Analisis Runtun Waktu

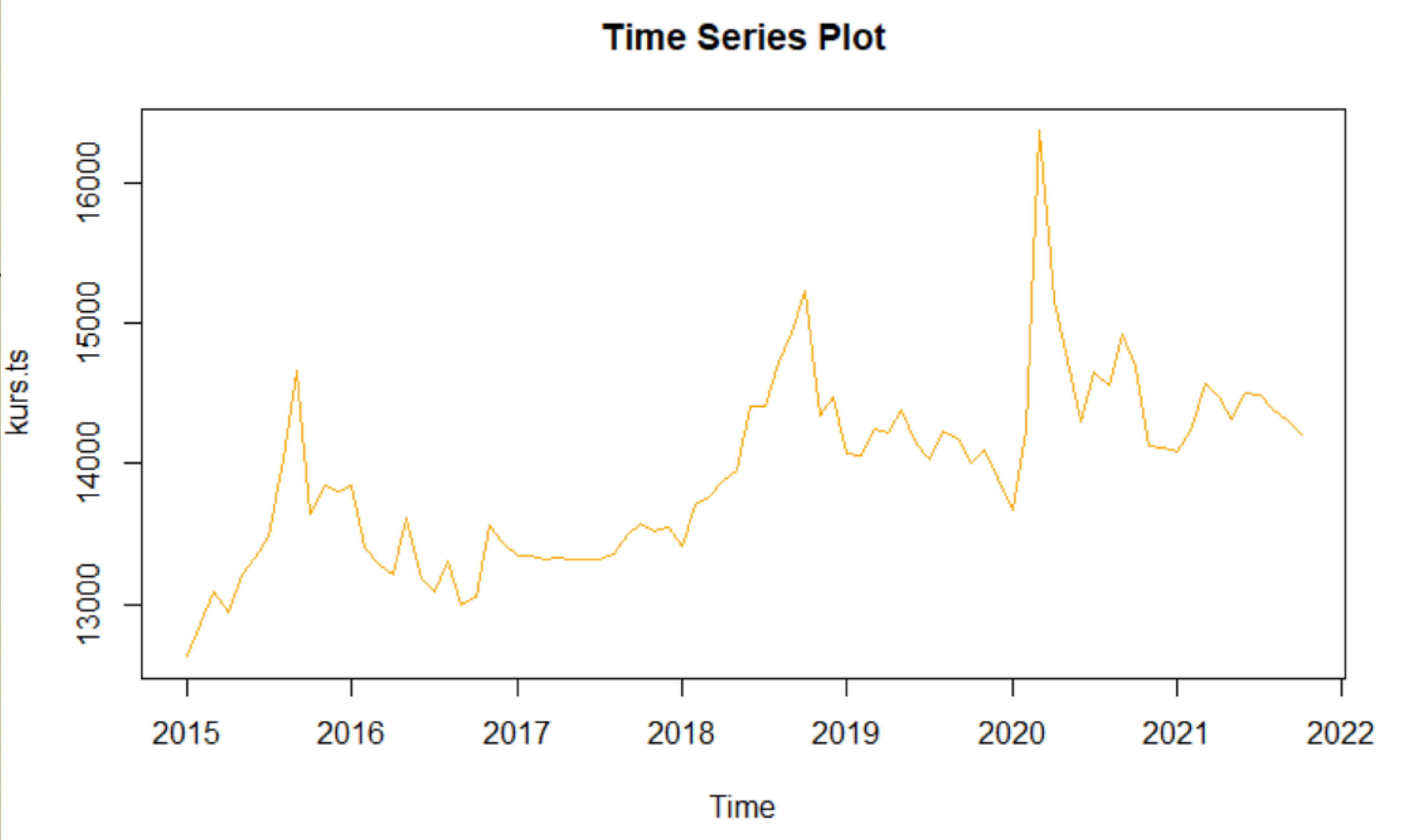
SARIMA

Rizka Putri Maulidya (19611009)

Laila Sirri Hayati (19611050)

Pra Proses Data

```
> kurs = read.csv(file.choose(), header=TRUE, sep=",")
> kurs.ts = ts(kurs$USD, start=c(2015,1), freq=12)
> ts.plot(kurs.ts, col ="orange", main="Time Series Plot")
```

