Time Series Analysis

Name: Piyumika Herath Samarasekara

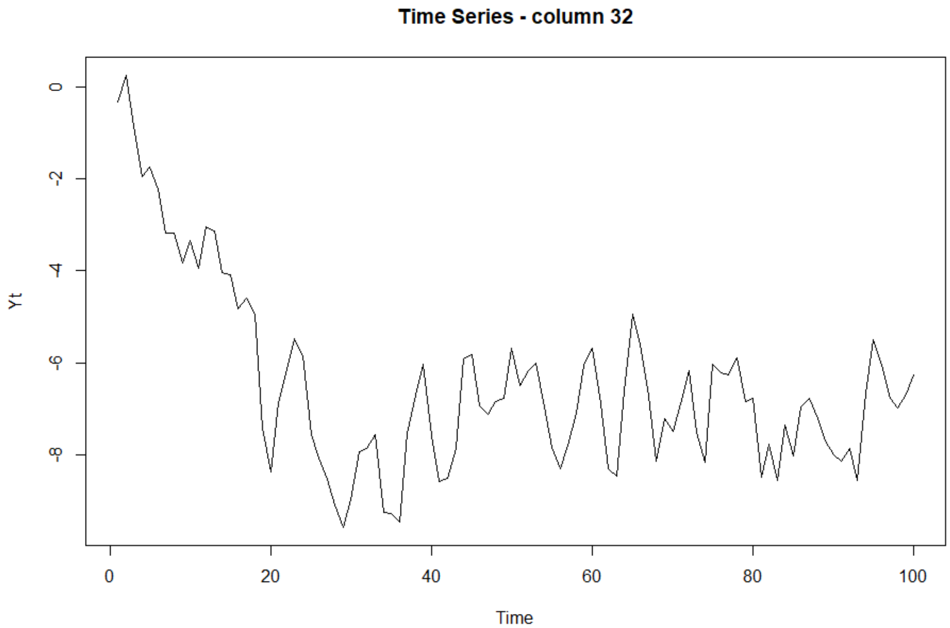
Column Assigned: 32

Assignment

Step 1: Preliminary Analysis of Orders

1. Plot the times series (Yt)

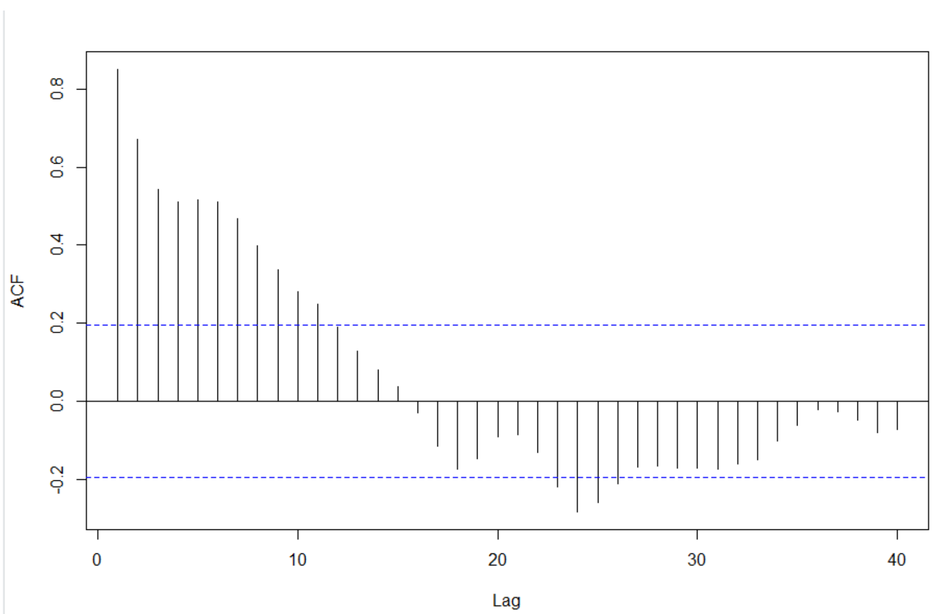
```
library(TSA)
library(lmtest)

#reading the data
data<-read.table("E:\\TUNI\\Semester2\\Time Series Analysis\\V32.txt", sep="\t", dec=".", header=TRUE)
attach(data)

#original series
Yt = data$V32

#plotting original series along with acf plot
plot.ts(Yt, main = 'Time Series - column 32')
acf(Yt,lag.max=40,type ="correlation",main = " " )
```



In this time series, there's a significant downward trend at the beginning. Then there's a slight increment. However, this effect levels off in the latter years. There are noticeable variations as well.

