SARIMA Time Series Modeling

João Pedro Apolonio de Sousa Matos

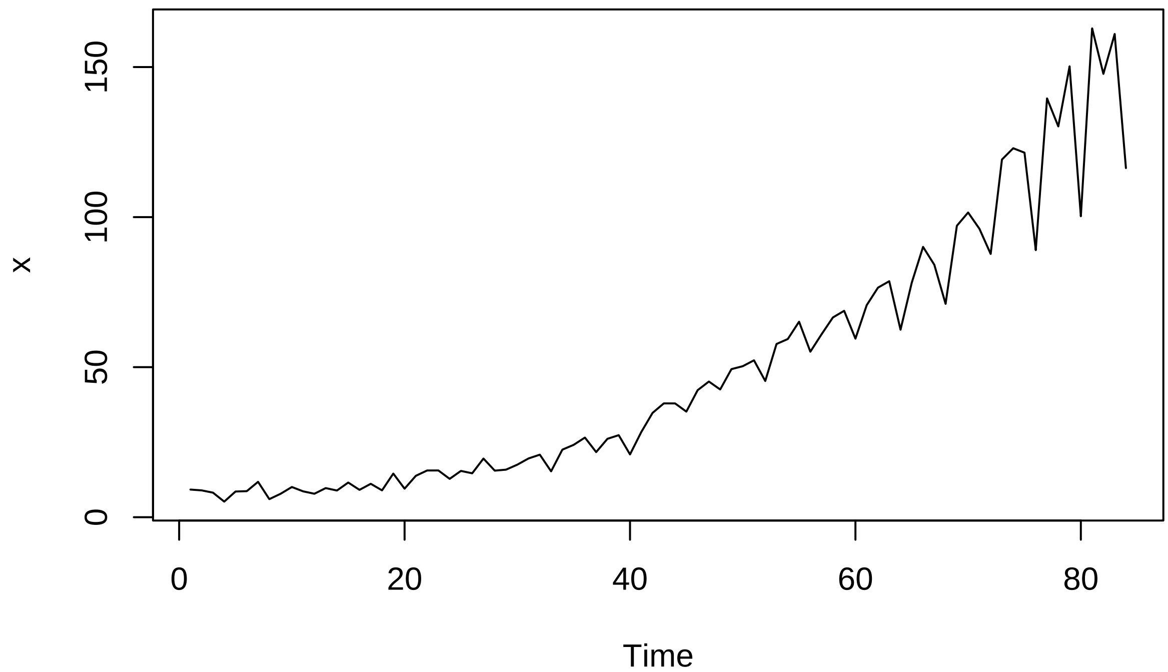
This project will use a SARIMA model for predicting a real Complex Time Series. It uses both Moving Averages and Autoregressive techniques combined with precautions for Tendencies and Seasonal Effects. The function auto.arima will not be used as it is very consistently shows limited results.

It has 5 parts:

a) Preparing the data: Making it stationary, removing Tendencies and finding the period.
b) Seasonal analysis and Model selection: findind the Seasonalities and models that reduce.
c) Model Diagnostics and Comparison: Comparing three models in terms of ACF and PACF.
d) Comparing Models Performance using MSE.
e) Forecasting.

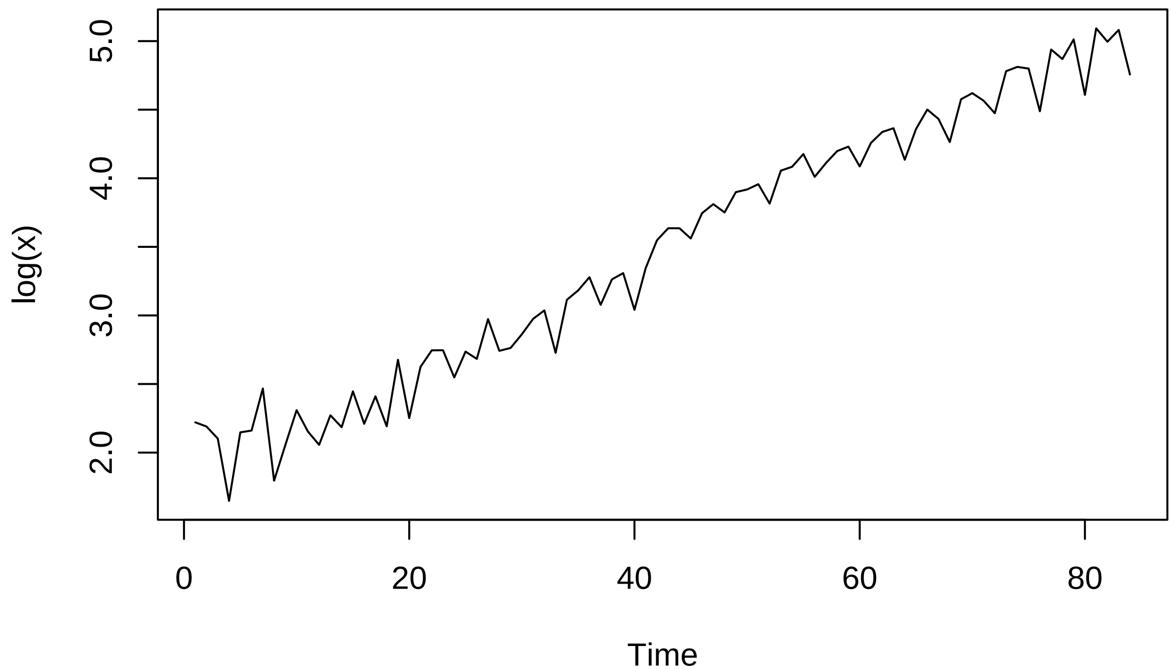
Part a) We will take a look on the Time Series and before calculating the periodogram, we will apply transformations to make it stationary.

```
library(GeneCycle)
library(forecast)
library(tseries)
library(astsa)
library(TSA)
library(lmtest)

# Loading the dataset
dt <- read.csv("C:\\Users\\USER\\Downloads\\MAE0327 - series para prova 2 - serie14 (1).csv")
x <- ts(dt$Xt)

# Plot the original time series
plot(x)
```



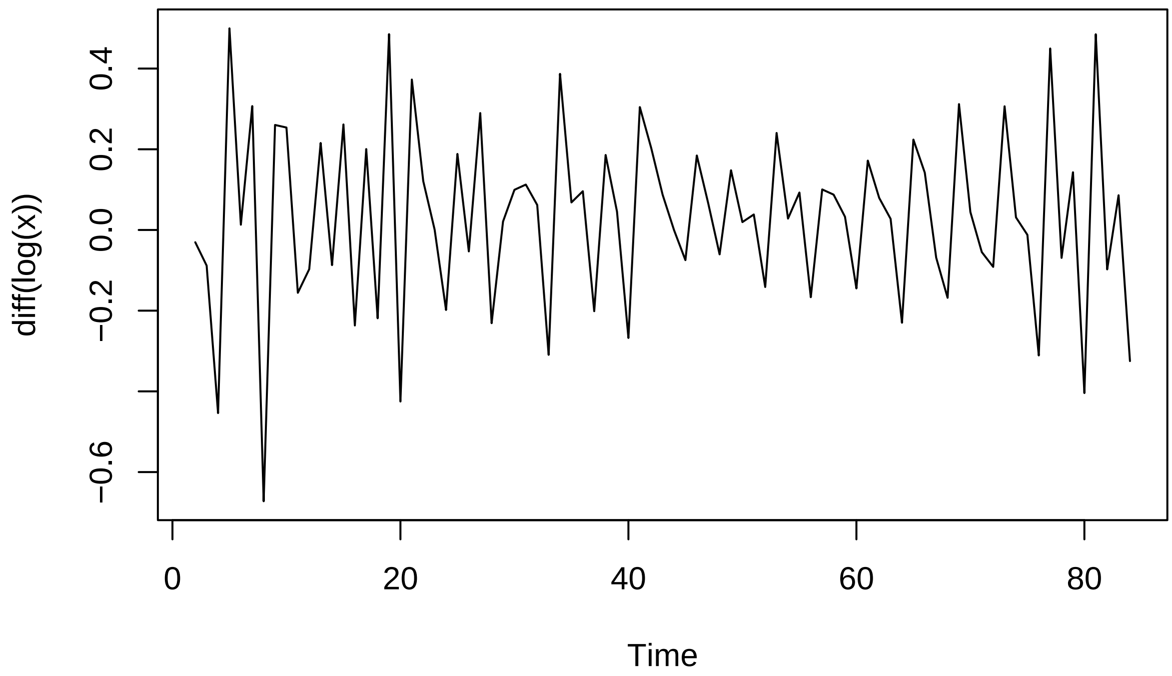
Clearly, the series is not stationary, so we will apply the log transformation

```
plot(log(x))
```



Apply first-order differencing to remove the trend

```r
plot(diff(log(x)))
```



Testing stationarity with the Augmented Dickey-Fuller test

```
adf.test(diff(log(x)))
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(log(x))
## Dickey-Fuller = -6.0174, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

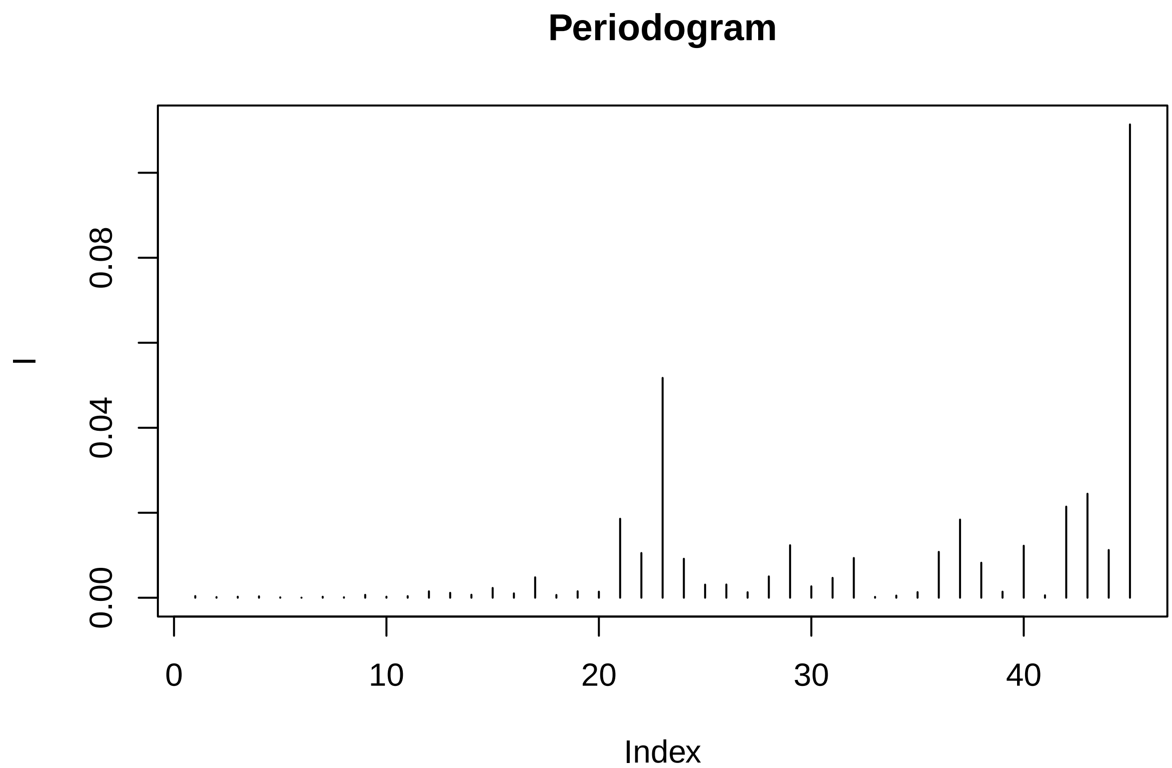
Conclusion: We reject the null hypothesis of a unit root, so the series is stationary.

Assigning the transformed series to 'dl' and performing Fisher's test to find periodicity and plotting the periodogram.

