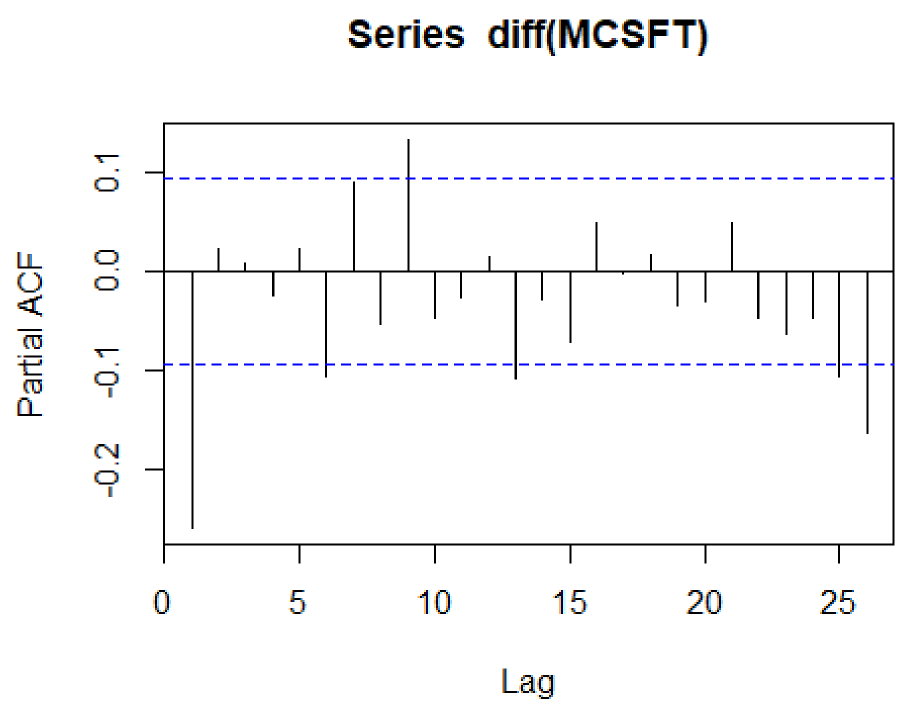
```r
#PLOT DATA MCSFT YG TLH DI DIFFERENCING 1X
plot(diff(MCSFT), main = "MCSFT DIFF 1X")
```




```r
#Uji Stasioneritas Data
adf.test(MCSFT) #tidak stasioner

##
##  Augmented Dickey-Fuller Test
##
## data:  MCSFT
## Dickey-Fuller = -3.0608, Lag order = 7, p-value = 0.1292
## alternative hypothesis: stationary

adf.test(diff(MCSFT)) #stlh di differencing 1x jdi stasioner

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

##
##  Augmented Dickey-Fuller Test
##
## data:  diff(MCSFT)
## Dickey-Fuller = -7.71, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary

# ======================= MODEL SPESIFICATION ===================== #
#Plot ACF dan PACF dari sampel
acf(diff(MCSFT))
```




```
pacf(diff(MCSFT))
```




```
eacf(diff(MCSFT))
```