2. Analysis of d

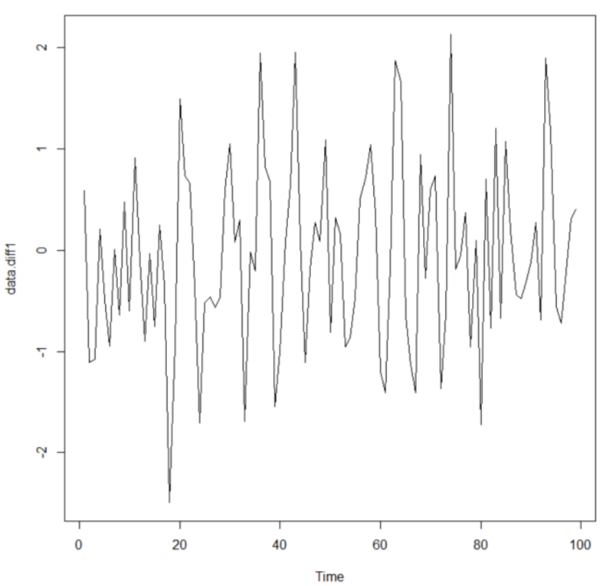
   (a) Autocorrelation function of Yt

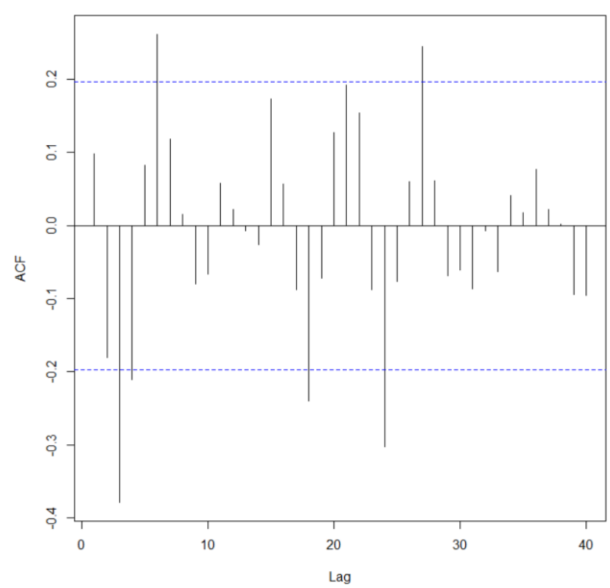


(1) The ACF plot clearly depicts that there's a correlation between the data points of the time series. So, the original series is not independent. And it's not stationary.

```
#obtaining the first difference series
data.diff1<- diff(Yt,1)
#plotting first difference series along with acf plot
plot.ts(data.diff1)
acf(data.diff1,lag.max=40,type = "correlation", main = " " )

#obtaining the second difference series
data.diff2<- diff(Yt,2)
#plotting second difference series along with acf plot
plot.ts(data.diff2)
acf(data.diff2,lag.max=40,type = "correlation", main = " " )
```

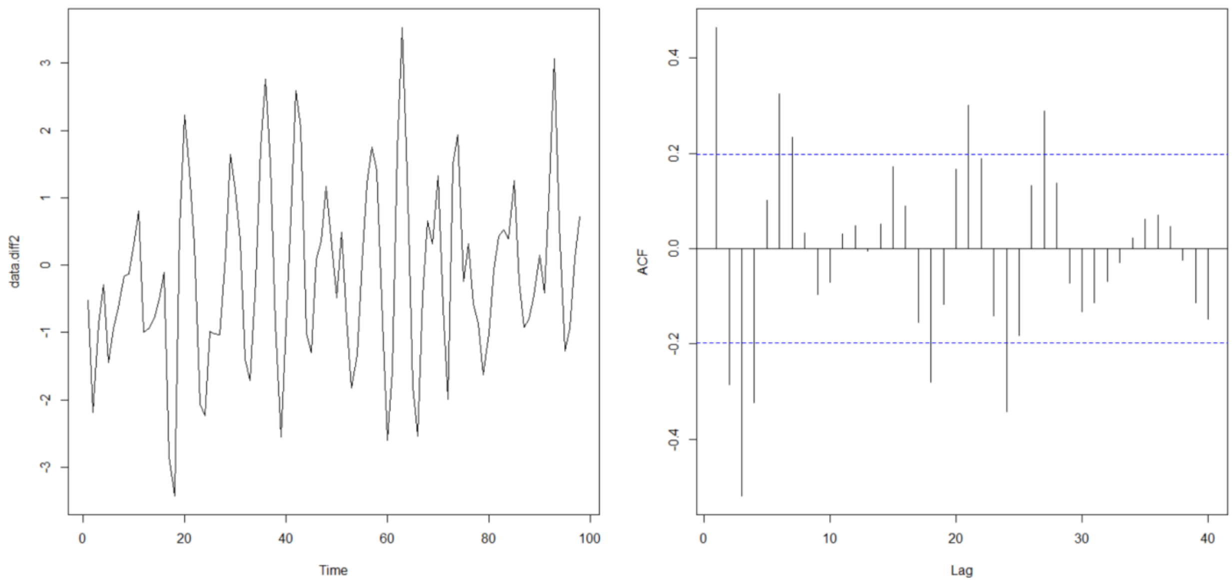
∇1Yt series and Autocorrelation function





(1) The trend has been disappeared after the first difference. And according to the ACF plot, it can be expressed that differenced series is much less correlated with a few significant autocorrelations.