```
dl <- diff(log(x))
fisher.g.test(dl)
```

```
## [1] 0.002797476
```

```
periodogram <- spec.pgram(dl, plot = FALSE, main = "Periodogram of Time Series x")
I <- periodogram$spec / (2 * pi)
plot(I, type = "h", main = "Periodogram")
```



Finding the dominant period

```
freq <- periodogram$freq[which.max(I)]
period <- 1 / freq
period
```

```
## [1] 2
```

However, considering the second-largest peak, we find a period that makes more sense given the quarterly data

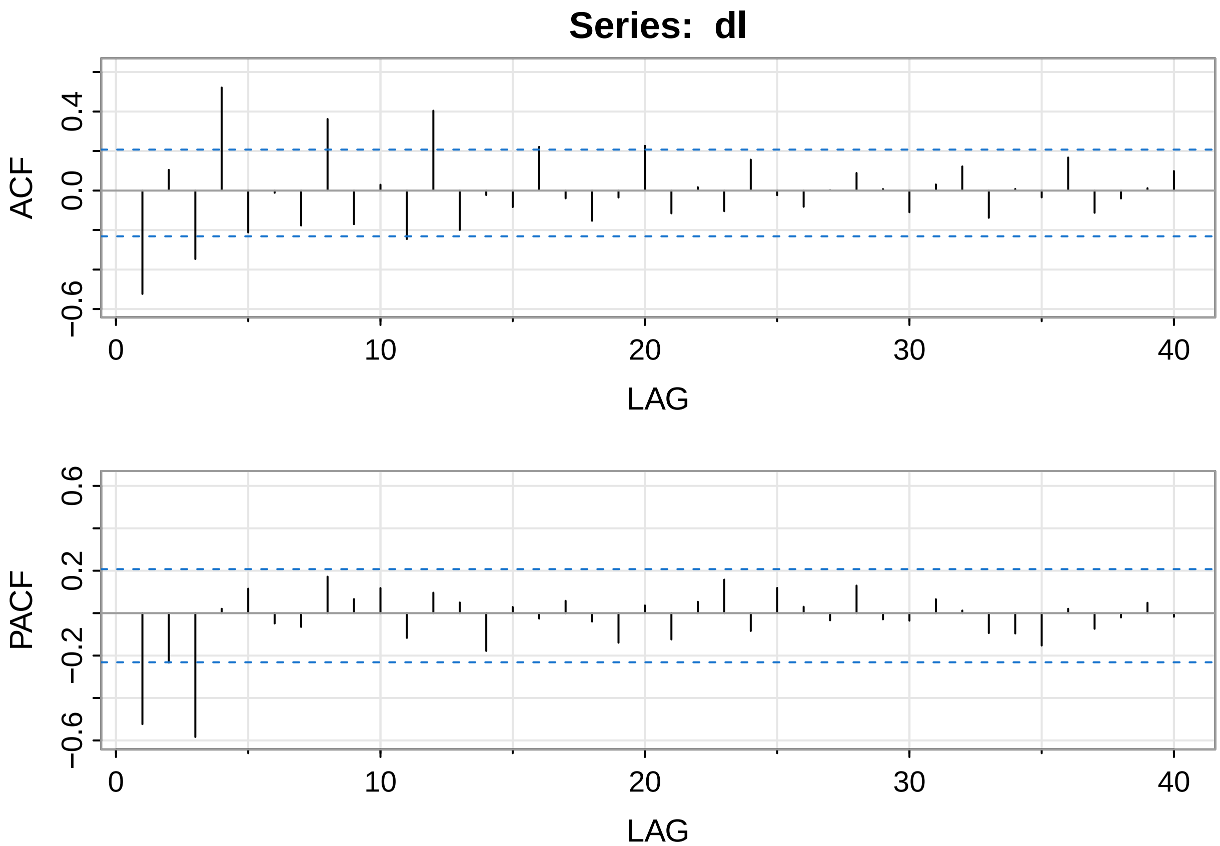
```
freq <- periodogram$freq[which.max(I[1:44])]
period <- 1 / freq
period
```

```
## [1] 3.913043
```

b) Now we will select the model that best fits the presented data.

Separating data into training and testing sets and checking for seasonality in the model

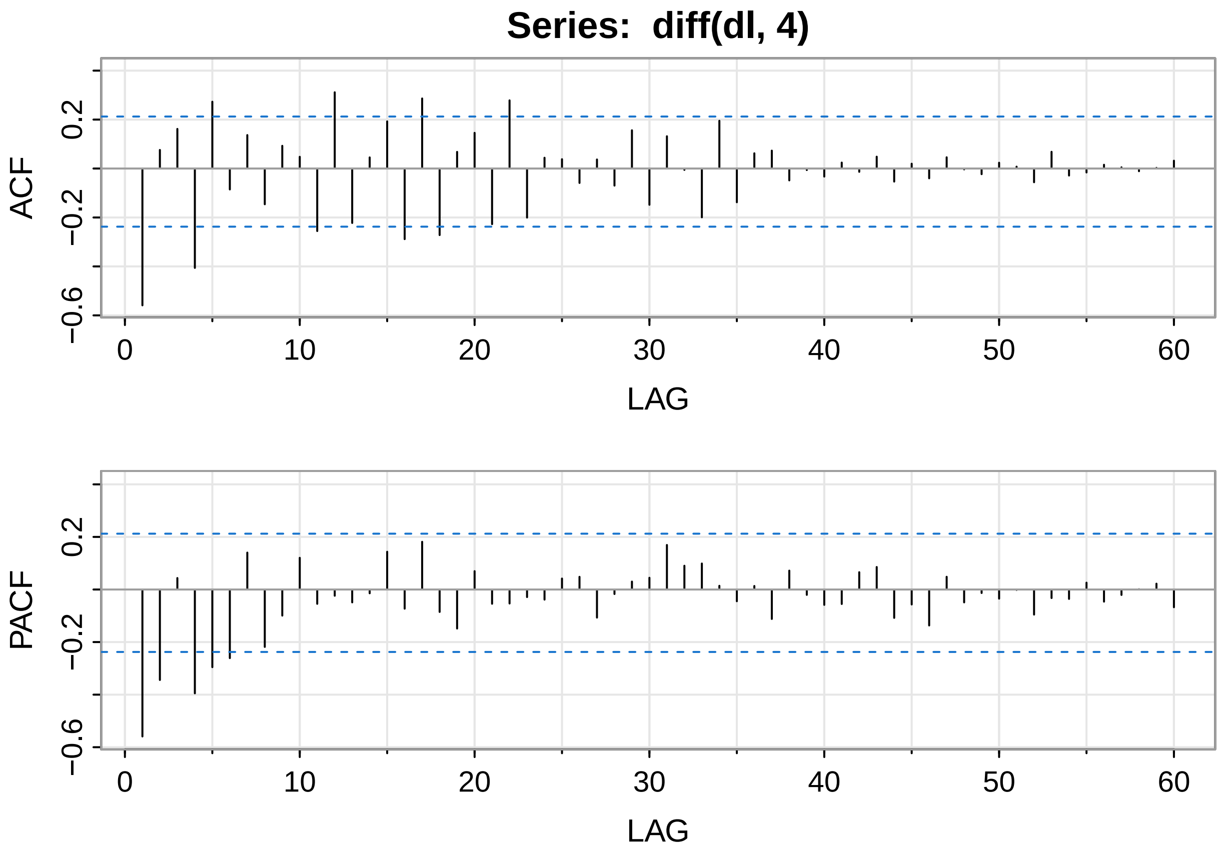
```
x_test <- x[74:83]
x_train <- ts(x[1:73])
acf2(dl, 40)
```




```
##       [,1]  [,2]  [,3] [,4]  [,5]  [,6]  [,7] [,8]  [,9] [,10] [,11] [,12]
## ACF  -0.52  0.10 -0.35 0.52 -0.21 -0.01 -0.18 0.36 -0.17  0.03 -0.25   0.4
## PACF -0.52 -0.23 -0.58 0.02  0.12 -0.05 -0.06 0.17  0.07  0.12 -0.12   0.1
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
## ACF  -0.20 -0.02 -0.08  0.22 -0.04 -0.15 -0.04  0.23 -0.12  0.02 -0.10  0.16
## PACF  0.05 -0.18  0.03 -0.03  0.06 -0.04 -0.14  0.04 -0.12  0.05  0.16 -0.08
##      [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36]
## ACF  -0.02 -0.08  0.00  0.09  0.01 -0.11  0.03  0.12 -0.14  0.01 -0.03  0.17
## PACF  0.12  0.03 -0.03  0.13 -0.03 -0.04  0.07  0.01 -0.09 -0.10 -0.15  0.02
##      [,37] [,38] [,39] [,40]
## ACF  -0.11 -0.04  0.01  0.10
## PACF -0.07 -0.02  0.05 -0.02
```

Applying differencing to account for the seasonality

```r
acf2(diff(dl, 4), 60)
```



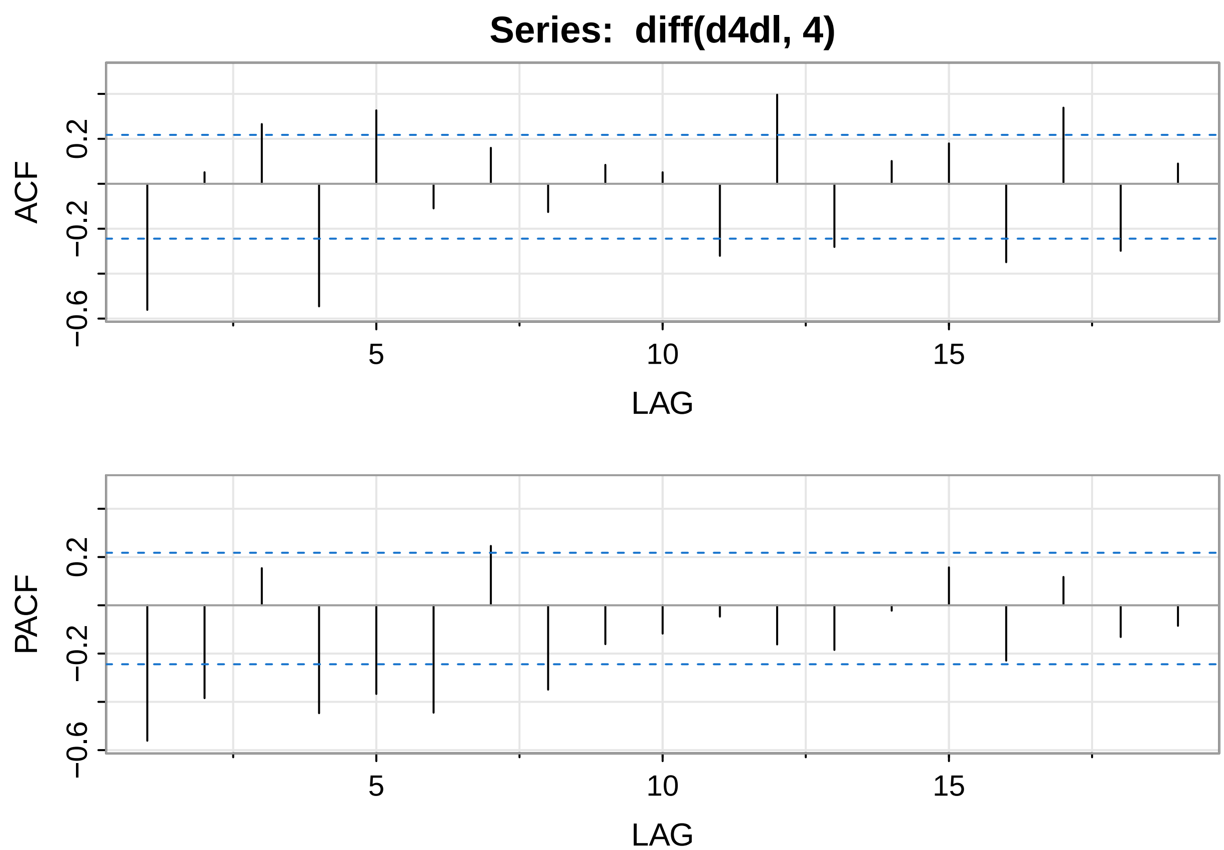

```
##       [,1]  [,2] [,3]  [,4]  [,5]  [,6] [,7]  [,8]  [,9] [,10] [,11] [,12]
## ACF  -0.56  0.08 0.16 -0.41  0.27 -0.09 0.14 -0.15  0.09  0.05 -0.26  0.31
## PACF -0.56 -0.34 0.04 -0.39 -0.30 -0.26 0.14 -0.22 -0.10  0.12 -0.05 -0.02
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
## ACF  -0.22  0.05  0.19 -0.29  0.29 -0.27  0.07  0.15 -0.23  0.28 -0.20  0.04
## PACF -0.05 -0.01  0.14 -0.07  0.18 -0.09 -0.15  0.07 -0.05 -0.05 -0.03 -0.04
##      [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36]
## ACF   0.04 -0.06  0.04 -0.07  0.16 -0.15  0.13 -0.01  -0.2  0.20 -0.14  0.06
## PACF  0.04  0.05 -0.11 -0.02  0.03  0.04  0.17  0.09   0.1  0.01 -0.04  0.01
##      [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48]
## ACF   0.07 -0.05 -0.01 -0.03  0.02 -0.01  0.05 -0.05  0.02 -0.04  0.05  0.00
## PACF -0.11  0.07 -0.02 -0.06 -0.06  0.07  0.09 -0.11 -0.06 -0.14  0.05 -0.05
##      [,49] [,50] [,51] [,52] [,53] [,54] [,55] [,56] [,57] [,58] [,59] [,60]
## ACF  -0.02  0.02  0.01 -0.06  0.07 -0.03 -0.02  0.02  0.00 -0.01  0.00  0.03
## PACF -0.01 -0.03  0.00 -0.10 -0.03 -0.04  0.03 -0.05 -0.02  0.00  0.02 -0.07
```

Updating the series with seasonal differencing