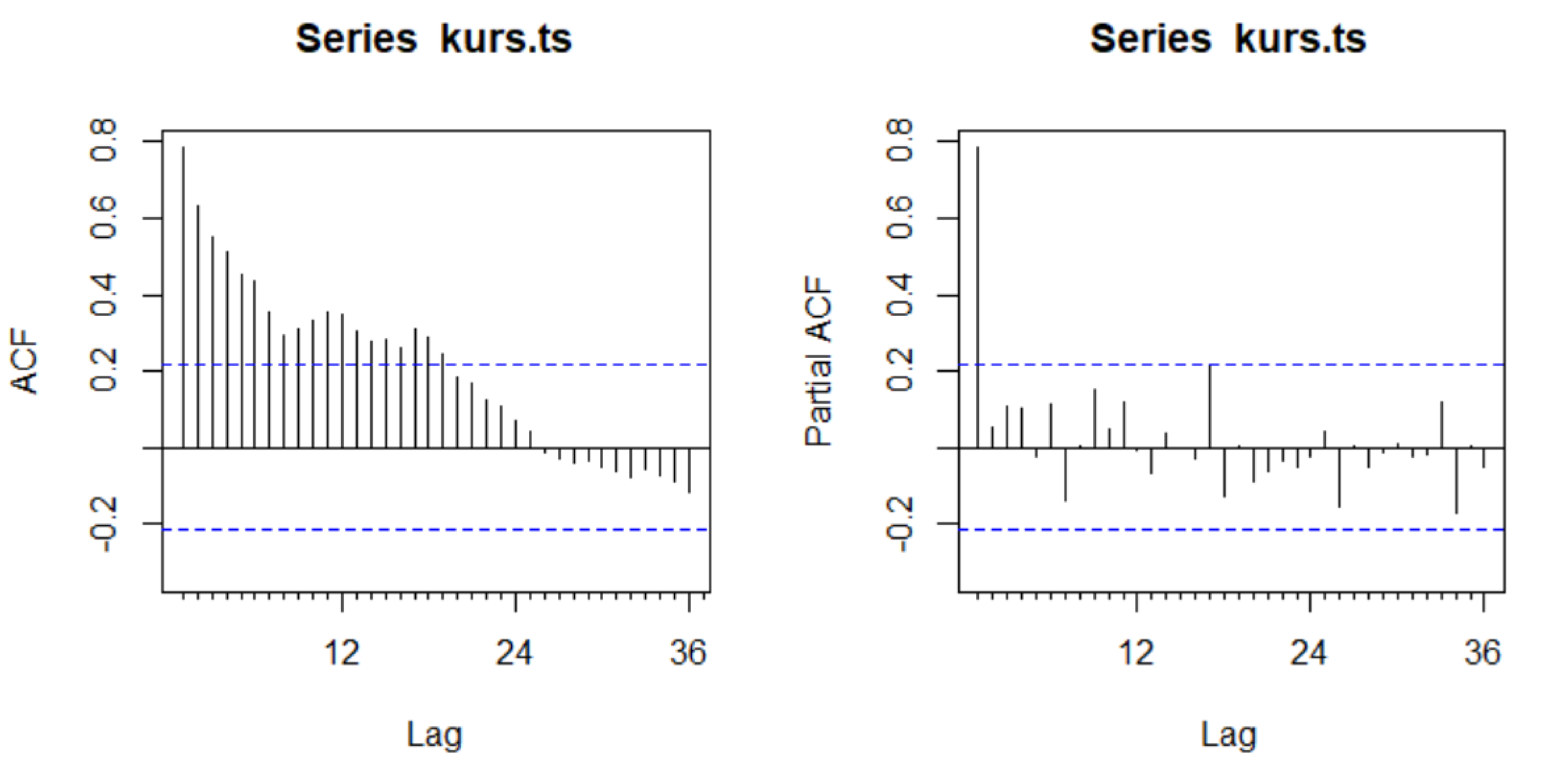


Terlihat adanya komponen musiman dari data "kurs", yaitu adanya perulangan pola yang sama dalam suatu periode waktu yang tetap. Hal tersebut menunjukkan bahwa data tidak stasioner.

Pra Proses Data (Cont'd)

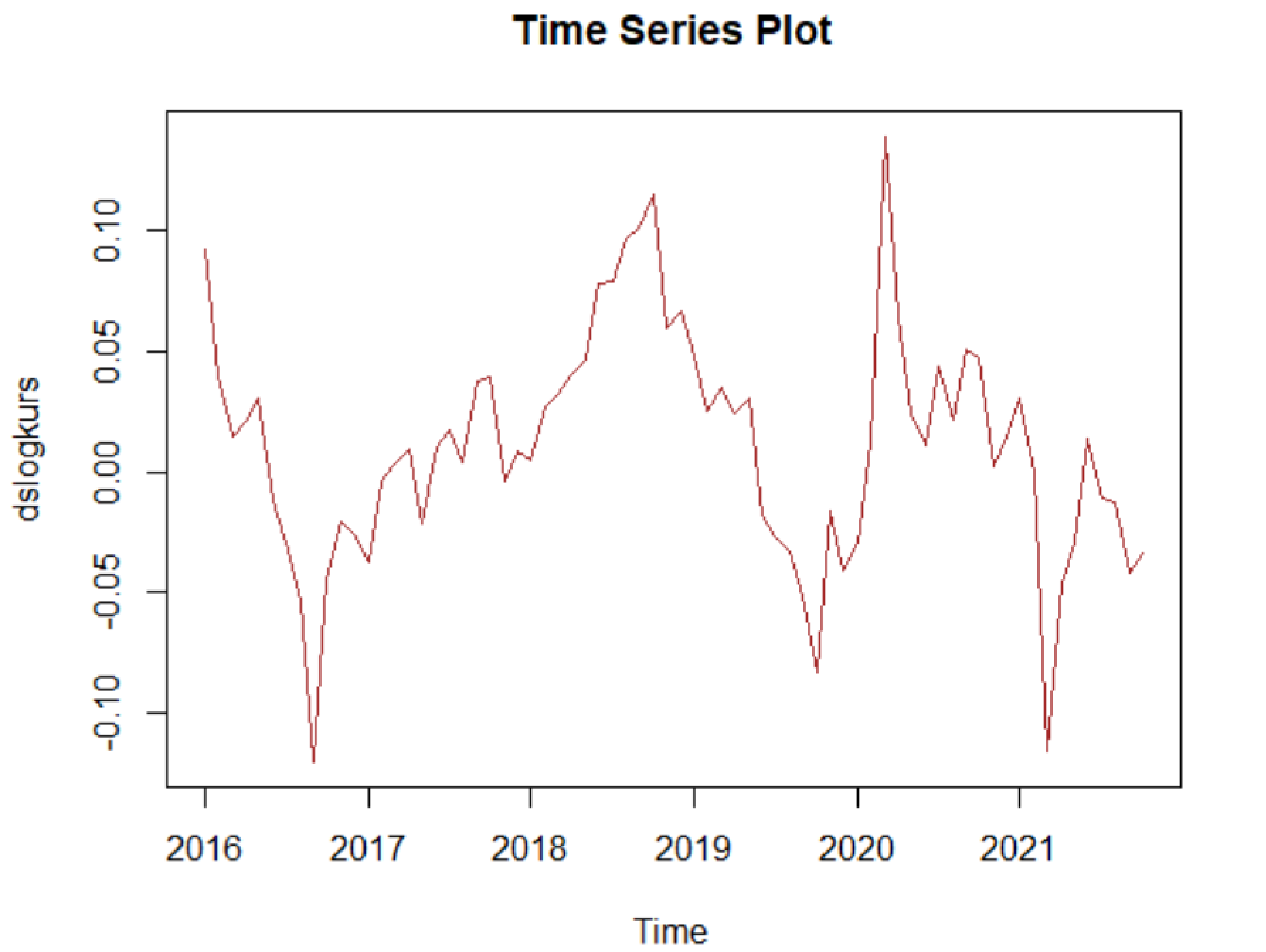
```
> library(forecast)
> par(mfrow=c(1,2))
> Acf(kurs.ts, lag.max=36)
> Pacf(kurs.ts, lag.max =36)
```

Diketahui dari output tersebut pada ACF dan PACF data "kurs" bergelombang karena data tersebut mengandung musiman sehingga membuktikan bahwa data tersebut tidak stasioner. Karena data tersebut terbukti tidak stasioner maka data akan didiferensi.