∇2Yt series and Autocorrelation function



(1) The trend is not in the second differenced series. And some variations visible in the first differenced series have been removed now. According to the ACF plot, it can be expressed that differenced series is much less correlated with a few significant autocorrelations.

(2) It’s visible a decrement in the trend. According to the ACF plot for Yt series, there’s a rapid decrease in autocorrelation as the lag increases. In a stationary time series, autocorrelations should decay quickly to zero or near zero as the lag increases. Hence Yt is not a stationary – series.

(b)

```
#Conducting unit root test
library(tseries)
adf.test(Yt)
adf.test(data.diff1)
adf.test(data.diff2)
```

```
> adf.test(Yt)

        Augmented Dickey-Fuller Test

data:  Yt
Dickey-Fuller = -2.902, Lag order = 4, p-value = 0.203
alternative hypothesis: stationary
```

According to the p-value = 0.203 which is greater than 0.05, we do not reject H0. This means that there exists a root on the unit circle. Which depicts Yt needs to be differenced to achieve stationarity.

```
> adf.test(data.diff1)

        Augmented Dickey-Fuller Test

data:  data.diff1
Dickey-Fuller = -6.9759, Lag order = 4, p-value = 0.01
alternative hypothesis: stationary
```

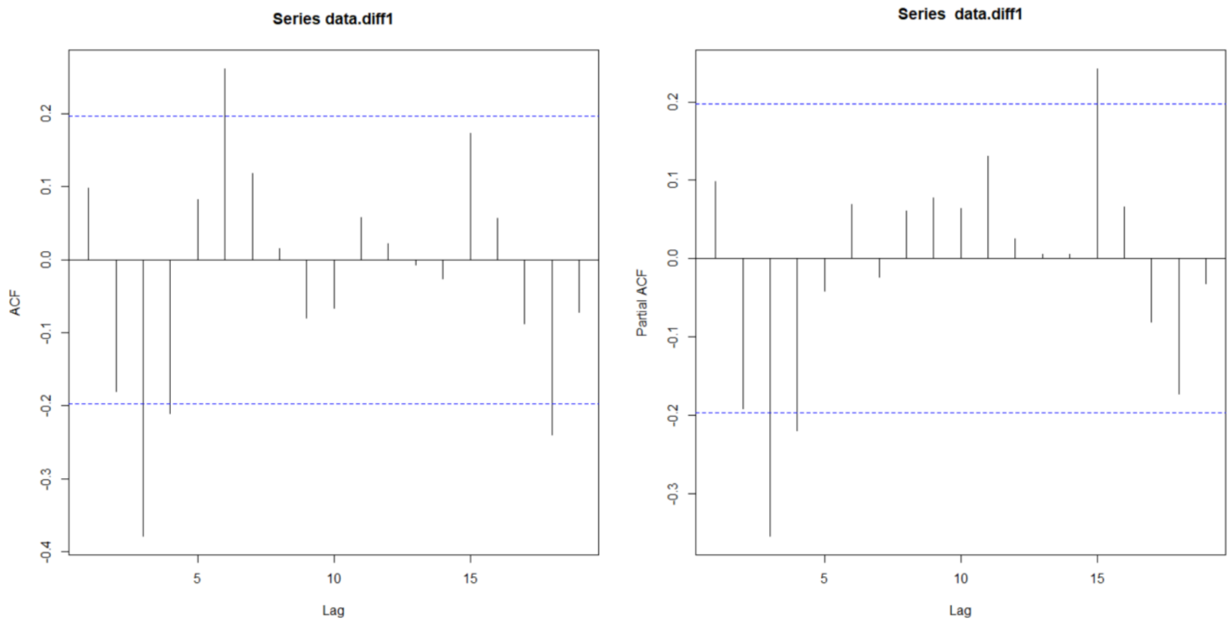
```
> adf.test(data.diff2)

        Augmented Dickey-Fuller Test

data:  data.diff2
Dickey-Fuller = -5.2285, Lag order = 4, p-value = 0.01
alternative hypothesis: stationary
```

In both the first differenced and second differenced series p-value is 0.01 which is less than 0.05. Hence, we can reject H0. This means there does not exist a root in the unit circle.

(c) After the first differencing the series shows promising results. The trend has been removed and there are less significant correlations between lags. And according to the ADF test, there does not exist a root on the unit circle. Hence, I prefer order 1 for d.

3. Analysis of (p, q)

```
##Step1 - 3
#Plotting ACF and PACF for selected series
acf(data.diff1)
pacf(data.diff1)
```



Both ACF and PACF plots dampen. In both cases ACF and PACF cutoff at lag 4. Hence, we can suggest 4 for p and q values. So, pmax = 4 and qmax = 4.