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

```r
#Dari plot ACF dan PACF kami menduga model ARIMA(2,1,2)

#acf pacf --> p=1, q=1 ARIMA(1,1,1)
#eacf --> ARIMA(0,1,1), (1,1,0) (0,1,2)

# ====================== MODEL ESTIMATION ========================== #



#Menggunakan fungsi auto arima untuk memilih secara otomatis model ARIMA
terbaik
arimaauto <- auto.arima(MCSFT)
summary(arimaauto)
```

```
## Series: MCSFT
## ARIMA(1,1,0) with drift
##
## Coefficients:
##           ar1   drift
##       -0.2589  0.3236
## s.e.   0.0464  0.1520
##
## sigma^2 = 15.94:  log likelihood = -1215.68
## AIC=2437.36   AICc=2437.41   BIC=2449.58
##
## Training set error measures:
##                          ME     RMSE      MAE         MPE     MAPE
MASE
## Training set -0.0008188922 3.978625 2.919575 -0.03593968 1.435965
0.9757554
##                     ACF1
## Training set 0.005522238
```

```r
#Dari fungsi auto arima didapatkan model terbaik ARIMA(1,1,0)

AIC(arimaauto)
```

```
## [1] 2437.358
```

```r
#Akan Dibuat Beberapa Model ARIMA dengan kombinasi p,q yang memungkinkan
Arima.1 <- Arima(MCSFT, order=c(1,1,1)) #acf&pacf
Arima.2 <- Arima(MCSFT, order=c(0,1,1)) #eacf
Arima.3 <- Arima(MCSFT, order=c(1,1,0), include.drift=TRUE) #Berdasarkan
fungsi auto arima
Arima.4 <- Arima(MCSFT, order=c(0,1,2)) #eacf

#Melihat ringkasan dari model
summary(Arima.1)
```

```
## Series: MCSFT
## ARIMA(1,1,1)
##
## Coefficients:
##           ar1    ma1
##       -0.3545  0.110
## s.e.   0.1601  0.169
##
## sigma^2 = 16.09:  log likelihood = -1217.72
## AIC=2441.44   AICc=2441.5   BIC=2453.66
##
## Training set error measures:
##                     ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 0.3932568 3.997389 2.938709 0.1512157 1.444805 0.9821503
##                      ACF1
## Training set -0.009021542
```

```r
summary(Arima.2)
```

```
## Series: MCSFT
## ARIMA(0,1,1)
##
## Coefficients:
##           ma1
##       -0.2239
## s.e.   0.0431
##
## sigma^2 = 16.19:  log likelihood = -1219.55
## AIC=2443.09   AICc=2443.12   BIC=2451.24
##
## Training set error measures:
##                     ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 0.4148932 4.014327 2.941757 0.1598796 1.451248 0.9831689
##                     ACF1
## Training set -0.03199713
```