```
d4dl <- diff(dl, 4)
```

Testing again for seasonality in the differenced data

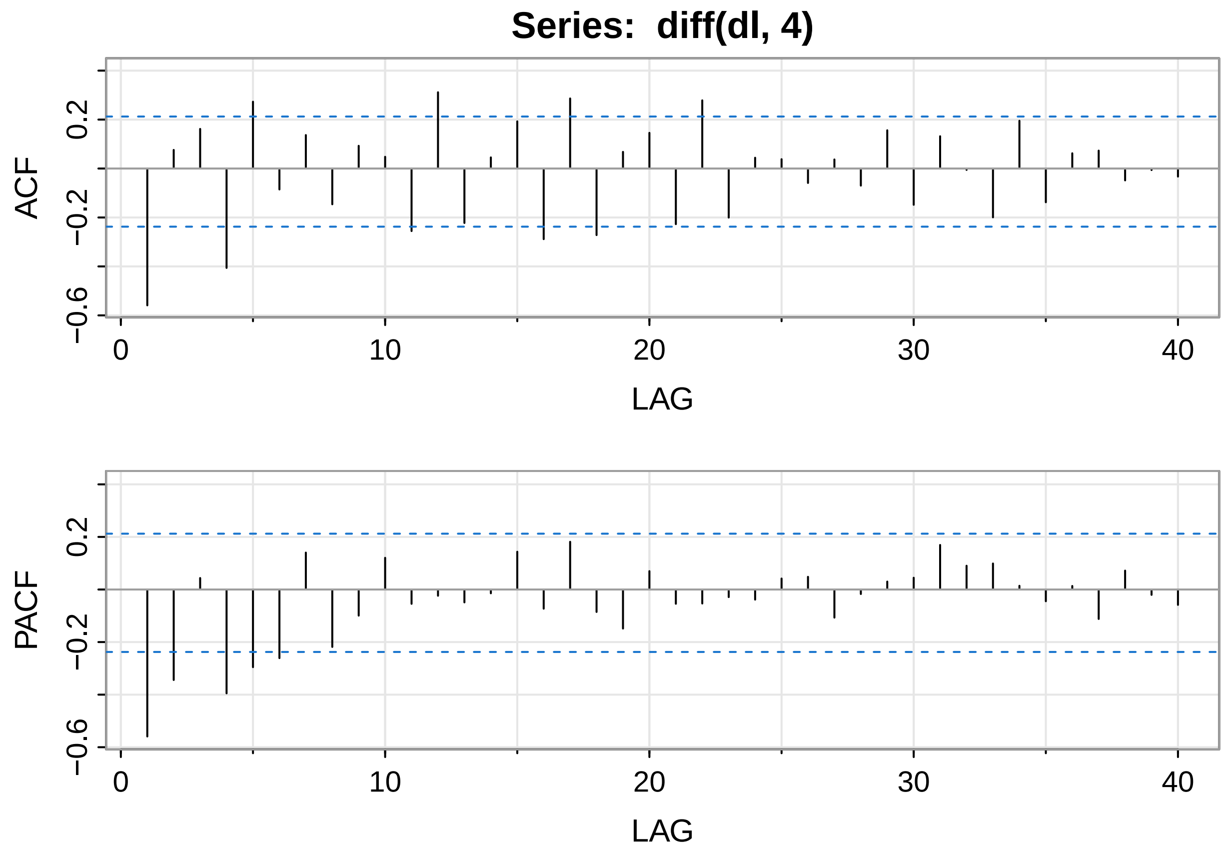
```
acf2(diff(d4dl, 4))
```




```
##       [,1]  [,2] [,3]  [,4]  [,5]  [,6] [,7]  [,8]  [,9] [,10] [,11] [,12]
## ACF  -0.56  0.05 0.27 -0.55  0.33 -0.11 0.16 -0.13  0.08  0.05 -0.32  0.40
## PACF -0.56 -0.38 0.15 -0.45 -0.37 -0.44 0.25 -0.35 -0.16 -0.12 -0.05 -0.16
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19]
## ACF  -0.28  0.10  0.18 -0.35  0.34 -0.30  0.09
## PACF -0.18 -0.02  0.16 -0.23  0.12 -0.13 -0.08
```

Final seasonality transformation: (1-B)(1-B^4)

```
acf2(diff(dl, 4), 40)
```



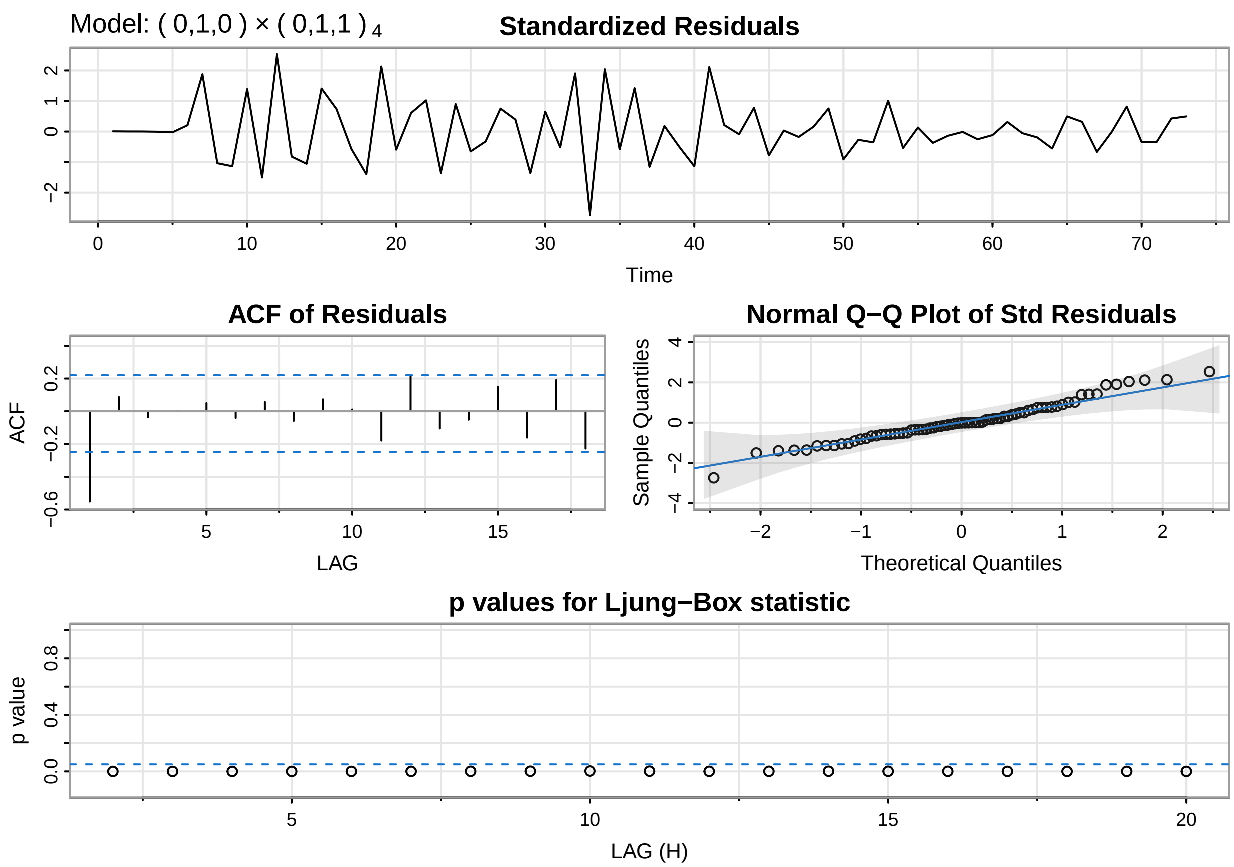

```
##       [,1]  [,2] [,3]  [,4]  [,5]  [,6] [,7]  [,8]  [,9] [,10] [,11] [,12]
## ACF  -0.56  0.08 0.16 -0.41  0.27 -0.09 0.14 -0.15  0.09  0.05 -0.26  0.31
## PACF -0.56 -0.34 0.04 -0.39 -0.30 -0.26 0.14 -0.22 -0.10  0.12 -0.05 -0.02
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
## ACF  -0.22  0.05  0.19 -0.29  0.29 -0.27  0.07  0.15 -0.23  0.28 -0.20  0.04
## PACF -0.05 -0.01  0.14 -0.07  0.18 -0.09 -0.15  0.07 -0.05 -0.05 -0.03 -0.04
##      [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36]
## ACF   0.04 -0.06  0.04 -0.07  0.16 -0.15  0.13 -0.01  -0.2  0.20 -0.14  0.06
## PACF  0.04  0.05 -0.11 -0.02  0.03  0.04  0.17  0.09   0.1  0.01 -0.04  0.01
##      [,37] [,38] [,39] [,40]
## ACF   0.07 -0.05 -0.01 -0.03
## PACF -0.11  0.07 -0.02 -0.06
```

Testing SARIMA models

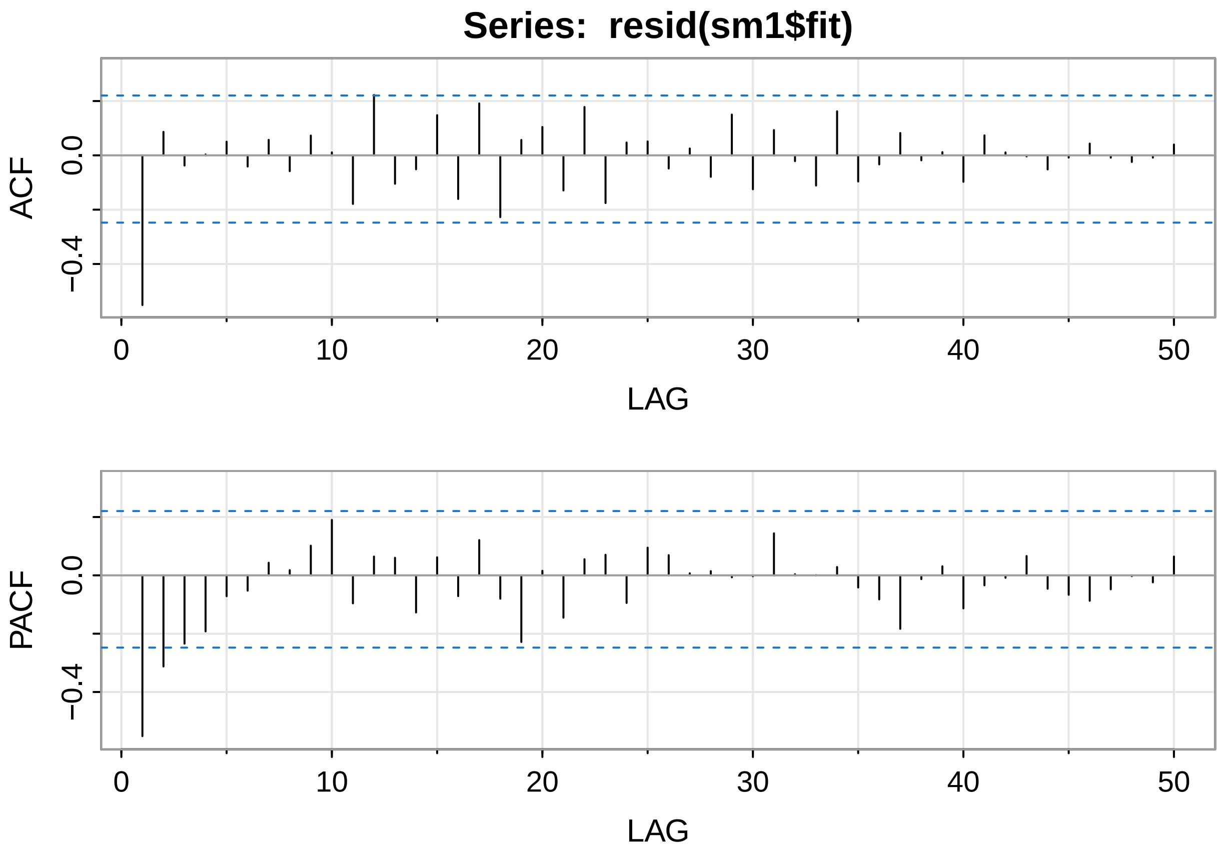
```r
sm1 <- sarima(log(x_train), 0, 1, 0, 0, 1, 1, 4)
```

```
## initial  value -1.549560
## iter   2 value -1.688684
## iter   3 value -1.700887
## iter   4 value -1.705703
## iter   5 value -1.706131
## iter   6 value -1.706766
## iter   7 value -1.706770
## iter   7 value -1.706770
```

```
## final  value -1.706770
## converged
## initial  value -1.713264
## iter   2 value -1.715059
## iter   3 value -1.715082
## iter   3 value -1.715082
## iter   3 value -1.715082
## final  value -1.715082
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##      Estimate     SE t.value p.value
## sma1  -0.6447 0.1131  -5.699       0
##
## sigma^2 estimated as 0.03137438 on 67 degrees of freedom
##
## AIC = -0.5334637  AICc = -0.5325725  BIC = -0.4681841
##
```