Step 2: Estimation and Selection of ARIMA Models

1.

```
##Step2 -1
#Fitting models

mod1 = arima(Yt, order = c(4,1,1))
res1=rstandard(mod1)
mod1
BIC(mod1)

mod2 = arima(Yt, order = c(4,1,2))
res2=rstandard(mod2)
mod2
BIC(mod2)

mod3 = arima(Yt, order = c(4,1,3))
res3=rstandard(mod3)
mod3
BIC(mod3)

mod4 = arima(Yt, order = c(4,1,4))
res4=rstandard(mod4)
mod4
BIC(mod4)
```

```r
mod5 = arima(Yt, order = c(3,1,1))
res5=rstandard(mod5)
mod5
BIC(mod5)

mod6 = arima(Yt, order = c(3,1,2))
res6=rstandard(mod6)
BIC(mod6)
mod6

mod7 = arima(Yt, order = c(3,1,3))
res7=rstandard(mod7)
mod7
BIC(mod7)

mod8 = arima(Yt, order = c(3,1,4))
res8=rstandard(mod8)
mod8
BIC(mod8)

mod9 = arima(Yt, order = c(2,1,1))
res9=rstandard(mod9)
mod9
BIC(mod9)

mod10 = arima(Yt, order = c(2,1,2))
res10=rstandard(mod10)
mod10
BIC(mod10)

mod11 = arima(Yt, order = c(2,1,3))
res11=rstandard(mod11)
mod11
BIC(mod11)
```

```r
mod12 = arima(Yt, order = c(2,1,4))
res12=rstandard(mod12)
mod12
BIC(mod12)

mod13 = arima(Yt, order = c(1,1,1))
res13=rstandard(mod13)
mod13
BIC(mod13)

mod14 = arima(Yt, order = c(1,1,2))
res14=rstandard(mod14)
mod14
BIC(mod14)

mod15 = arima(Yt, order = c(1,1,3))
res15=rstandard(mod15)
mod15
BIC(mod15)

mod16 = arima(Yt, order = c(1,1,4))
res16=rstandard(mod16)
mod16
BIC(mod16)
```

```
> mod1

Call:
arima(x = Yt, order = c(4, 1, 1))

Coefficients:
         ar1      ar2      ar3      ar4      ma1
      0.0673  -0.1797  -0.3278  -0.1788  -0.0870
s.e.  0.3711   0.0957   0.1059   0.1641   0.3717

sigma^2 estimated as 0.6677:  log likelihood = -120.81,  aic = 251.62
> BIC(mod1)
[1] 269.1872
```

```
> mod2

Call:
arima(x = Yt, order = c(4, 1, 2))

Coefficients:
         ar1      ar2      ar3      ar4      ma1     ma2
      0.3216  -0.3381  -0.2713  -0.1206  -0.3422  0.1721
s.e.  0.7583   0.3175   0.1631   0.2763   0.7603  0.3222

sigma^2 estimated as 0.666:  log likelihood = -120.69,  aic = 253.38
> BIC(mod2)
[1] 273.5431
```

```
> mod3

Call:
arima(x = Yt, order = c(4, 1, 3))

Coefficients:
         ar1      ar2     ar3      ar4      ma1     ma2      ma3
      0.1106  -0.7564  0.2081  -0.3233  -0.0866  0.5622  -0.4845
s.e.  0.2132   0.2215  0.2214   0.1152   0.1944  0.1897   0.1783

sigma^2 estimated as 0.6721:  log likelihood = -121.25,  aic = 256.5
> BIC(mod3)
[1] 279.2654
```

```
> mod4

Call:
arima(x = Yt, order = c(4, 1, 4))

Coefficients:
         ar1      ar2     ar3      ar4      ma1     ma2      ma3     ma4
      0.7500  -1.2367  0.7084  -0.6321  -0.7465  1.0666  -0.9375  0.5115
s.e.  0.2032   0.1633  0.1773   0.1209   0.2281  0.1441   0.1855  0.1670

sigma^2 estimated as 0.6267:  log likelihood = -119.17,  aic = 254.34
> BIC(mod4)
[1] 279.6976
```

```
> mod5

Call:
arima(x = Yt, order = c(3, 1, 1))

Coefficients:
         ar1      ar2      ar3      ma1
      0.3853  -0.1860  -0.3045  -0.3901
s.e.  0.1752   0.1035   0.1076   0.1692

sigma^2 estimated as 0.6741:  log likelihood = -121.26,  aic = 250.52
> BIC(mod5)
[1] 265.4976
```

```
> BIC(mod6)
[1] 269.0192
> mod6

Call:
arima(x = Yt, order = c(3, 1, 2))

Coefficients:
         ar1      ar2      ar3      ma1     ma2