```r
summary(Arima.3)
```

```
## Series: MCSFT
## ARIMA(1,1,0) with drift
##
```

```
## Coefficients:
##           ar1   drift
##       -0.2589  0.3236
## s.e.   0.0464  0.1520
##
## sigma^2 = 15.94:  log likelihood = -1215.68
## AIC=2437.36   AICc=2437.41   BIC=2449.58
##
## Training set error measures:
##                         ME     RMSE      MAE         MPE     MAPE
MASE
## Training set -0.0008188922 3.978625 2.919575 -0.03593968 1.435965
0.9757554
##                     ACF1
## Training set 0.005522238
```

```
summary(Arima.4)
```

```
## Series: MCSFT
## ARIMA(0,1,2)
##
## Coefficients:
##           ma1     ma2
##       -0.2423  0.0933
## s.e.   0.0477  0.0478
##
## sigma^2 = 16.09:  log likelihood = -1217.67
## AIC=2441.33   AICc=2441.39   BIC=2453.55
##
## Training set error measures:
##                     ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 0.3785478 3.996895 2.936657 0.1452341 1.443495 0.9814643
##                     ACF1
## Training set -0.01064381
```

```
#melihat nilai AIC
AIC(Arima.1)
```

```
## [1] 2441.44
```

```
AIC(Arima.2)
```

```
## [1] 2443.094
```

```
AIC(Arima.3)
```

```
## [1] 2437.358
```

```
AIC(Arima.4)
```

```
## [1] 2441.333
```

```r
#Berdasarkan pemilihan model arima oleh fungsi auto arima dan membandingkan nilai-nilai error pada model
#Kami memilih model terbaik yakni Arima.3
#Model terbaik juga tercermin dalam uji signifikansi koefisien


#Uji Signifikansi Koefisien
coeftest(Arima.1)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value Pr(>|z|)
## ar1 -0.35447    0.16010 -2.2141  0.02682 *
## ma1  0.10997    0.16896  0.6508  0.51516
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
coeftest(Arima.2)
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error z value Pr(>|z|)
## ma1 -0.223949   0.043111 -5.1947 2.05e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

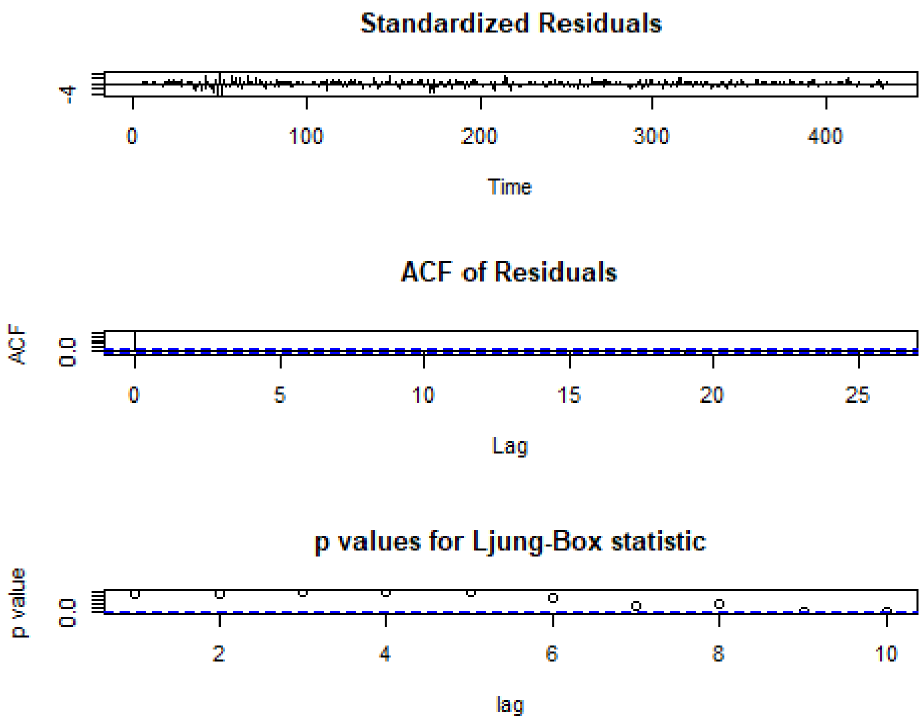
```r
coeftest(Arima.3) #paling signifikan dan nilai Z test paling kecil
```