Transformasi Data Awal

```
> dslogkurs=diff(log(kurs.ts), lag=12, differences = 1)
> ts.plot(dslogkurs, col="brown", main="Time Series Plot")
```



Karena data "kurs" memiliki komponen musiman, maka perlu dilakukan transformasi musiman terhadap data. Akan digunakan deferensi musiman terhadap data log dari data "kurs". Maka untuk melakukan pembedaan orde 1 dengan lag musiman dengan orde musiman s terhadap data logkurs, yaitu menghitung log yt – log yt–12

Tranformasi Data Awal

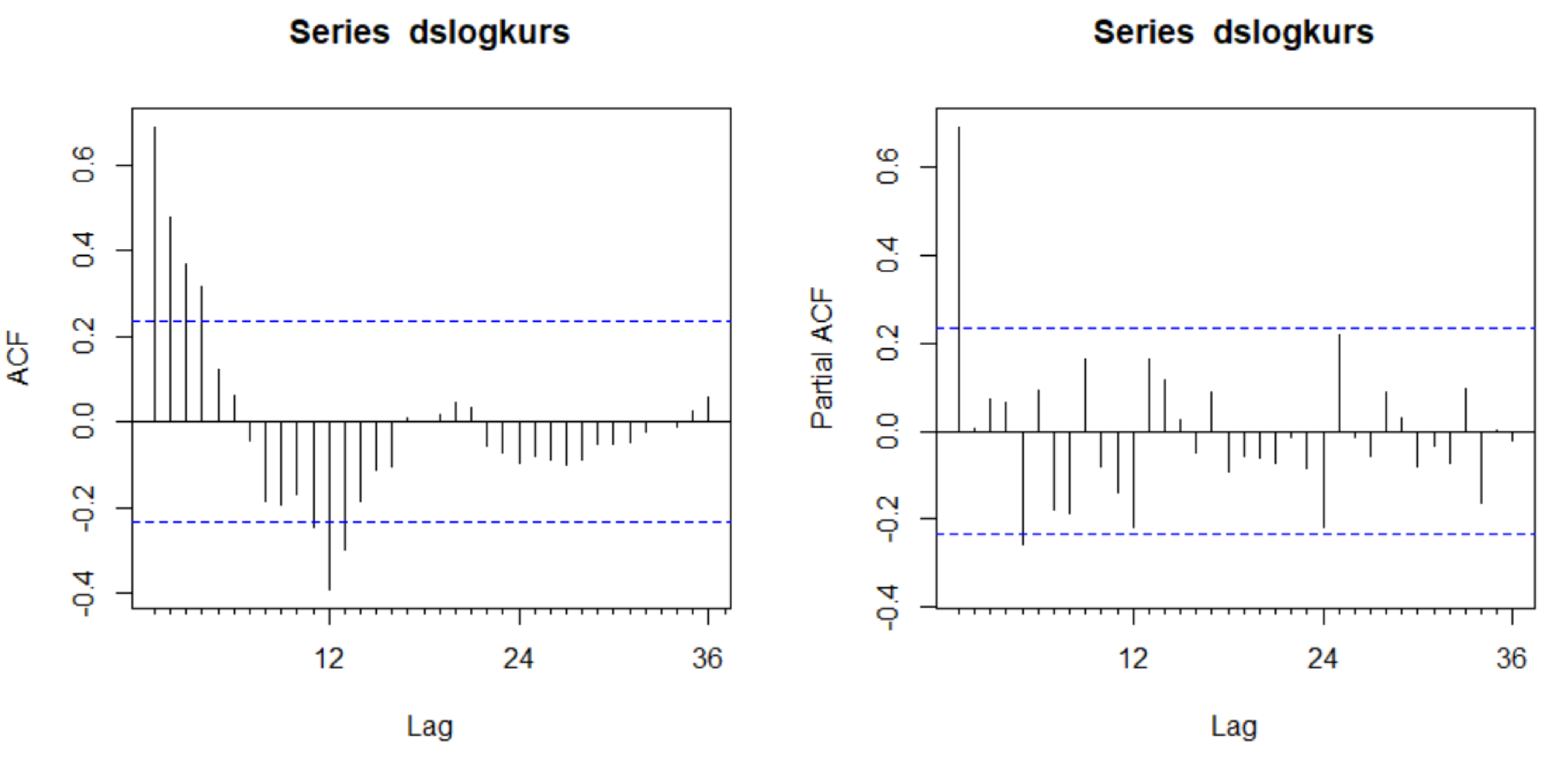
Berdasarkan uji ADF dan plot ACF/PACF, terlihat bahwa masih terdapat sifat nonstasioner dalam data

```
> library(tseries)
> adf.test(dslogkurs)

        Augmented Dickey-Fuller Test

data:  dslogkurs
Dickey-Fuller = -2.8003, Lag order = 4, p-value = 0.2498
alternative hypothesis: stationary

> par(mfrow=c(1,2))
> Acf(dslogkurs, lag.max=36)
> Pacf(dslogkurs, lag.max =36)
```