      0.6544  -0.4343  -0.2207  -0.6760  0.2685
s.e.  0.3435   0.2672   0.1669   0.3502  0.2387

sigma^2 estimated as 0.6665:  log likelihood = -120.72,  aic = 251.45
>
```

```
> mod7

Call:
arima(x = Yt, order = c(3, 1, 3))

Coefficients:
         ar1      ar2      ar3      ma1     ma2      ma3
      0.6678  -0.4548  -0.2044  -0.6888  0.2911  -0.0163
s.e.  0.6133   0.7618   0.5626   0.6076  0.8013   0.4899

sigma^2 estimated as 0.6665:  log likelihood = -120.72,  aic = 253.45
> BIC(mod7)
[1] 273.6126
```

```
> mod8

Call:
arima(x = Yt, order = c(3, 1, 4))

Coefficients:
         ar1      ar2     ar3      ma1     ma2     ma3      ma4
      0.3659  -0.1528  -0.437  -0.3843  -0.025  0.1724  -0.0373
s.e.     NaN      NaN     NaN      NaN     NaN     NaN      NaN

sigma^2 estimated as 0.6672:  log likelihood = -120.77,  aic = 255.54
> BIC(mod8)
 [1] 278.2998


> mod9

Call:
arima(x = Yt, order = c(2, 1, 1))

Coefficients:
         ar1      ar2      ma1
      0.6916  -0.3511  -0.6391
s.e.  0.1303   0.0960   0.1071

sigma^2 estimated as 0.7205:  log likelihood = -124.44,  aic = 254.88
> BIC(mod9)
[1] 267.257
> mod10

Call:
arima(x = Yt, order = c(2, 1, 2))

Coefficients:
         ar1      ar2      ma1     ma2
      1.0684  -0.7413  -1.0700  0.4842
s.e.  0.1349   0.1158   0.1794  0.1543

sigma^2 estimated as 0.6735:  log likelihood = -121.22,  aic = 250.44
> BIC(mod10)
[1] 265.4202
```

```
> mod11

Call:
arima(x = Yt, order = c(2, 1, 3))

Coefficients:
         ar1      ar2      ma1     ma2      ma3
      0.8716  -0.7220  -0.8836  0.5651  -0.1826
s.e.  0.2082   0.1186   0.2189  0.1868   0.1595

sigma^2 estimated as 0.6675:  log likelihood = -120.8,  aic = 251.59
> BIC(mod11)
[1] 269.161
```

```
> mod12

Call:
arima(x = Yt, order = c(2, 1, 4))

Coefficients:
         ar1      ar2      ma1     ma2      ma3     ma4
      1.2308  -0.4677  -1.4005  0.3923  -0.0911  0.3053
s.e.  0.1444   0.1475   0.1600  0.2474   0.2091  0.1246

sigma^2 estimated as 0.5957:  log likelihood = -117.62,  aic = 247.23
> BIC(mod12)
[1] 267.4005
```

```
> mod13

Call:
arima(x = Yt, order = c(1, 1, 1))

Coefficients:
          ar1     ma1
      -0.1441  0.2707
s.e.   0.3832  0.3633

sigma^2 estimated as 0.8262:  log likelihood = -131.03,  aic = 266.06
> BIC(mod13)
[1] 275.8488
```

```
> mod14

Call:
arima(x = Yt, order = c(1, 1, 2))

Coefficients:
         ar1      ma1      ma2
      0.4323  -0.4372  -0.2365
s.e.  0.1784   0.1653   0.0852

sigma^2 estimated as 0.7604:  log likelihood = -127.01,  aic = 260.03
> BIC(mod14)
[1] 272.4061
```

```
> mod15

Call:
arima(x = Yt, order = c(1, 1, 3))

Coefficients:
         ar1      ma1      ma2      ma3
      0.1625  -0.1440  -0.1509  -0.2296
s.e.  0.2439   0.2304   0.1134   0.0946

sigma^2 estimated as 0.7298:  log likelihood = -125.01,  aic = 258.03
> BIC(mod15)
[1] 273.0051
```

```
> mod16

Call:
arima(x = Yt, order = c(1, 1, 4))

Coefficients:
          ar1     ma1      ma2      ma3      ma4
      -0.1859  0.2239  -0.0958  -0.2748  -0.1852
s.e.   0.3741  0.3574   0.1085   0.0915   0.1333

sigma^2 estimated as 0.7212:  log likelihood = -124.45,  aic = 258.91
> BIC(mod16)
[1] 276.4785
```