```r
acf2(resid(sm1$fit), 50)
```

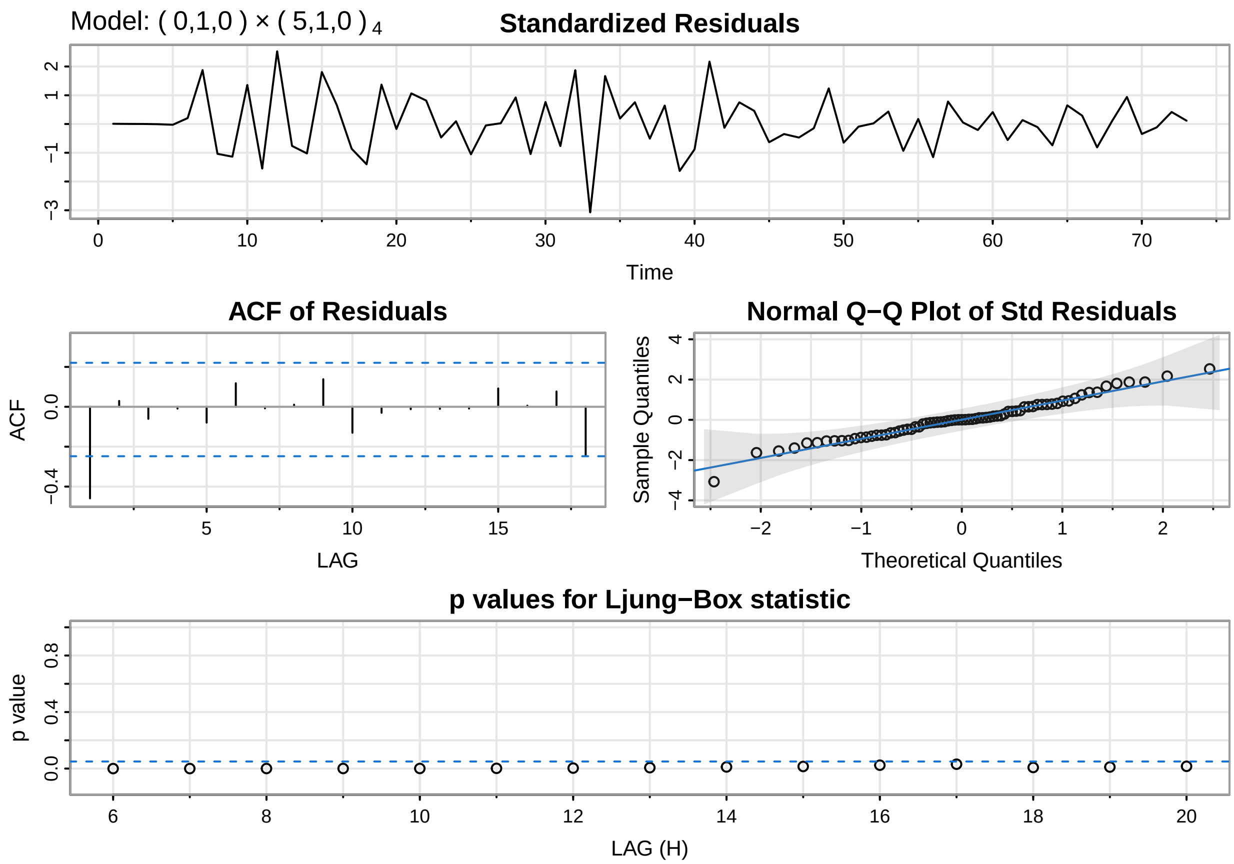



```
##       [,1]  [,2]  [,3]  [,4]  [,5]  [,6] [,7]  [,8] [,9] [,10] [,11] [,12]
## ACF  -0.55  0.09 -0.04  0.00  0.05 -0.04 0.06 -0.06 0.07  0.01 -0.18  0.22
## PACF -0.55 -0.31 -0.23 -0.19 -0.07 -0.05 0.04  0.02 0.10  0.19 -0.10  0.06
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
## ACF  -0.10 -0.05  0.15 -0.16  0.19 -0.23  0.06  0.10 -0.13  0.18 -0.18  0.05
## PACF  0.06 -0.13  0.06 -0.07  0.12 -0.08 -0.23  0.02 -0.15  0.06  0.07 -0.09
##      [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36]
## ACF   0.05 -0.05  0.03 -0.08  0.15 -0.12  0.09 -0.02 -0.11  0.16 -0.10 -0.03
## PACF  0.09  0.07  0.01  0.01 -0.01  0.00  0.14  0.00  0.00  0.03 -0.04 -0.08
##      [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48]
## ACF   0.08 -0.02  0.01 -0.10  0.07  0.01  0.00 -0.05 -0.01  0.04 -0.01 -0.02
## PACF -0.18 -0.01  0.03 -0.11 -0.03 -0.01  0.07 -0.05 -0.07 -0.09 -0.05  0.00
##      [,49] [,50]
## ACF  -0.01  0.04
## PACF -0.02  0.06
```

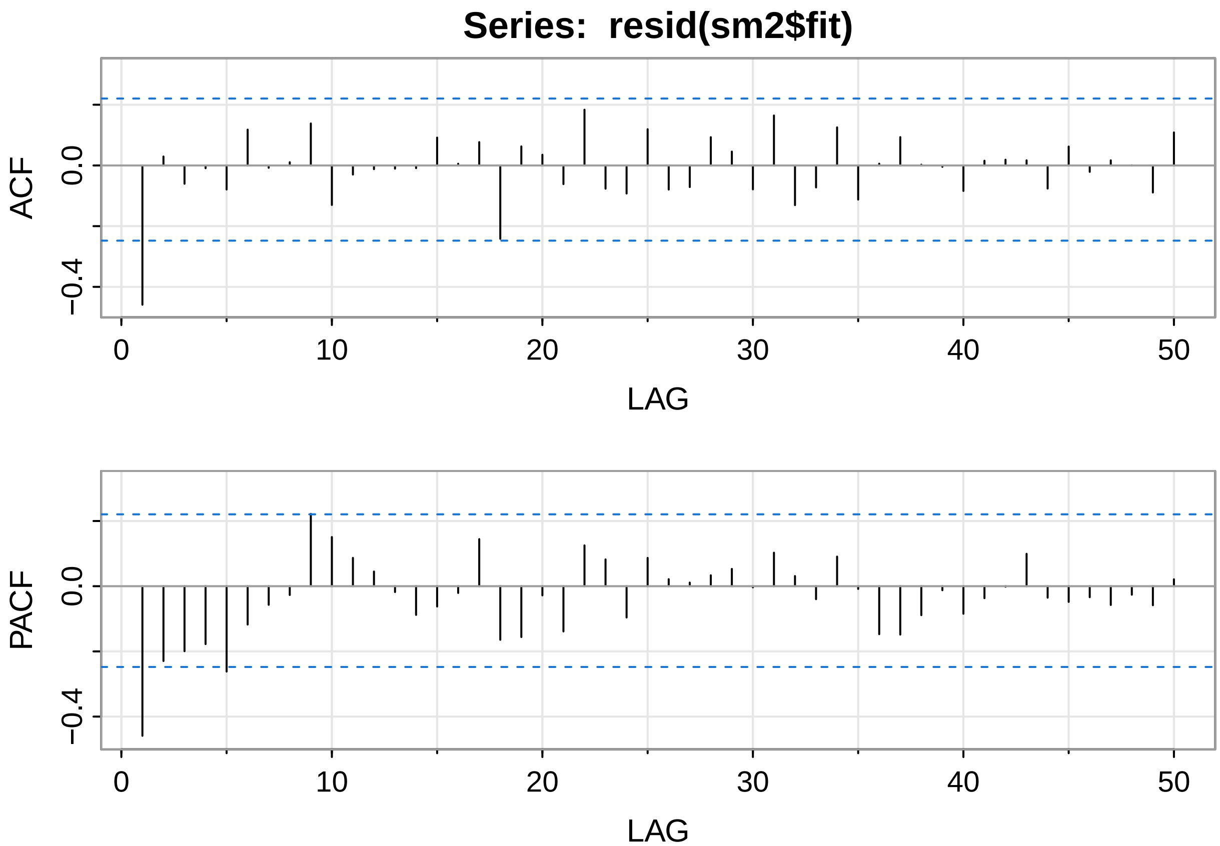
```
sm2 <- sarima(log(x_train), 0, 1, 0, 5, 1, 0, 4)
```

```
## initial  value -1.796921
## iter   2 value -1.913217
## iter   3 value -1.938166
## iter   4 value -1.954542
## iter   5 value -1.968833
## iter   6 value -1.969968
## iter   7 value -1.970123
```

```
## iter   8 value -1.970125
## iter   8 value -1.970125
## iter   8 value -1.970125
## final  value -1.970125
## converged
## initial  value -1.751264
## iter   2 value -1.785966
## iter   3 value -1.785973
## iter   4 value -1.786378
## iter   5 value -1.786387
## iter   6 value -1.786388
## iter   6 value -1.786388
## iter   6 value -1.786388
## final  value -1.786388
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##      Estimate     SE t.value p.value
## sar1  -0.4862 0.1278 -3.8050  0.0003
## sar2  -0.4566 0.1400 -3.2620  0.0018
## sar3   0.0193 0.1551  0.1247  0.9011
## sar4  -0.2405 0.1422 -1.6918  0.0956
## sar5   0.0805 0.1344  0.5991  0.5512
##
## sigma^2 estimated as 0.02640218 on 63 degrees of freedom
##
## AIC = -0.558428  AICc = -0.5441965  BIC = -0.3625891
##
```




```
acf2(resid(sm2$fit), 50)
```



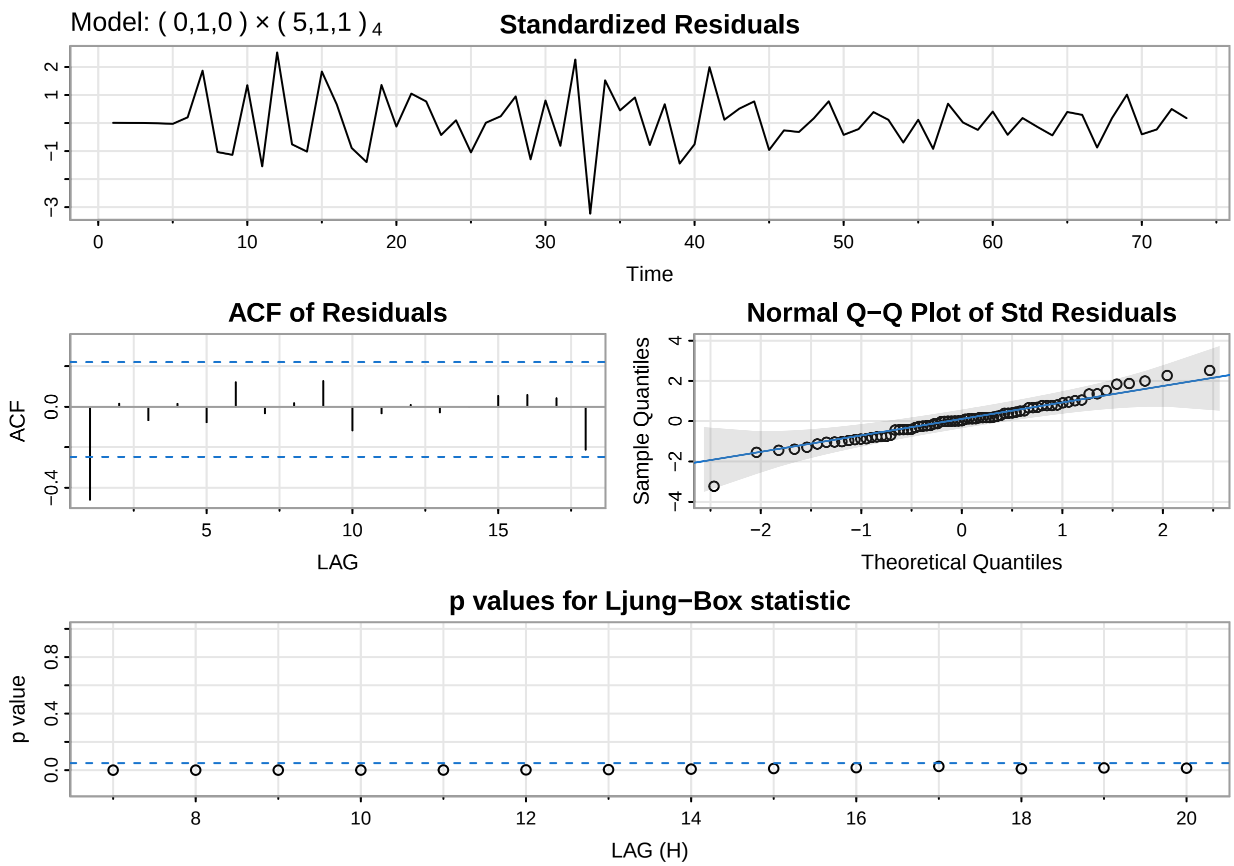

```
##       [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8] [,9] [,10] [,11] [,12]
## ACF  -0.46  0.03 -0.06 -0.01 -0.08  0.12 -0.01  0.01 0.14 -0.13 -0.03 -0.01
## PACF -0.46 -0.23 -0.20 -0.18 -0.26 -0.12 -0.06 -0.03 0.22  0.15  0.09  0.04
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
## ACF  -0.01 -0.01  0.09  0.01  0.08 -0.24  0.06  0.04 -0.06  0.18 -0.08 -0.09
## PACF -0.02 -0.09 -0.06 -0.02  0.14 -0.16 -0.16 -0.03 -0.14  0.13  0.08 -0.10
##      [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36]
## ACF   0.12 -0.08 -0.07  0.09  0.05 -0.08  0.16 -0.13 -0.07  0.13 -0.11  0.01
## PACF  0.09  0.02  0.01  0.03  0.05  0.00  0.10  0.03 -0.04  0.09 -0.01 -0.15
##      [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48]
## ACF   0.09  0.00  0.00 -0.08  0.02  0.02  0.02 -0.08  0.06 -0.02  0.02  0.00
## PACF -0.15 -0.09 -0.01 -0.08 -0.04  0.00  0.10 -0.04 -0.05 -0.03 -0.06 -0.03
##      [,49] [,50]
## ACF  -0.09  0.11
## PACF -0.06  0.02
```