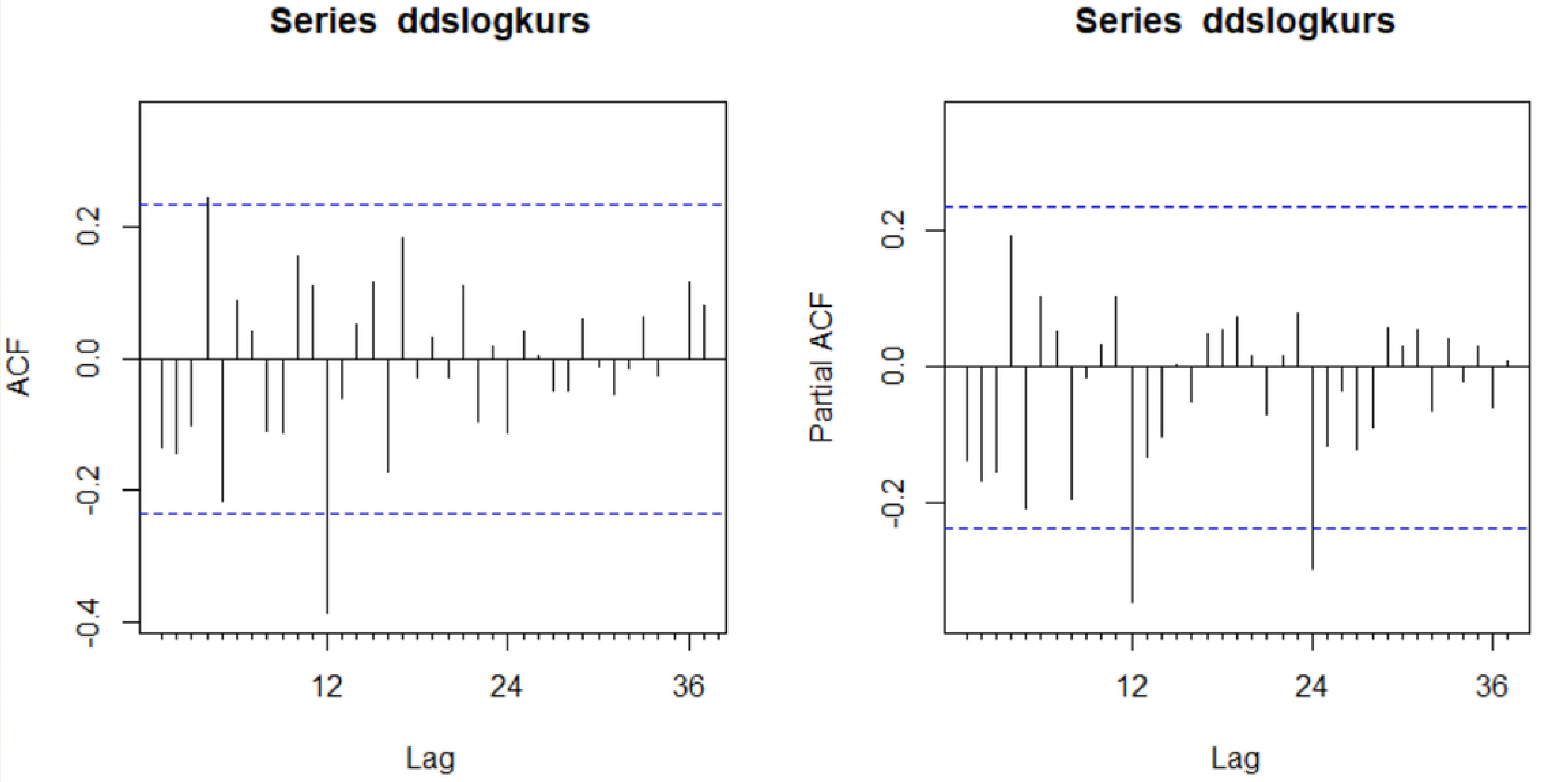


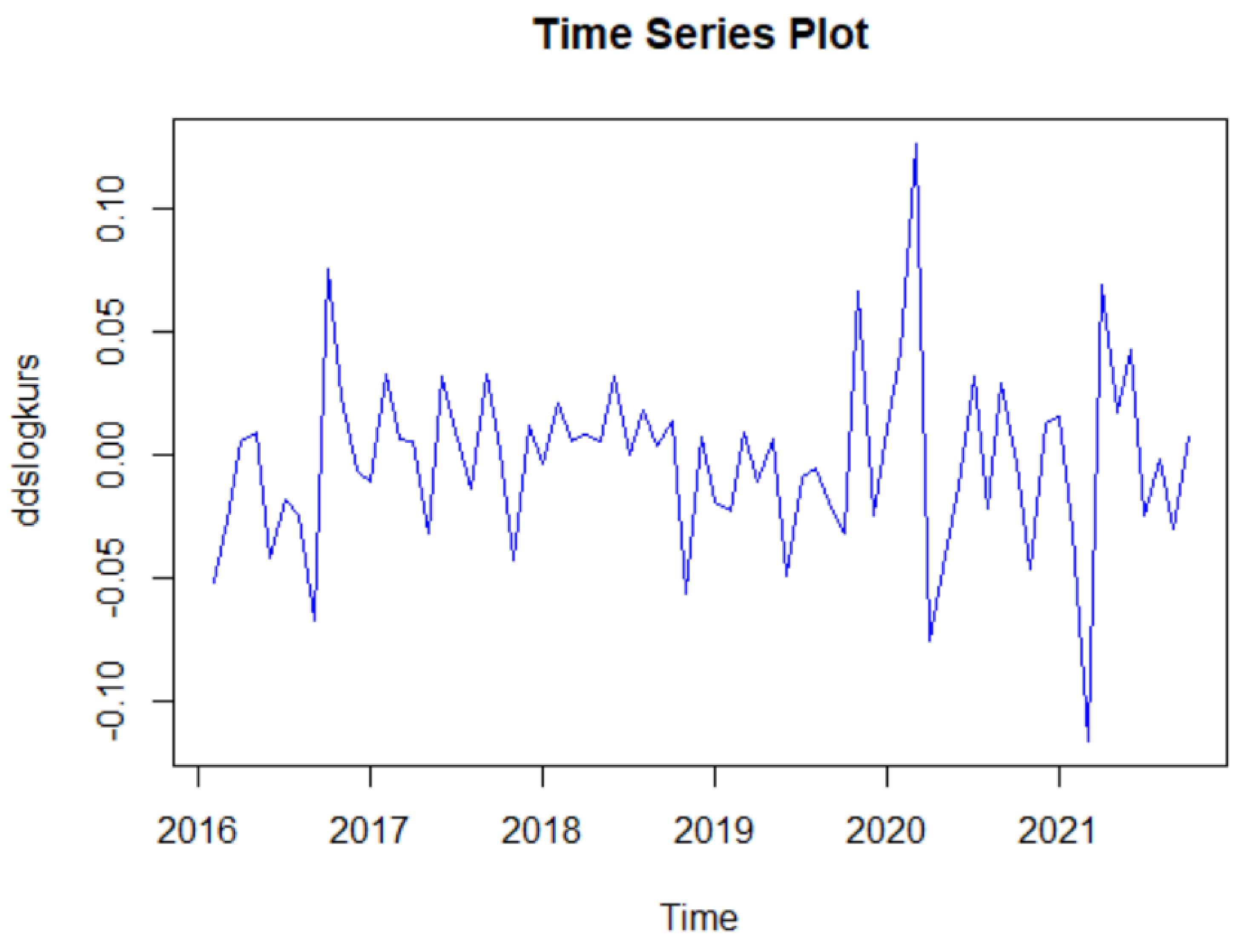
```
> ddslogkurs=diff(dslogkurs, lag=1, difference=1)
> ts.plot(ddslogkurs, col="blue", main="Time Series Plot")
> adf.test(ddslogkurs)

        Augmented Dickey-Fuller Test

data:  ddslogkurs
Dickey-Fuller = -4.405, Lag order = 4, p-value = 0.01
alternative hypothesis: stationary

Warning message:
In adf.test(ddslogkurs) : p-value smaller than printed p-value
> par(mfrow=c(1,2))
> Acf(ddslogkurs, lag.max=37)
> Pacf(ddslogkurs, lag.max =37)
```