2. According to the above results below mentioned models have been selected. The model with the lowest AIC value has a slightly large BIC value. The model with the lowest BIC value has a slightly higher AIC value. So, by considering both factors these 3 models have been obtained which have lower AIC and BIC values compared to other models.

Model	AIC	BIC
Model 5 – ARIMA (3,1,1)	250.52	265.49
Model10 – ARIMA (2,1,2)	250.44	265.42
Model12 – ARIMA (2,1,4)	247.23	267.4

```
#Extimates of the best models
coeftest(mod5)
coeftest(mod10)
coeftest(mod12)
```

The estimated parameters for each of the selected models are as follows.

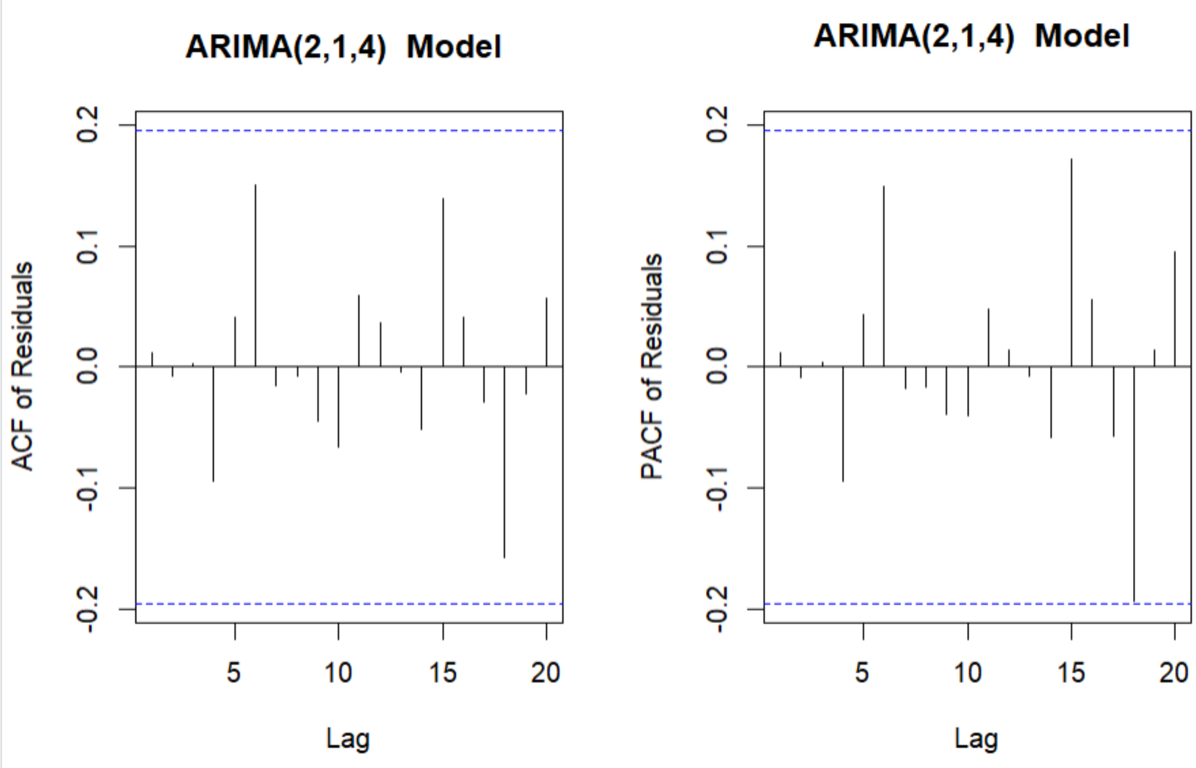
```
> #Extimates of the best models
> coeftest(mod5)

z test of coefficients:

    Estimate Std. Error z value Pr(>|z|)
ar1  0.38533    0.17520  2.1993 0.027853 *
ar2 -0.18596    0.10353 -1.7962 0.072464 .
ar3 -0.30450    0.10762 -2.8295 0.004663 **
ma1 -0.39013    0.16918 -2.3060 0.021111 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> coeftest(mod10)

z test of coefficients:

    Estimate Std. Error z value  Pr(>|z|)
ar1  1.06835    0.13495  7.9169 2.435e-15 ***
ar2 -0.74131    0.11583 -6.4000 1.554e-10 ***
ma1 -1.07003    0.17943 -5.9636 2.468e-09 ***
ma2  0.48422    0.15434  3.1373  0.001705 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> coeftest(mod12)

z test of coefficients:

     Estimate Std. Error z value  Pr(>|z|)
ar1  1.230833   0.144378  8.5251 < 2.2e-16 ***
ar2 -0.467702   0.147518 -3.1705  0.001522 **
ma1 -1.400525   0.160043 -8.7510 < 2.2e-16 ***
ma2  0.392250   0.247428  1.5853  0.112895
ma3 -0.091106   0.209141 -0.4356  0.663112
ma4  0.305318   0.124607  2.4502  0.014276 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Step 3: Diagnostic tests

1. LjungBox test for the first 10 lags

```
##Step3 - 1

#Ljung-Box test for the selected 3 models with ACF and PACF plots
#ARIMA(2,1,4)
Box.test(res12,lag =10,type="Ljung-Box")
acf(residuals(mod12),main= "ARIMA(2,1,4)  Model ",ylab="ACF of Residuals")
pacf(residuals(mod12),main= "ARIMA(2,1,4)  Model ",ylab="PACF of Residuals")

#ARIMA(3,1,1)
Box.test(res5,lag =10,type="Ljung-Box")
acf(residuals(mod5),main= "ARIMA(3,1,1)  Model ",ylab="ACF of Residuals")
pacf(residuals(mod5),main= "ARIMA(3,1,1)  Model ",ylab="PACF of Residuals")