```
sm3 <- sarima(log(x_train), 0, 1, 0, 5, 1, 1, 4)
```

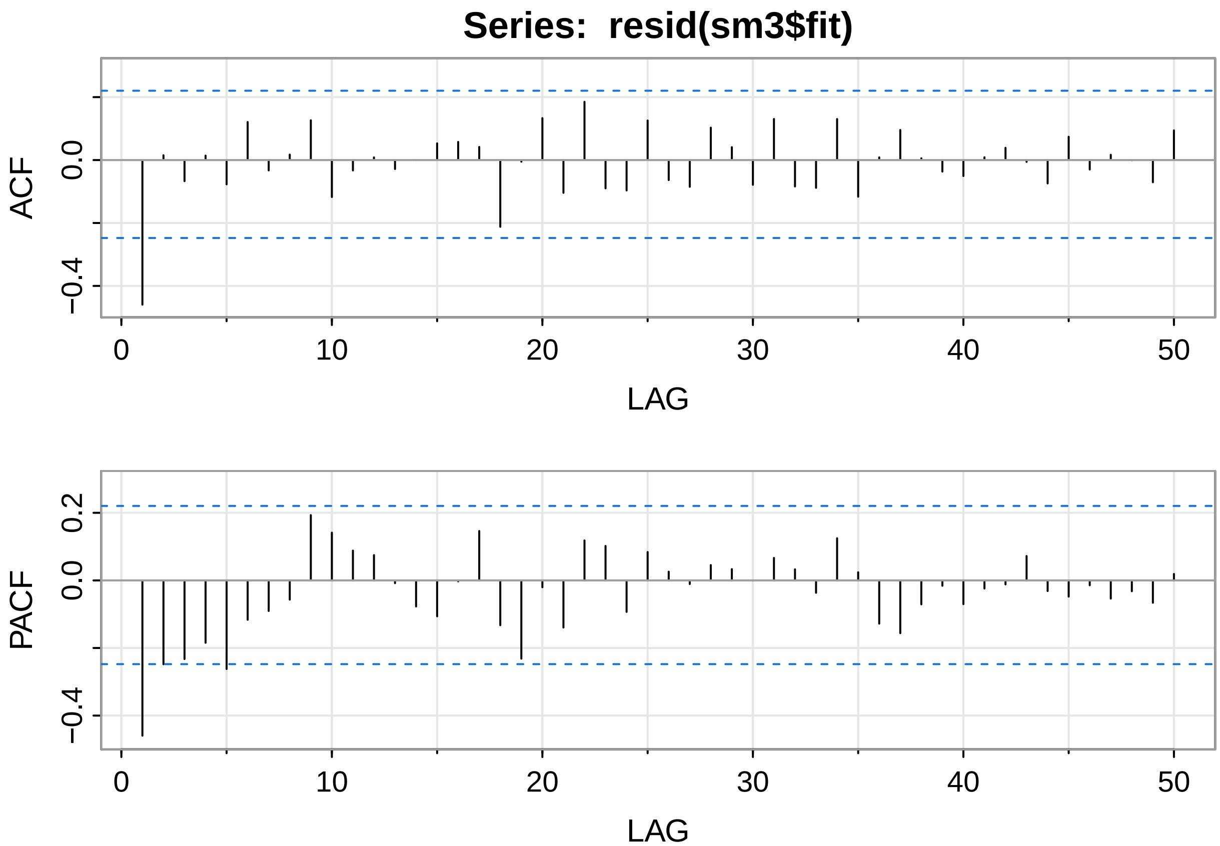
```
## initial  value -1.796921
## iter   2 value -1.924965
## iter   3 value -1.950340
## iter   4 value -1.960040
## iter   5 value -1.978997
## iter   6 value -1.982129
## iter   7 value -2.008565
```

```
## iter   8 value -2.011926
## iter   9 value -2.014116
## iter  10 value -2.014285
## iter  11 value -2.015657
## iter  12 value -2.016107
## iter  13 value -2.017176
## iter  14 value -2.018302
## iter  15 value -2.025281
## iter  16 value -2.027908
## iter  17 value -2.029325
## iter  18 value -2.030016
## iter  19 value -2.030537
## iter  20 value -2.030584
## iter  21 value -2.030587
## iter  22 value -2.030589
## iter  23 value -2.030589
## iter  23 value -2.030589
## iter  23 value -2.030589
## final  value -2.030589
## converged
## initial  value -1.764107
## iter   2 value -1.778126
## iter   3 value -1.782119
## iter   4 value -1.784040
## iter   5 value -1.784187
## iter   6 value -1.784265
## iter   7 value -1.785665
## iter   8 value -1.785907
## iter   9 value -1.787677
## iter  10 value -1.789661
## iter  11 value -1.790160
## iter  12 value -1.790366
## iter  13 value -1.790665
## iter  14 value -1.790824
## iter  15 value -1.790830
## iter  16 value -1.790875
## iter  17 value -1.790880
## iter  18 value -1.790895
## iter  19 value -1.790900
## iter  20 value -1.790903
## iter  21 value -1.790904
## iter  21 value -1.790904
## iter  21 value -1.790904
## final  value -1.790904
## converged
## <><><><><><><><><><><><><><><>
##
## Coefficients:
##      Estimate     SE t.value p.value
## sar1   0.4123 0.1404  2.9376  0.0046
## sar2   0.0074 0.1394  0.0531  0.9578
## sar3   0.4044 0.1313  3.0788  0.0031
## sar4  -0.2814 0.1355 -2.0757  0.0421
## sar5   0.2099 0.1467  1.4302  0.1577
```

```
## sma1  -1.0000 0.1124 -8.8970  0.0000
##
## sigma^2 estimated as 0.02502072 on 62 degrees of freedom
##
## AIC = -0.538048  AICc = -0.5177972  BIC = -0.3095692
##
```




```
acf2(resid(sm3$fit), 50)
```




```
##       [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8] [,9] [,10] [,11] [,12]
## ACF  -0.46  0.02 -0.07  0.01 -0.08  0.12 -0.03  0.02 0.13 -0.12 -0.03  0.01
## PACF -0.46 -0.25 -0.23 -0.18 -0.26 -0.12 -0.09 -0.06 0.19  0.14  0.09  0.07
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
## ACF  -0.03  0.00  0.05  0.06  0.04 -0.21 -0.01  0.13 -0.10  0.19 -0.09 -0.10
## PACF -0.01 -0.08 -0.11  0.00  0.15 -0.13 -0.23 -0.02 -0.14  0.12  0.10 -0.09
##      [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36]
## ACF   0.13 -0.06 -0.09  0.10  0.04 -0.08  0.13 -0.08 -0.09  0.13 -0.12  0.01
## PACF  0.08  0.03 -0.01  0.05  0.03  0.00  0.07  0.03 -0.04  0.12  0.02 -0.13
##      [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48]
## ACF   0.10  0.01 -0.04 -0.05  0.01  0.04 -0.01 -0.07  0.07 -0.03  0.02  0.00
## PACF -0.16 -0.07 -0.02 -0.07 -0.02 -0.01  0.07 -0.03 -0.05 -0.01 -0.05 -0.03
##      [,49] [,50]
## ACF  -0.07  0.09
## PACF -0.07  0.02
```

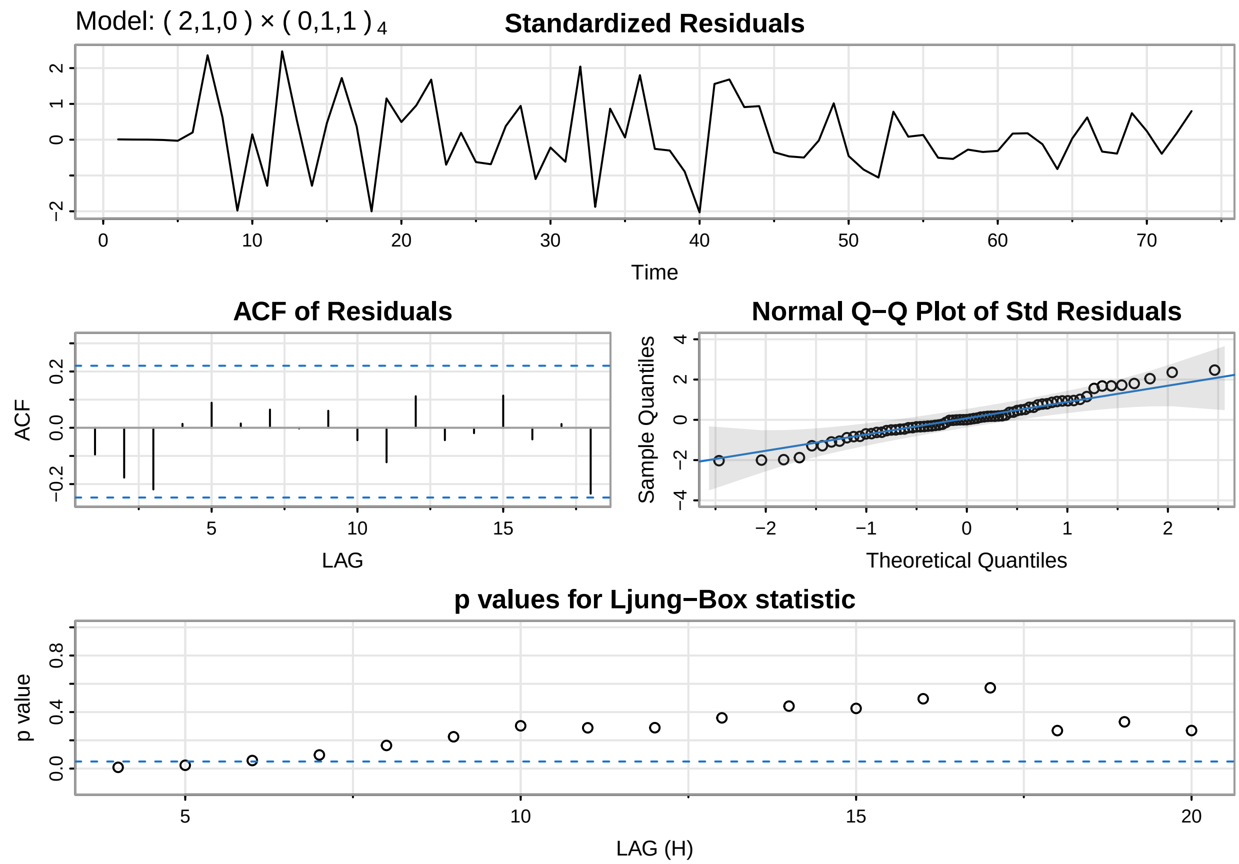
I will proceed with SARIMA (0,1,0)(0,1,1)^4 since it reduced the most significant terms and is simpler.

c) Model Diagnostics and Comparison: We will create some more specific models to select the best one.

```
# Testing AR(2), MA(1) and ARMA(2,1)
m1 <- sarima(log(x_train), 2, 1, 0, 0, 1, 1, 4)
```

```
## initial  value -1.561111
```

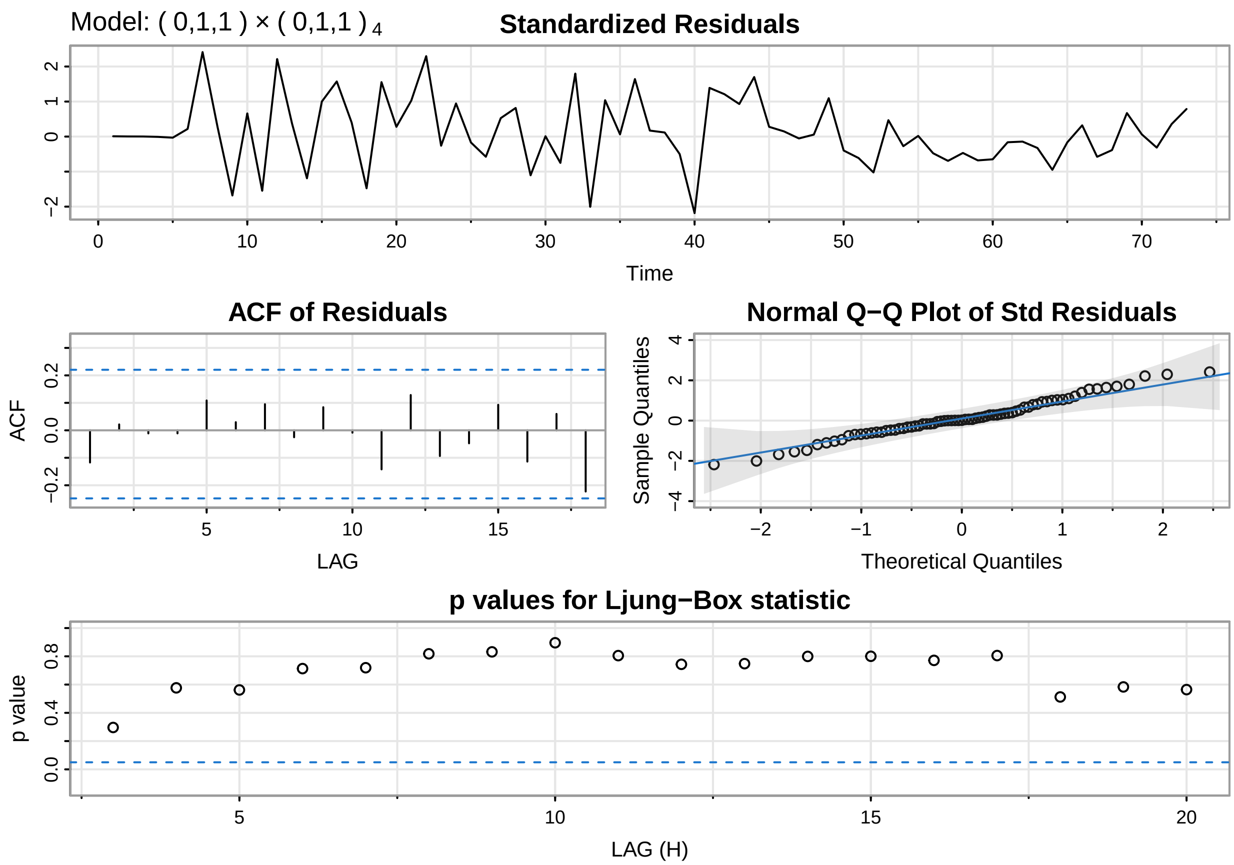
```
## iter   2 value -1.852658
## iter   3 value -1.892416
## iter   4 value -1.933203
## iter   5 value -1.938998
## iter   6 value -1.939078
## iter   7 value -1.939080
## iter   8 value -1.939081
## iter   8 value -1.939081
## final  value -1.939081
## converged
## initial  value -1.925506
## iter   2 value -1.929571
## iter   3 value -1.929661
## iter   4 value -1.929665
## iter   4 value -1.929665
## iter   4 value -1.929665
## final  value -1.929665
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##      Estimate     SE t.value p.value
## ar1   -0.7028 0.1177 -5.9730  0.0000
## ar2   -0.2957 0.1196 -2.4720  0.0161
## sma1  -0.5807 0.1173 -4.9493  0.0000
##
## sigma^2 estimated as 0.02036151 on 65 degrees of freedom
##
## AIC = -0.9038051  AICc = -0.8982904  BIC = -0.7732459
##
```