Data stasioner

Estimasi Model

Setelah data stasioner, lakukan estimasi parameter dengan ketiga model. Dari plot ACF dan PACF tersebut praktikan akan melakukan estimasi 3 model.

```
> model1=Arima(kurs.ts, order = c(0,1,4), seasonal=list(order=c(2,1,1), period=12),include.mean =F )
> summary(model1)
Series: kurs.ts
ARIMA(0,1,4)(2,1,1)[12]

Coefficients:
          ma1      ma2      ma3     ma4     sar1     sar2    sma1
      -0.2250  -0.1796  -0.1466  0.1605  -0.7003  -0.4132  0.0411
s.e.   0.2053   0.1732   0.1165  0.1484   1.8373   0.8408  2.0428

sigma^2 estimated as 181452:  log likelihood=-515.71
AIC=1047.41   AICc=1049.81   BIC=1065.29

Training set error measures:
                    ME    RMSE      MAE        MPE     MAPE      MASE        ACF1
Training set -39.60243 370.399 231.3738 -0.3164935 1.632043 0.4293788 -0.02766549
```

```
> model2=Arima(kurs.ts, order = c(0,1,3), seasonal=list(order=c(1,1,0), period=12),include.mean =F )
> summary(model2)
Series: kurs.ts
ARIMA(0,1,3)(1,1,0)[12]

Coefficients:
          ma1      ma2      ma3     sar1
      -0.2139  -0.1437  -0.1559  -0.5578
s.e.   0.1318   0.1285   0.1507   0.1103

sigma^2 estimated as 204905:  log likelihood=-520.15
AIC=1050.3   AICc=1051.25   BIC=1061.47

Training set error measures:
                    ME    RMSE      MAE        MPE     MAPE      MASE        ACF1
Training set -32.4766 403.019 267.4818 -0.2696281 1.885302 0.4963872 -0.01764678
```

```
> model3=Arima(kurs.ts, order = c(0,1,1), seasonal=list(order=c(1,1,0), period=12),include.mean =F )
> summary(model3)
Series: kurs.ts
ARIMA(0,1,1)(1,1,0)[12]

Coefficients:
          ma1     sar1
      -0.2697  -0.5621
s.e.   0.1381   0.1096

sigma^2 estimated as 205390:  log likelihood=-521.23
AIC=1048.46   AICc=1048.83   BIC=1055.16

Training set error measures:
                    ME     RMSE      MAE        MPE     MAPE      MASE       ACF1
Training set -27.00742 409.6572 262.6079 -0.2265955 1.849244 0.4873424 0.03563274
```

Estimasi Model

Untuk melihat signifikansi dari koefisien model, perlu ditambahkan fungsi prinstatarima

```
> printstatarima <- function (x, digits = 4,se=T,...){
+   if (length(x$coef) > 0) {
+     cat("\nCoefficients:\n")
+     coef <- round(x$coef, digits = digits)
+     if (se && nrow(x$var.coef)) {
+       ses <- rep(0, length(coef))
+       ses[x$mask] <- round(sqrt(diag(x$var.coef)), digits = digits)
+       coef <- matrix(coef, 1, dimnames = list(NULL, names(coef)))
+       coef <- rbind(coef, s.e. = ses)
+       statt <- coef[1,]/ses
+       pval  <- 2*pt(abs(statt), df=length(x$residuals)-1, lower.tail = F)
+       coef <- rbind(coef, t=round(statt,digits=digits),sign.=round(pval,digits=digits))
+       coef <- t(coef)
+     }
+     print.default(coef, print.gap = 2)
+   }
+ }
```