#ARIMA(2,1,2)
Box.test(res10,lag =10,type="Ljung-Box")
acf(residuals(mod10),main= "ARIMA(2,1,2)  Model ",ylab="ACF of Residuals")
pacf(residuals(mod10),main= "ARIMA(2,1,2)  Model ",ylab="PACF of Residuals")
```

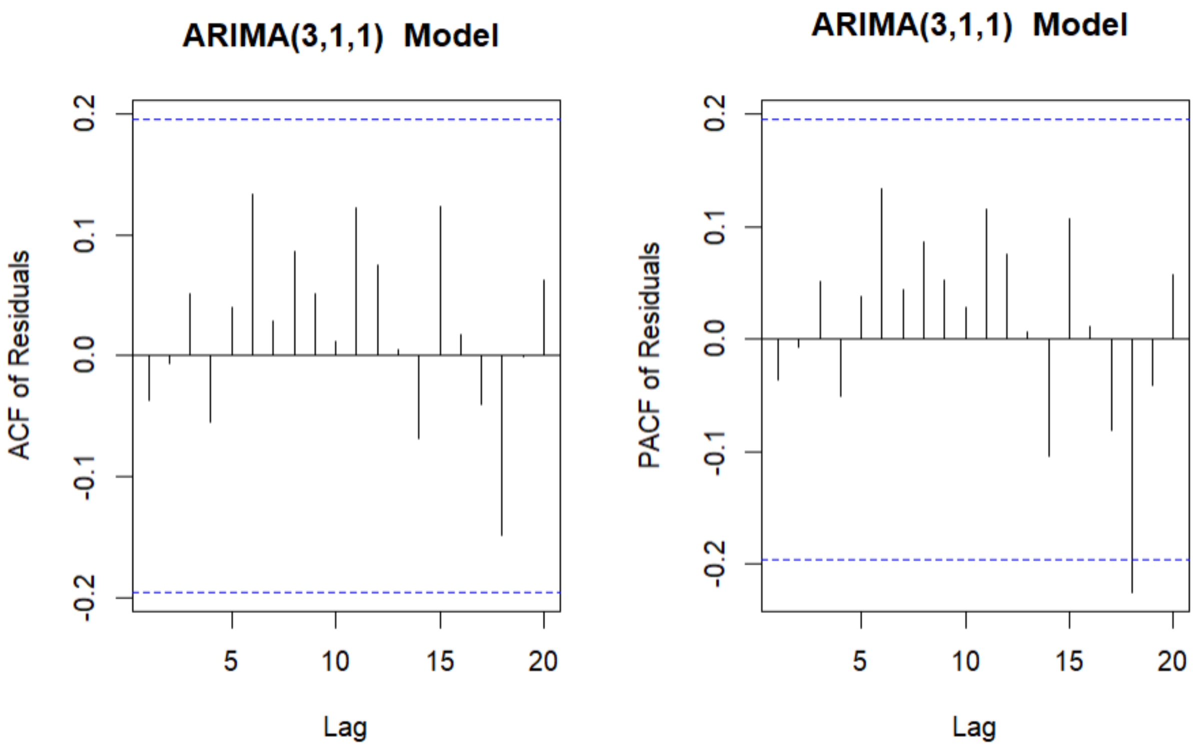
```
> Box.test(res12,lag =10,type="Ljung-Box")

        Box-Ljung test

data:  res12
X-squared = 4.3455, df = 10, p-value = 0.9304
```



The residuals from the ARIMA (2,1,4) model are not correlated with other lags. The Ljung-Box test indicates that, jointly, the residual autocorrelations are small. Here the p-value is greater than 0.05.

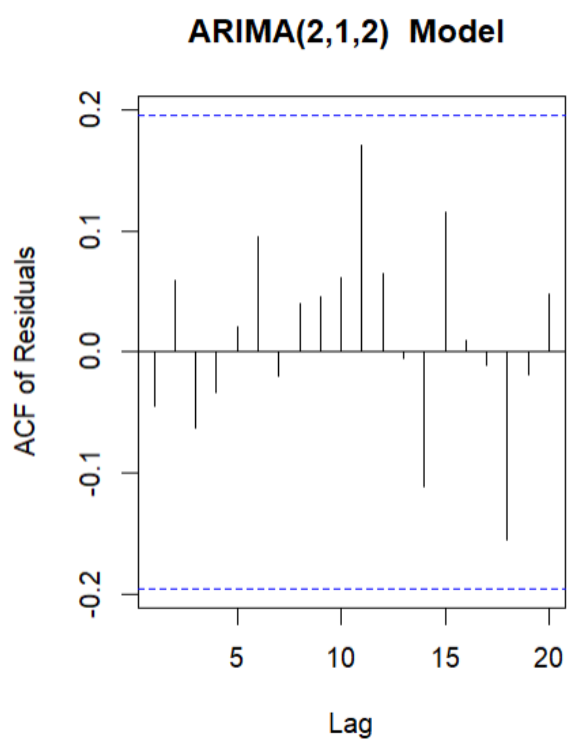
Hence, we do not reject H0. Which means observations are independent and similarly distributed with finite variance.

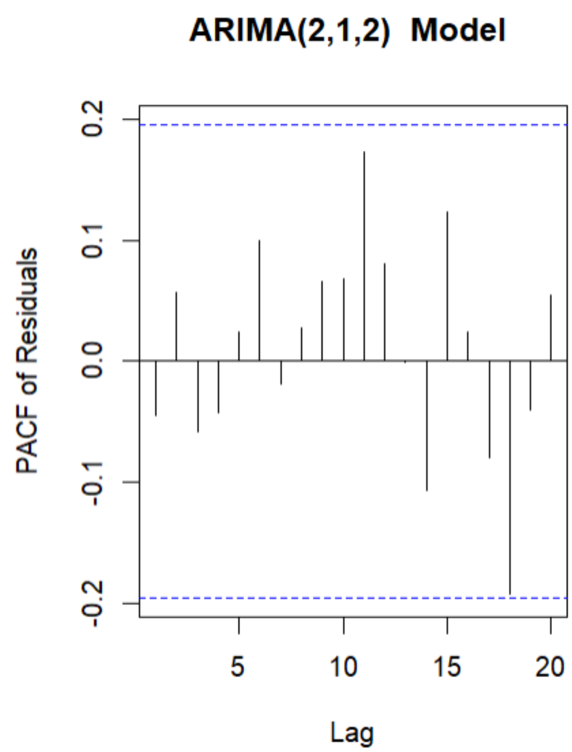
```
> Box.test(res5,lag =10,type="Ljung-Box")