```r
m2 <- sarima(log(x_train), 0, 1, 1, 0, 1, 1, 4)
```

```
## initial  value -1.549560
## iter   2 value -1.916364
## iter   3 value -1.948709
## iter   4 value -1.952101
## iter   5 value -1.953629
## iter   6 value -1.953902
## iter   7 value -1.953911
## iter   8 value -1.953912
## iter   8 value -1.953912
## iter   8 value -1.953912
## final  value -1.953912
## converged
## initial  value -1.969244
## iter   2 value -1.974837
## iter   3 value -1.977712
## iter   4 value -1.978135
## iter   5 value -1.978146
## iter   6 value -1.978146
## iter   6 value -1.978146
## final  value -1.978146
## converged
## <><><><><><><><><><><><><><>
##
```

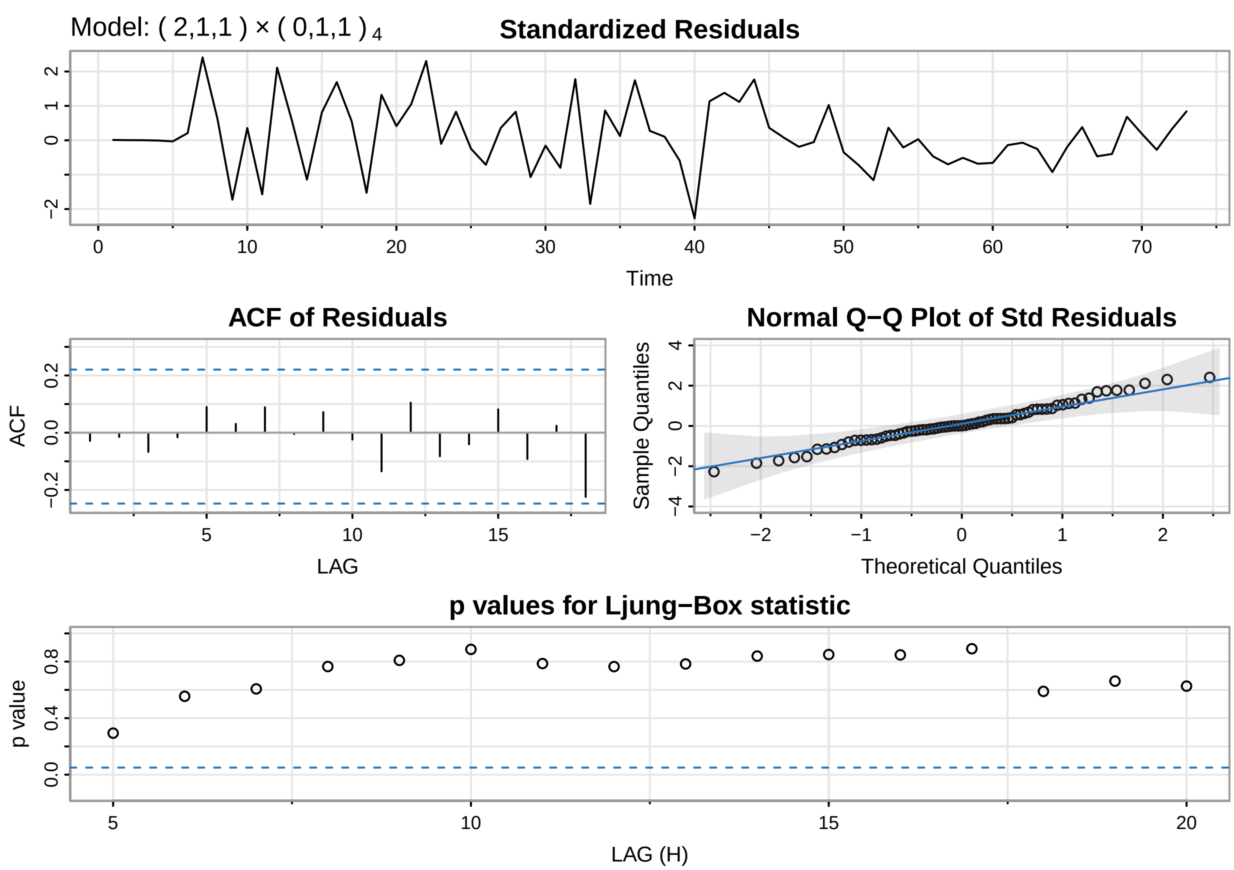
```
## Coefficients:
##      Estimate     SE t.value p.value
## ma1   -0.7862 0.0804 -9.7737       0
## sma1  -0.5609 0.1262 -4.4454       0
##
## sigma^2 estimated as 0.01832007 on 66 degrees of freedom
##
## AIC = -1.03018  AICc = -1.027465  BIC = -0.932261
##
```




```r
m3 <- sarima(log(x_train), 2, 1, 1, 0, 1, 1, 4)
```

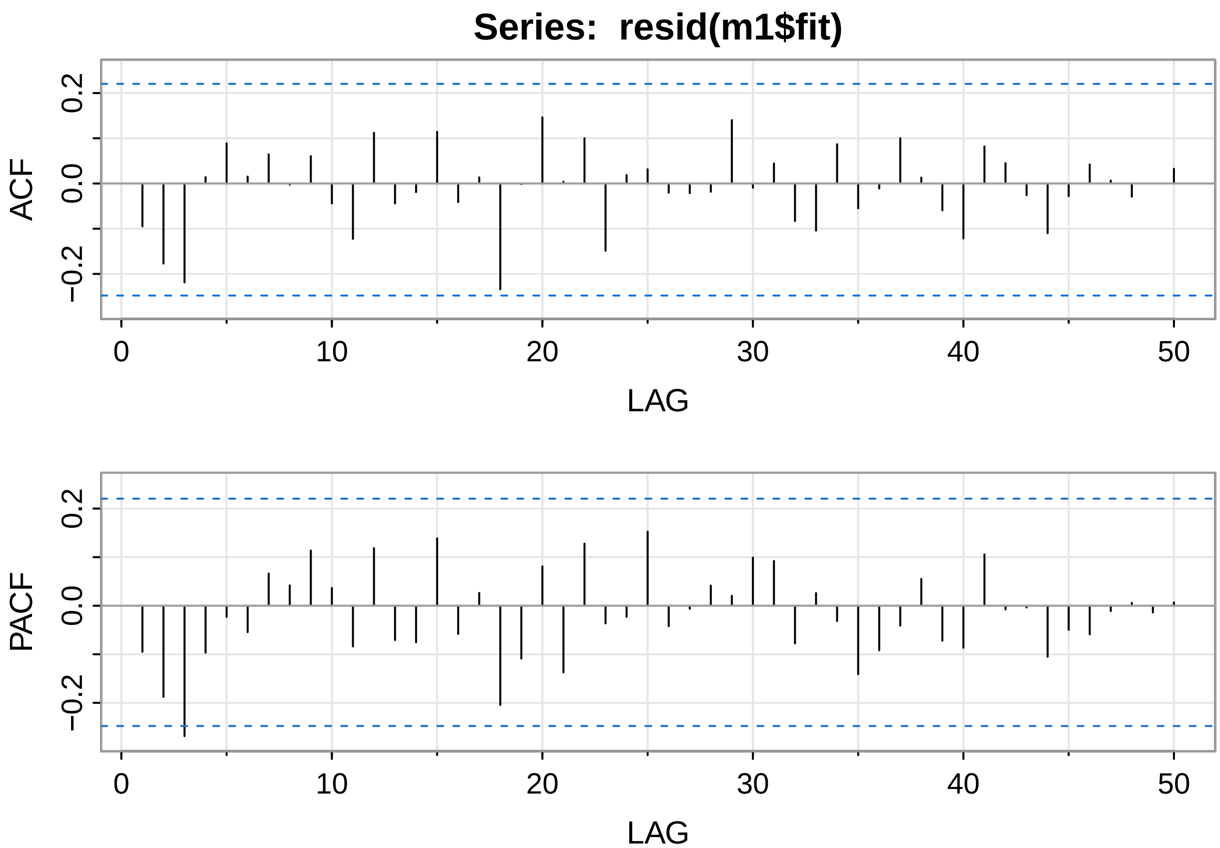
```
## initial  value -1.561111
## iter   2 value -1.906852
## iter   3 value -1.985286
## iter   4 value -1.993541
## iter   5 value -2.000147
## iter   6 value -2.000298
## iter   7 value -2.000317
## iter   8 value -2.000319
## iter   8 value -2.000319
## iter   8 value -2.000319
## final  value -2.000319
## converged
## initial  value -1.975230
```

```
## iter   2 value -1.979983
## iter   3 value -1.980683
## iter   4 value -1.984793
## iter   5 value -1.985278
## iter   6 value -1.985326
## iter   7 value -1.985327
## iter   7 value -1.985327
## iter   7 value -1.985327
## final  value -1.985327
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##      Estimate     SE t.value p.value
## ar1   -0.1698 0.1735 -0.9786  0.3315
## ar2   -0.0501 0.1575 -0.3185  0.7511
## ma1   -0.7076 0.1350 -5.2418  0.0000
## sma1  -0.5765 0.1241 -4.6437  0.0000
##
## sigma^2 estimated as 0.01806681 on 64 degrees of freedom
##
## AIC = -0.9857185  AICc = -0.9763814  BIC = -0.8225194
##
```



Performing residual diagnostics for model m1