Estimasi Model

Terlihat koefisien model3 signifikan

```
> printstatarima(model1)

Coefficients:
                    s.e.        t    sign.
ma1    -0.2250  0.2053  -1.0960  0.2763
ma2    -0.1796  0.1732  -1.0370  0.3028
ma3    -0.1466  0.1165  -1.2584  0.2119
ma4     0.1605  0.1484   1.0815  0.2827
sar1   -0.7003  1.8373  -0.3812  0.7041
sar2   -0.4132  0.8408  -0.4914  0.6244
sma1    0.0411  2.0428   0.0201  0.9840
> printstatarima(model2)

Coefficients:
                    s.e.        t    sign.
ma1    -0.2139  0.1318  -1.6229  0.1085
ma2    -0.1437  0.1285  -1.1183  0.2667
ma3    -0.1559  0.1507  -1.0345  0.3040
sar1   -0.5578  0.1103  -5.0571  0.0000
> printstatarima(model3)

Coefficients:
                    s.e.        t    sign.
ma1    -0.2697  0.1381  -1.9529  0.0543
sar1   -0.5621  0.1096  -5.1286  0.0000
```

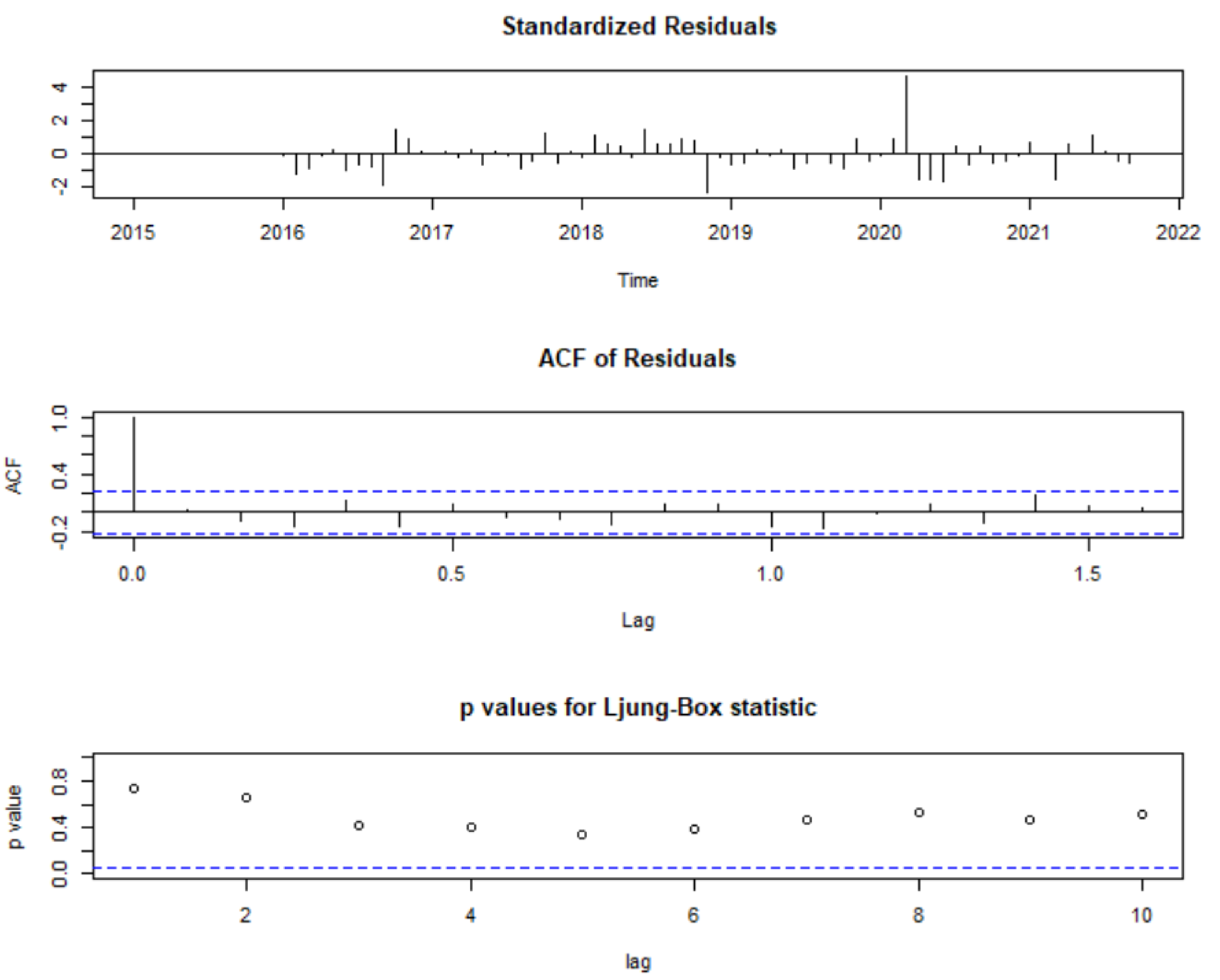
Uji Diagnostik

```
> acfStat <- function(x,lag=36)
+ {
+   out1=acf(x,lag.max=lag,plot=F,na.action=na.pass)
+   acfout=out1$acf
+   out2=pacf(x,lag.max=lag,plot=F,na.action=na.pass)
+   pacfout=NULL
+   pacfout[1]=1
+   pacfout=c(pacfout,out2$acf)
+   temp1=NULL
+   temp1[1]=NULL
+   temp2=NULL
+   temp2[1]=NULL
+   for (i in 1:lag)
+   {
+     temp1[i+1]=Box.test(x,lag=i,type="Ljung")$statistic
+     temp2[i+1]=Box.test(x,lag=i,type="Ljung")$p.value
+   }
+   result=cbind(ACF = acfout, PACF = pacfout, "Q-Stats" = temp1, "P-Value" = temp2)
+   rownames(result)= 0:lag
+   # print(length(acfout))
+   # print(length(pacfout))
+   # print(length(temp1))
+   # print(length(temp2))
+   print(result)
+ }
```

Uji Diagnostik

```
> tsdiag(model3)
```

Pada hasil output data residualnya mengandung whitenoise karena tidak terdapat lag yang keluar dari garis. Dari uji diagnosis didapatkan nilai korelasi residualpada model3 sebanyak 36 lag didapatkan nilai p-value lebih dari alpha (0.05) yang menunjukkan bahwa tidak terdapatkorelasi menurut uji LjungBox.