```
##
## z test of coefficients:
##
##        Estimate Std. Error z value  Pr(>|z|)
## ar1   -0.258903   0.046366 -5.5839 2.351e-08 ***
## drift  0.323585   0.151950  2.1295   0.03321 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

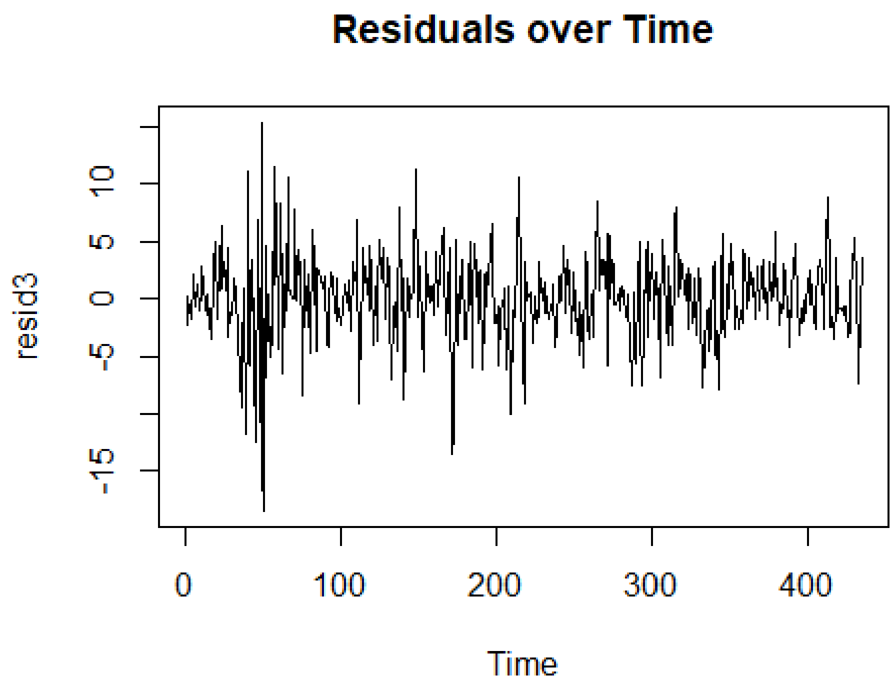
```r
coeftest(Arima.4)
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error z value  Pr(>|z|)
## ma1 -0.242258   0.047706 -5.0782 3.811e-07 ***
## ma2  0.093312   0.047846  1.9503   0.05115 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

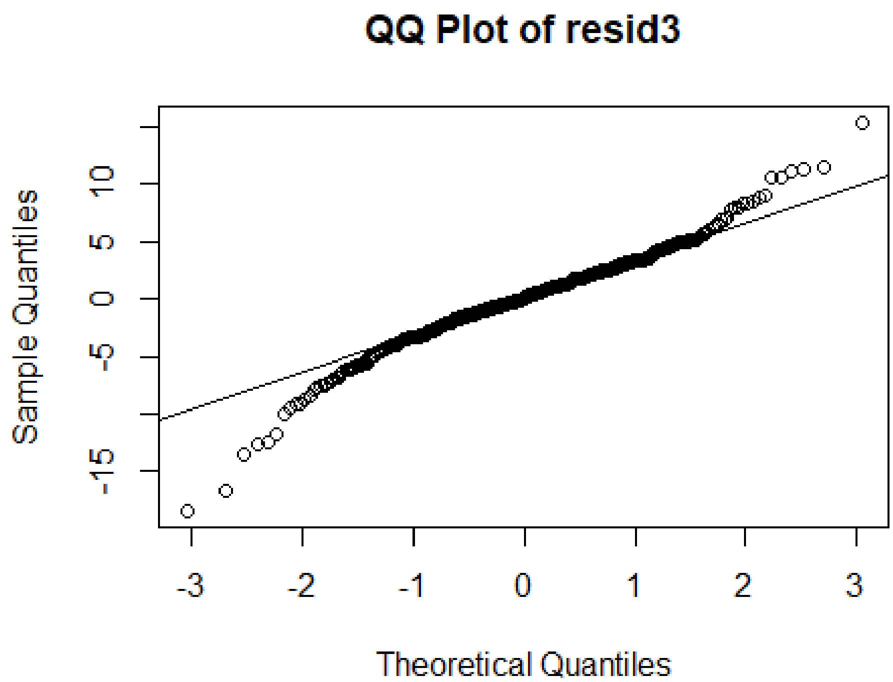
```
# ==================== MODEL DIAGNOSTIC ============================== #
#Melihat grafik standardized residual,
#liat grafik acf ga boleh melebihi garis batas
tsdiag(Arima.3)
```




```
#menyimpan resiudual  model
resid3 <- Arima.3$residuals


# Plot residuals over time
plot(resid3, main="Residuals over Time", type="l")
```

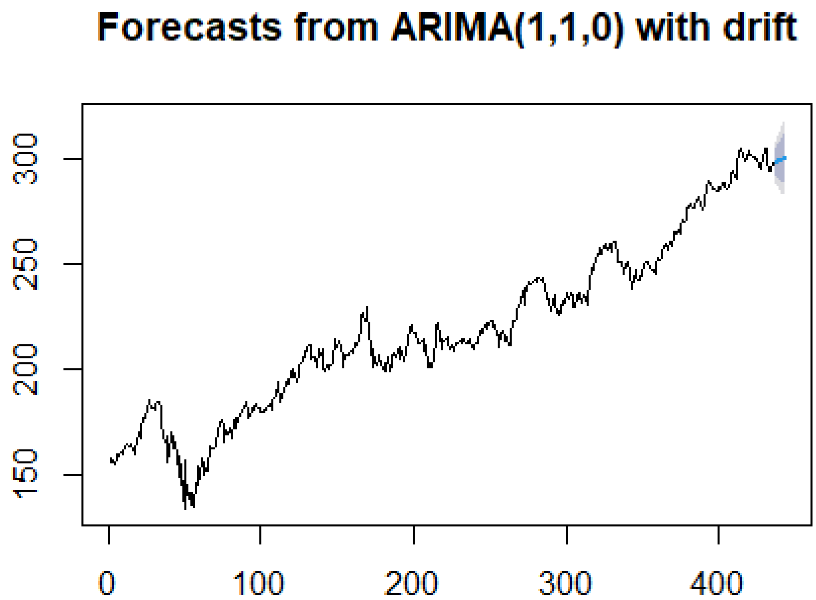