```r
acf2(resid(m1$fit), 50)
```



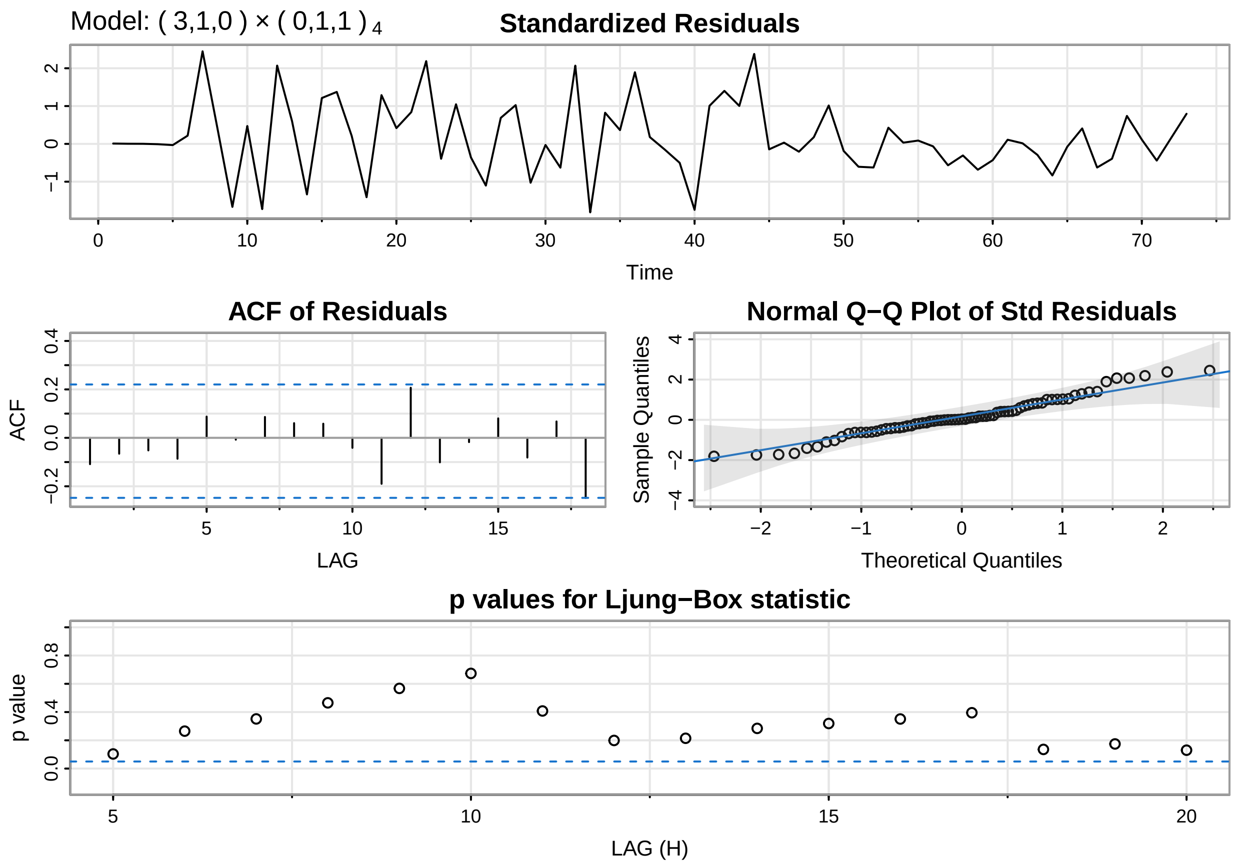

```
##       [,1]  [,2]  [,3]  [,4]  [,5]  [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  -0.09 -0.18 -0.22  0.01  0.09  0.02 0.06 0.00 0.06 -0.04 -0.12  0.11 -0.04
## PACF -0.09 -0.19 -0.27 -0.10 -0.02 -0.05 0.07 0.04 0.11  0.04 -0.08  0.12 -0.07
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
## ACF  -0.02  0.11 -0.04  0.01 -0.23  0.00  0.15  0.00  0.10 -0.15  0.02  0.03
## PACF -0.08  0.14 -0.06  0.03 -0.20 -0.11  0.08 -0.14  0.13 -0.04 -0.02  0.15
##      [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37]
## ACF  -0.02 -0.02 -0.02  0.14 -0.01  0.04 -0.08 -0.10  0.09 -0.05 -0.01  0.10
## PACF -0.04 -0.01  0.04  0.02  0.10  0.09 -0.08  0.03 -0.03 -0.14 -0.09 -0.04
##      [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48] [,49]
## ACF   0.01 -0.06 -0.12  0.08  0.04 -0.03 -0.11 -0.03  0.04  0.01 -0.03  0.00
## PACF  0.06 -0.07 -0.09  0.11 -0.01  0.00 -0.11 -0.05 -0.06 -0.01  0.01 -0.01
##      [,50]
## ACF   0.03
## PACF  0.01
```

Adding AR(3)

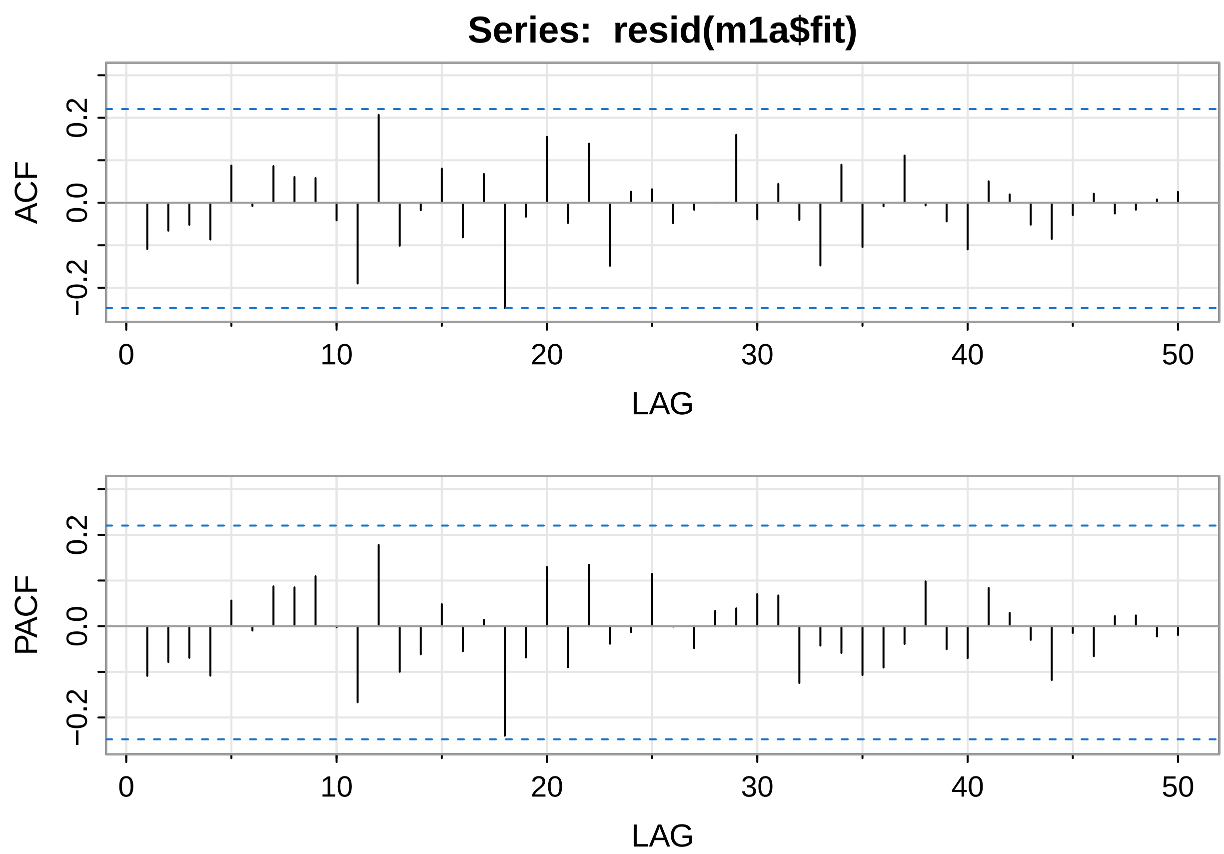
```r
m1a <- sarima(log(x_train), 3, 1, 0, 0, 1, 1, 4)
```

```
## initial  value -1.561750
## iter   2 value -1.851798
```

```
## iter   3 value -1.898228
## iter   4 value -1.977155
## iter   5 value -1.980806
## iter   6 value -1.981260
## iter   7 value -1.981375
## iter   8 value -1.981383
## iter   9 value -1.981384
## iter   9 value -1.981384
## iter   9 value -1.981384
## final  value -1.981384
## converged
## initial  value -1.958418
## iter   2 value -1.964839
## iter   3 value -1.967125
## iter   4 value -1.967961
## iter   5 value -1.968203
## iter   6 value -1.968373
## iter   7 value -1.968411
## iter   8 value -1.968413
## iter   8 value -1.968413
## iter   8 value -1.968413
## final  value -1.968413
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##      Estimate     SE t.value p.value
## ar1   -0.8060 0.1190 -6.7718  0.0000
## ar2   -0.5711 0.1720 -3.3199  0.0015
## ar3   -0.4162 0.2369 -1.7571  0.0837
## sma1  -0.8402 0.2863 -2.9347  0.0046
##
## sigma^2 estimated as 0.01835465 on 64 degrees of freedom
##
## AIC = -0.9518896  AICc = -0.9425525  BIC = -0.7886905
##
```



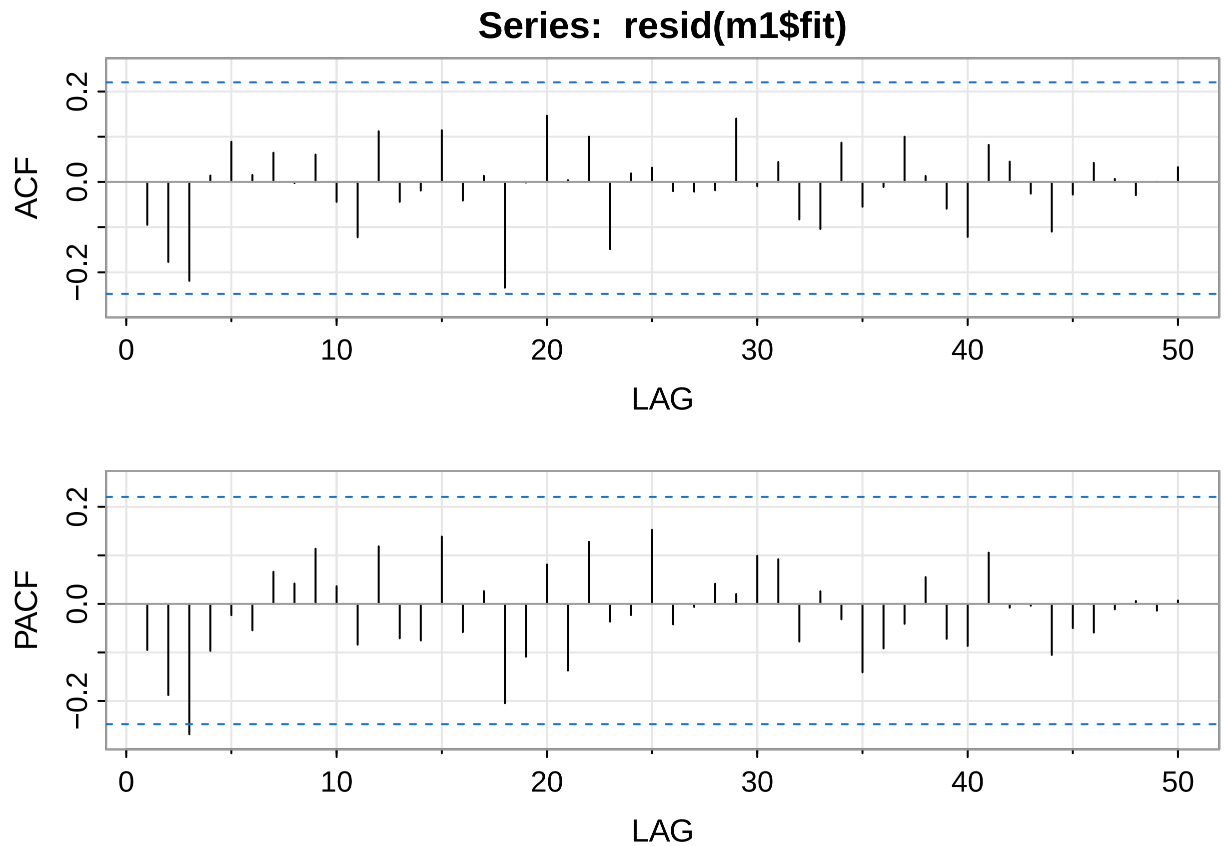

```r
acf2(resid(m1a$fit), 50)
```




```
##       [,1]  [,2]  [,3]  [,4] [,5]  [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  -0.11 -0.07 -0.05 -0.09 0.09 -0.01 0.09 0.06 0.06 -0.04 -0.19  0.21  -0.1
## PACF -0.11 -0.08 -0.07 -0.11 0.06 -0.01 0.09 0.09 0.11  0.00 -0.17  0.18  -0.1
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
## ACF  -0.02  0.08 -0.08  0.07 -0.25 -0.03  0.15 -0.05  0.14 -0.15  0.03  0.03
## PACF -0.06  0.05 -0.05  0.01 -0.24 -0.07  0.13 -0.09  0.13 -0.04 -0.01  0.11
##      [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37]
## ACF  -0.05 -0.02  0.00  0.16 -0.04  0.04 -0.04 -0.15  0.09 -0.10 -0.01  0.11
## PACF  0.00 -0.05  0.03  0.04  0.07  0.07 -0.12 -0.04 -0.06 -0.11 -0.09 -0.04
##      [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48] [,49]
## ACF  -0.01 -0.04 -0.11  0.05  0.02 -0.05 -0.08 -0.03  0.02 -0.03 -0.02  0.01
## PACF  0.10 -0.05 -0.07  0.08  0.03 -0.03 -0.12 -0.01 -0.07  0.02  0.02 -0.02
##      [,50]
## ACF   0.03
## PACF -0.02
```

Removing AR(3) as it is not significant

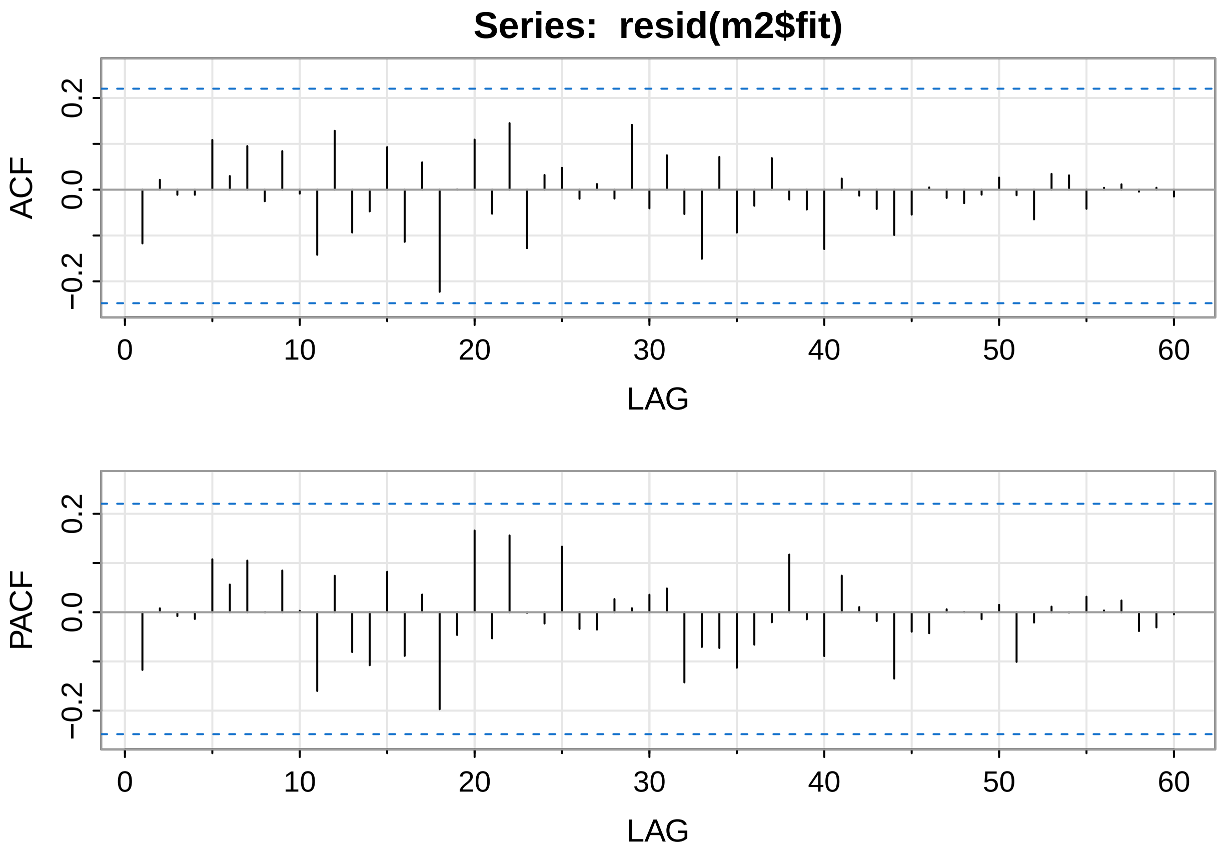
```r
acf2(resid(m1$fit), 50)
```