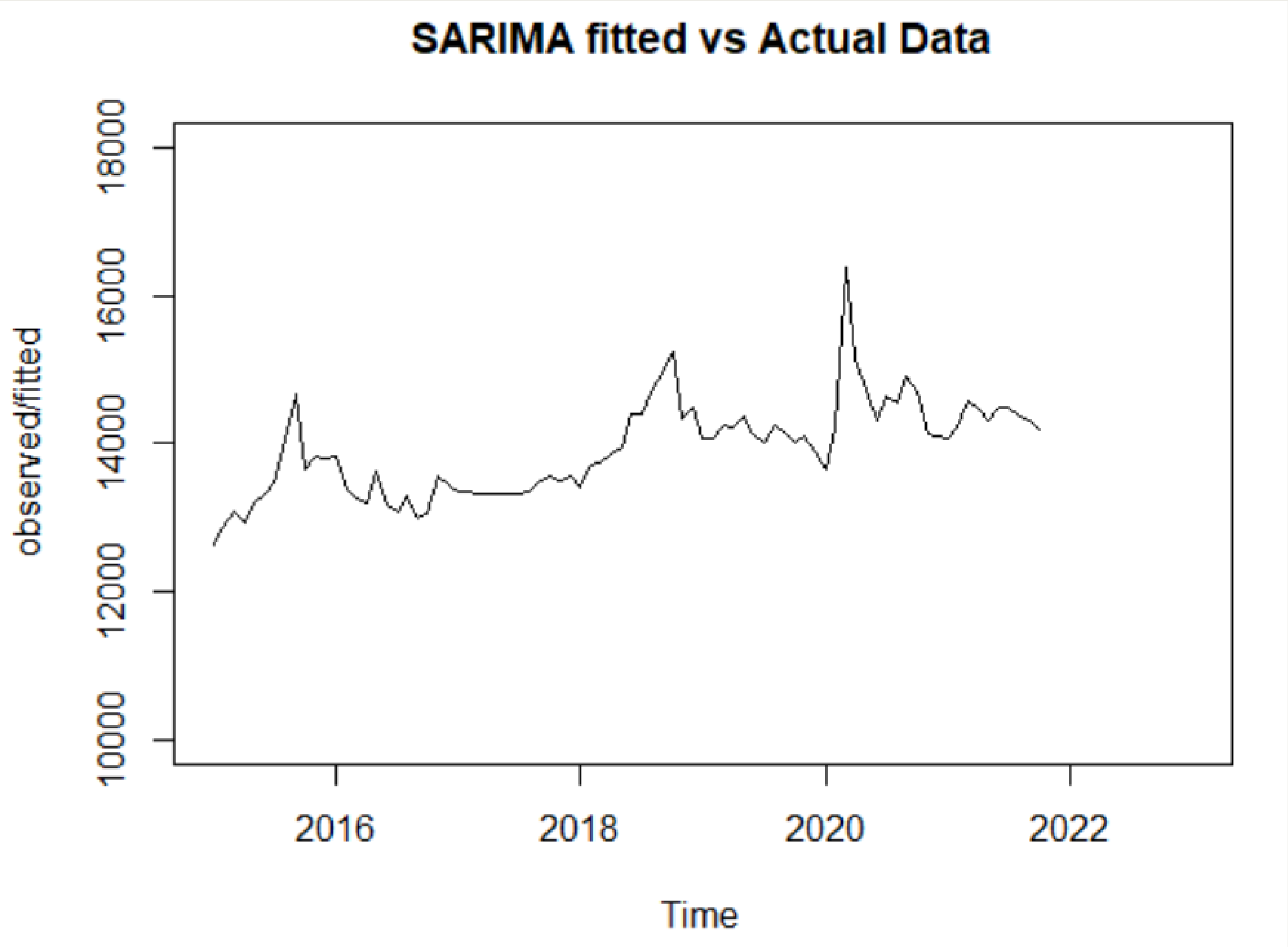



```
> acfStat(model3$residuals, lag=36)
            ACF         PACF     Q-Stats   P-Value
0   1.000000000  1.000000000          NA        NA
1   0.035632739  0.035632739   0.1079709 0.7424662
2  -0.093823469 -0.095214054   0.8658957 0.6485943
3  -0.150636595 -0.145080144   2.8443545 0.4162503
4   0.115653477  0.119956011   4.0255341 0.4025612
5  -0.136670269 -0.179746377   5.6964343 0.3368864
6   0.082605360  0.105518136   6.3148721 0.3888577
7  -0.058375008 -0.071206753   6.6278299 0.4686300
8  -0.071098237 -0.115059559   7.0983509 0.5260595
9  -0.128605407 -0.066737808   8.6589406 0.4693336
10  0.080104853  0.006508546   9.2728133 0.5064232
11  0.086808681  0.083749287  10.0038869 0.5300376
12 -0.138892230 -0.198805876  11.9021264 0.4535717
13 -0.163489636 -0.116448197  14.5703640 0.3349358
14 -0.017005125 -0.049349150  14.5996557 0.4060567
15  0.076989091  0.003849189  15.2090196 0.4364686
16 -0.109451445 -0.150416979  16.4592561 0.4213932
17  0.181739298  0.171790231  19.9593322 0.2763146
18  0.066515683  0.024543496  20.4355013 0.3088422
19  0.039877353  0.035417301  20.6093635 0.3587812
20 -0.010798640  0.077818679  20.6223186 0.4196543
21  0.076630566 -0.070686232  21.2854008 0.4416372
22 -0.099335807 -0.034464899  22.4182016 0.4351771
23  0.012108215  0.027417258  22.4353176 0.4941185
24 -0.138121521 -0.171183054  24.7009416 0.4221547
25  0.003046194 -0.033310119  24.7020630 0.4791666
26 -0.077983872 -0.077934489  25.4500855 0.4936277
27 -0.049868457 -0.122533860  25.7615312 0.5318637
28 -0.013111167 -0.006654213  25.7834583 0.5849460
29  0.105656629  0.030167552  27.2342691 0.5590672
30 -0.120708809 -0.116224308  29.1643154 0.5089773
```