```r
#Uji Normalitas Residual

#QQ-Plot
qqnorm(resid3, main = "QQ Plot of resid3"); qqline(resid3) #model arima.3
```

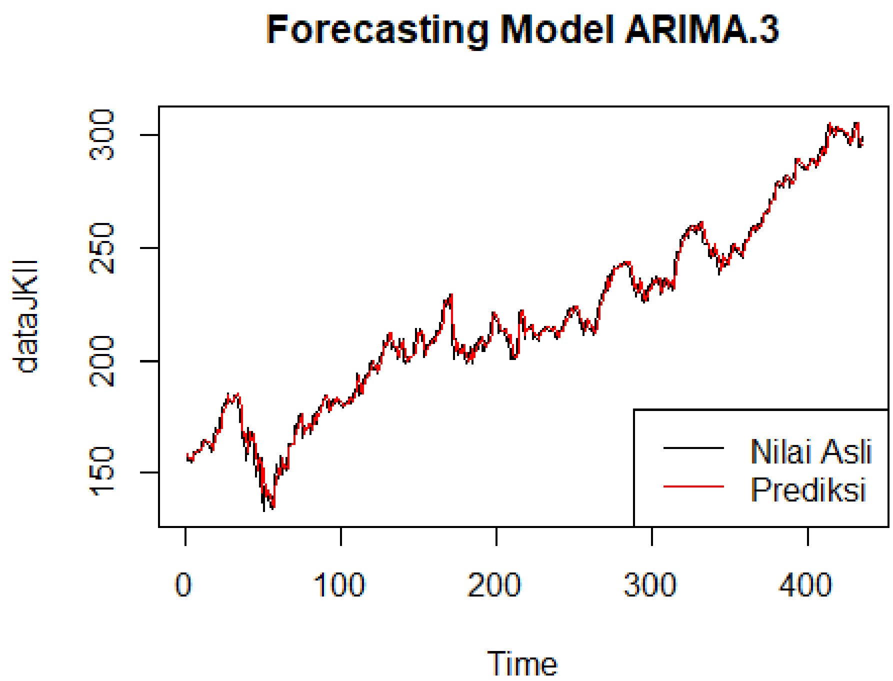



```
#Ljung Box Test
Box.test(resid3,  type = "Ljung-Box")

##
##  Box-Ljung test
##
## data:  resid3
## X-squared = 0.013357, df = 1, p-value = 0.908

#===================================== FORECASTING
=============================== #

#PLOT HASIL FORECASTING
# Generate forecasts for Arima.3
forecast_data3 <- forecast(Arima.3, h = 8)

#print nilai prediksi
forecast_data3

##     Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 436       298.0087 292.8922 303.1252 290.1837 305.8337
## 437       298.5640 292.1956 304.9323 288.8244 308.3036
## 438       298.8276 291.2346 306.4205 287.2152 310.4400
## 439       299.1667 290.5630 307.7703 286.0086 312.3248
## 440       299.4862 289.9689 309.0036 284.9308 314.0417
## 441       299.8109 289.4626 310.1592 283.9845 315.6372
```

```
## 442        300.1342 289.0163 311.2521 283.1308 317.1376
## 443        300.4578 288.6204 312.2953 282.3540 318.5617

plot(forecast_data3)
```




```
#Membandingkan plot forecasting dengan Arima.3
#Plot Forecasting model Arima.3 dengan nilai asli
dataJKII <- df$Close
fit.data = fitted(Arima.3)
par(mfrow=c(1,1))
ts.plot(dataJKII, main = "Forecasting Model ARIMA.3")
lines(fit.data, col="red")
legend("bottomright", legend = c("Nilai Asli", "Prediksi"), col = c("black",
"red"), lty = 1)
```



Forecasting Model ARIMA.3
dataJKII
Time
0
100
200
300
400
150
200
250
300
Nilai Asli
Prediksi