```
##       [,1]  [,2]  [,3]  [,4]  [,5]  [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  -0.09 -0.18 -0.22  0.01  0.09  0.02 0.06 0.00 0.06 -0.04 -0.12  0.11 -0.04
## PACF -0.09 -0.19 -0.27 -0.10 -0.02 -0.05 0.07 0.04 0.11  0.04 -0.08  0.12 -0.07
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
## ACF  -0.02  0.11 -0.04  0.01 -0.23  0.00  0.15  0.00  0.10 -0.15  0.02  0.03
## PACF -0.08  0.14 -0.06  0.03 -0.20 -0.11  0.08 -0.14  0.13 -0.04 -0.02  0.15
##      [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37]
## ACF  -0.02 -0.02 -0.02  0.14 -0.01  0.04 -0.08 -0.10  0.09 -0.05 -0.01  0.10
## PACF -0.04 -0.01  0.04  0.02  0.10  0.09 -0.08  0.03 -0.03 -0.14 -0.09 -0.04
##      [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48] [,49]
## ACF   0.01 -0.06 -0.12  0.08  0.04 -0.03 -0.11 -0.03  0.04  0.01 -0.03  0.00
## PACF  0.06 -0.07 -0.09  0.11 -0.01  0.00 -0.11 -0.05 -0.06 -0.01  0.01 -0.01
##      [,50]
## ACF   0.03
## PACF  0.01
```

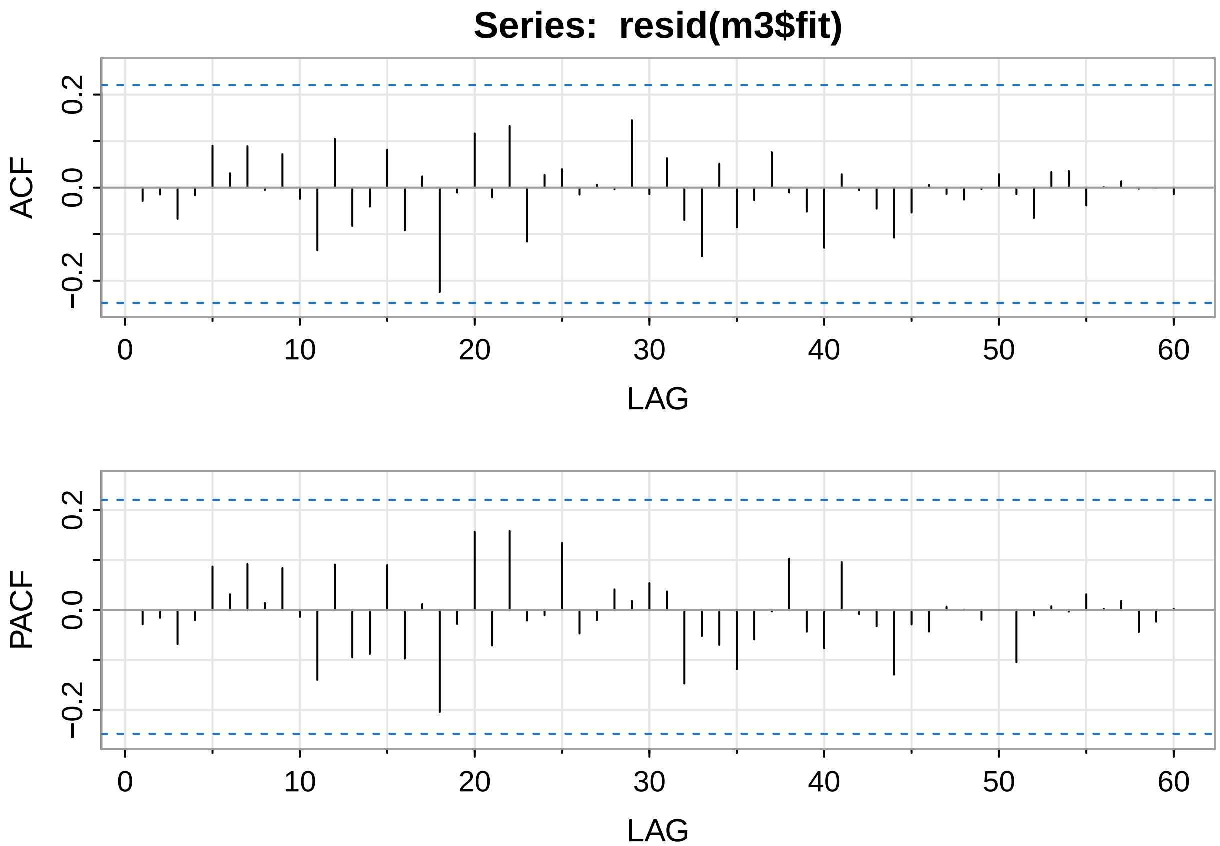
Residual diagnostics for m2 and m3

```
acf2(resid(m2$fit), 60)
```




```
##       [,1] [,2]  [,3]  [,4] [,5] [,6] [,7]  [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  -0.12 0.02 -0.01 -0.01 0.11 0.03  0.1 -0.03 0.08 -0.01 -0.14  0.13 -0.09
## PACF -0.12 0.01 -0.01 -0.01 0.11 0.06  0.1  0.00 0.08  0.00 -0.16  0.07 -0.08
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
## ACF  -0.05  0.09 -0.11  0.06 -0.22  0.00  0.11 -0.05  0.15 -0.13  0.03  0.05
## PACF -0.11  0.08 -0.09  0.04 -0.20 -0.05  0.17 -0.05  0.16  0.00 -0.02  0.13
##      [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37]
## ACF  -0.02  0.01 -0.02  0.14 -0.04  0.08 -0.05 -0.15  0.07 -0.09 -0.04  0.07
## PACF -0.03 -0.04  0.03  0.01  0.04  0.05 -0.14 -0.07 -0.07 -0.11 -0.07 -0.02
##      [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48] [,49]
## ACF  -0.02 -0.04 -0.13  0.02 -0.01 -0.04 -0.10 -0.05  0.00 -0.02 -0.03 -0.01
## PACF  0.12 -0.01 -0.09  0.07  0.01 -0.02 -0.13 -0.04 -0.04  0.01  0.00 -0.01
##      [,50] [,51] [,52] [,53] [,54] [,55] [,56] [,57] [,58] [,59] [,60]
## ACF   0.03 -0.01 -0.06  0.03  0.03 -0.04     0  0.01  0.00  0.00 -0.01
## PACF  0.02 -0.10 -0.02  0.01  0.00  0.03     0  0.02 -0.04 -0.03  0.00
```

```
acf2(resid(m3$fit), 60)
```




```
##        [,1]  [,2]  [,3]  [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  -0.03 -0.01 -0.07 -0.02 0.09 0.03 0.09 0.00 0.07 -0.02 -0.14  0.11 -0.08
## PACF -0.03 -0.02 -0.07 -0.02 0.09 0.03 0.09 0.01 0.08 -0.01 -0.14  0.09 -0.09
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
## ACF  -0.04  0.08 -0.09  0.02 -0.22 -0.01  0.12 -0.02  0.13 -0.12  0.03  0.04
## PACF -0.09  0.09 -0.10  0.01 -0.20 -0.03  0.16 -0.07  0.16 -0.02 -0.01  0.13
##      [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37]
## ACF  -0.02  0.01  0.00  0.14 -0.01  0.06 -0.07 -0.15  0.05 -0.09 -0.03  0.08
## PACF -0.05 -0.02  0.04  0.02  0.05  0.04 -0.15 -0.05 -0.07 -0.12 -0.06  0.00
##      [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48] [,49]
## ACF  -0.01 -0.05 -0.13  0.03 -0.01 -0.05 -0.11 -0.05  0.01 -0.01 -0.03  0.00
## PACF  0.10 -0.04 -0.08  0.10 -0.01 -0.03 -0.13 -0.03 -0.04  0.01  0.00 -0.02
##      [,50] [,51] [,52] [,53] [,54] [,55] [,56] [,57] [,58] [,59] [,60]
## ACF   0.03 -0.01 -0.07  0.03  0.04 -0.04     0  0.01  0.00  0.00 -0.01
## PACF  0.00 -0.10 -0.01  0.01  0.00  0.03     0  0.02 -0.04 -0.02  0.00
```

d) Comparing Model Performance: Model Comparisons using AIC, BIC, and MSE

```
mse_m1 <- mean((m1$fit$residuals)^2)
mse_m2 <- mean((m2$fit$residuals)^2)
mse_m3 <- mean((m3$fit$residuals)^2)

cat("MSE of m1:", mse_m1, "\n")
```

```
## MSE of m1: 0.01896717
```

```r
cat("MSE of m2:", mse_m2, "\n")
```

```
## MSE of m2: 0.01706556
```

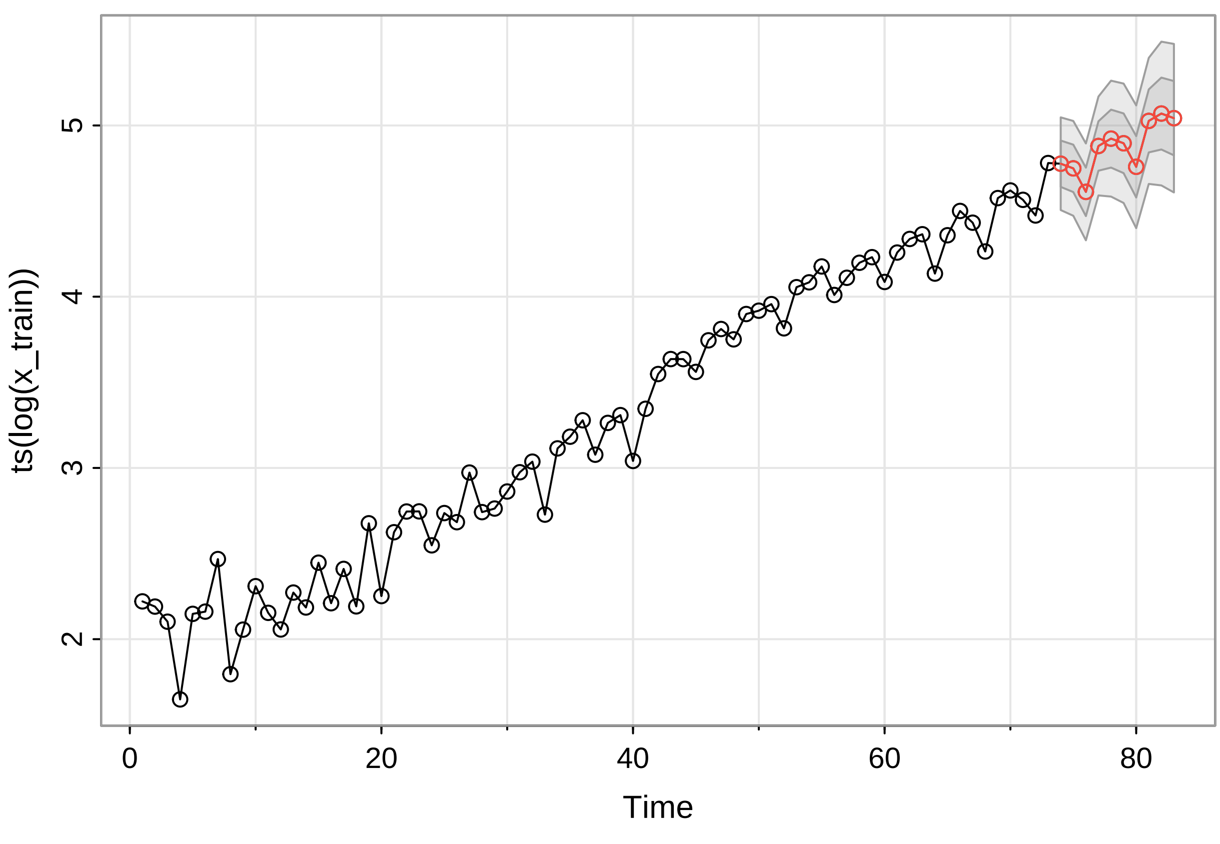
```r
cat("MSE of m3:", mse_m3, "\n")
```

```
## MSE of m3: 0.01682965
```

The results suggest model m2 has a good combination of low AIC, BIC, and MSE.

e) Forecasting 10 values.

```r
# Forecasting with the preferred SARIMA model
pred_c <- sarima.for(ts(log(x_train)), 0, 1, 1, 0, 1, 1, 4, n.ahead = 10)$pred
```




```r
# Converting predictions back from log scale
mse_forecast <- mean((exp(pred_c) - x_test)^2)
mse_forecast
```