Model Terbaik

```
> logkurs.fitted=fitted(model3)
> kurs.fitted=exp(logkurs.fitted)
> ts.plot(kurs.ts, ylab="observed/fitted", xlim=c(2015, 2023), ylim=c(10000,18000), main="SARIMA fitted vs Actual Dat
```

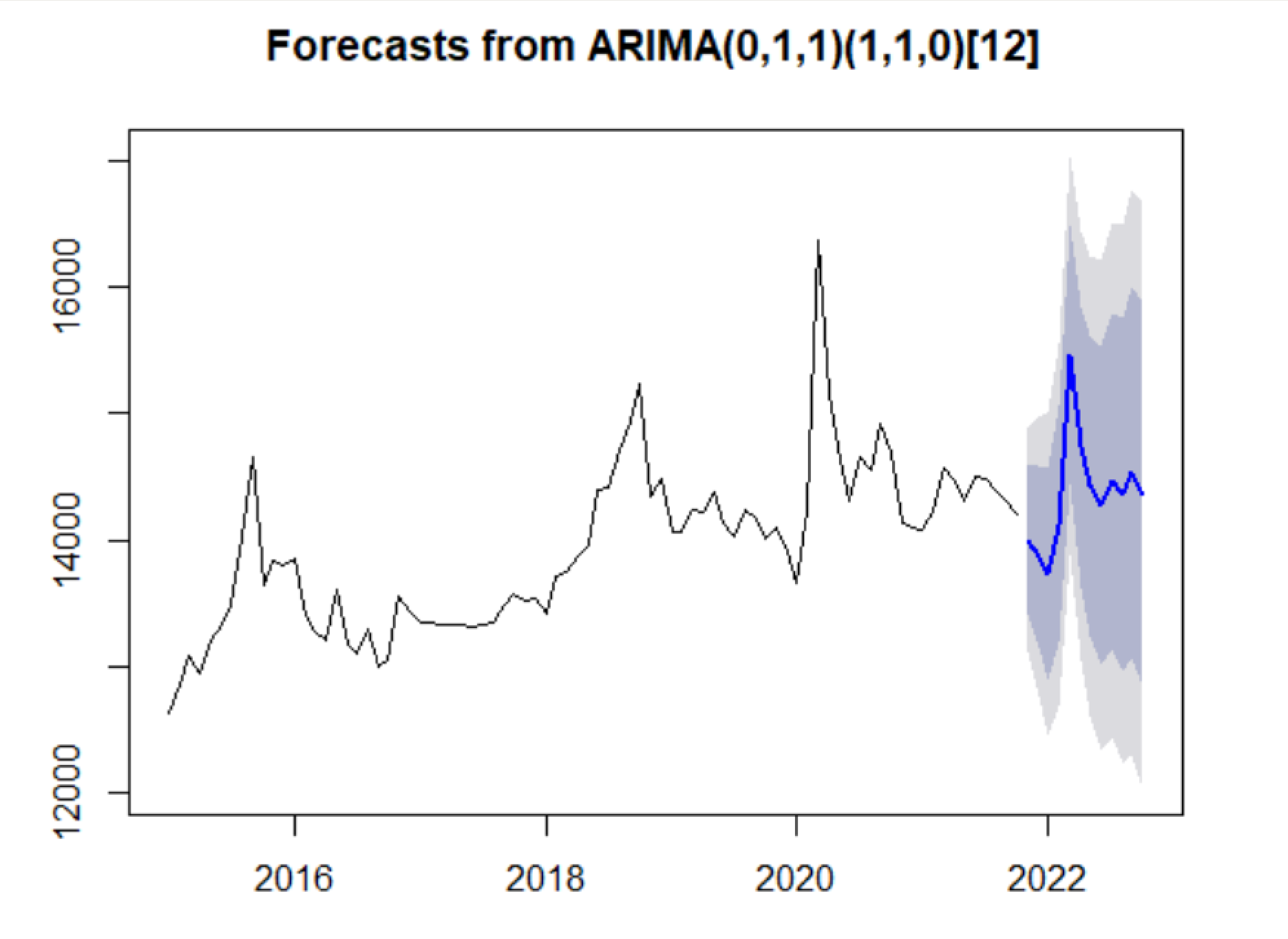
```
> kurs.fitted
          Jan      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep
2015 12556.35 12831.49 13062.82 12921.52 13198.13 13321.08 13471.41 14017.75 14647.80
2016 14309.33 13973.57 13626.18 13272.04 13657.87 13610.34 13302.65 13665.70 13893.64
2017 13098.45 13312.92 13397.90 13201.43 13703.06 13226.27 13369.95 13767.04 13491.20
2018 13501.18 13299.24 13487.23 13671.86 13956.56 13731.04 14291.42 14463.15 14469.81
2019 14567.80 14436.69 13995.36 14409.53 14187.47 14503.64 14237.98 14203.56 14528.34
2020 13588.77 13730.10 14398.68 15967.94 15042.79 14890.32 14827.33 14923.02 14726.45
2021 13862.90 14298.05 15248.83 14108.89 14465.15 14013.98 14389.08 14555.85 14430.86
          Oct      Nov      Dec
2015 13632.23 13833.57 13863.05
2016 12385.88 13341.17 13355.48
2017 13105.21 13905.77 13332.08
2018 15020.86 15393.08 14387.61
2019 14353.29 13635.25 14224.21
2020 14925.46 14296.67 14071.50
2021 14238.81
```



Fitting model seasonal ARIMA terbaik

Forecasting Model

```
> logkurs.pred=predict(model3, n.ahead=12)
> logkurs.pred=forecast(model3, 12)
> plot(logkurs.pred)
```



Hasil peramalan dengan menggunakan model seasonal ARIMA terbaik