	Box-Ljung test

data:  res5
X-squared = 4.0721, df = 10, p-value = 0.944
```



The residuals from the ARIMA (3,1,1) model are not correlated with other lags. The Ljung-Box test indicates that, jointly, the residual autocorrelations are small. Here the p-value is greater than 0.05. Hence, we do not reject H0. Which means observations are independent and similarly distributed with finite variance.

```
> Box.test(res10,lag =10,type="Ljung-Box")

	Box-Ljung test

data:  res10
X-squared = 3.0165, df = 10, p-value = 0.981
```

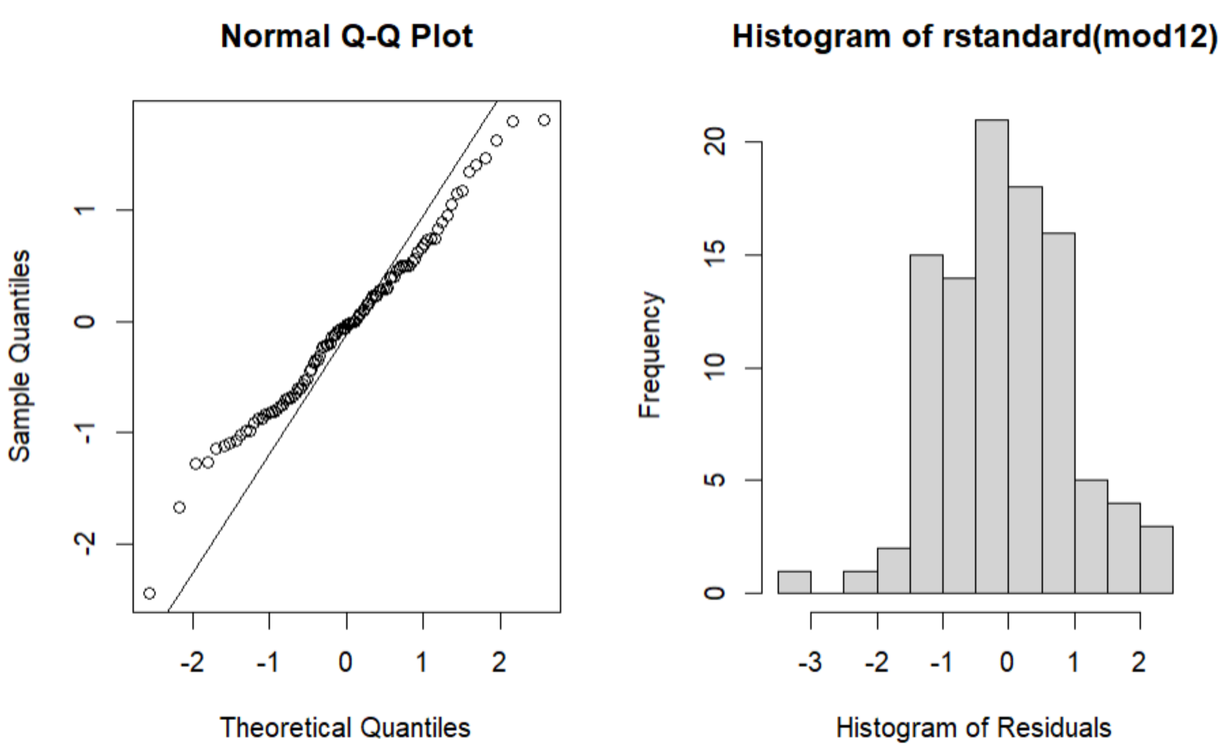




The residuals from the ARIMA (2,1,2) model are not correlated with other lags. The Ljung-Box test indicates that, jointly, the residual autocorrelations are small. Here the value is greater than 0.05. Hence, we do not reject H0. Which means observations are independent and similarly distributed with finite variance.

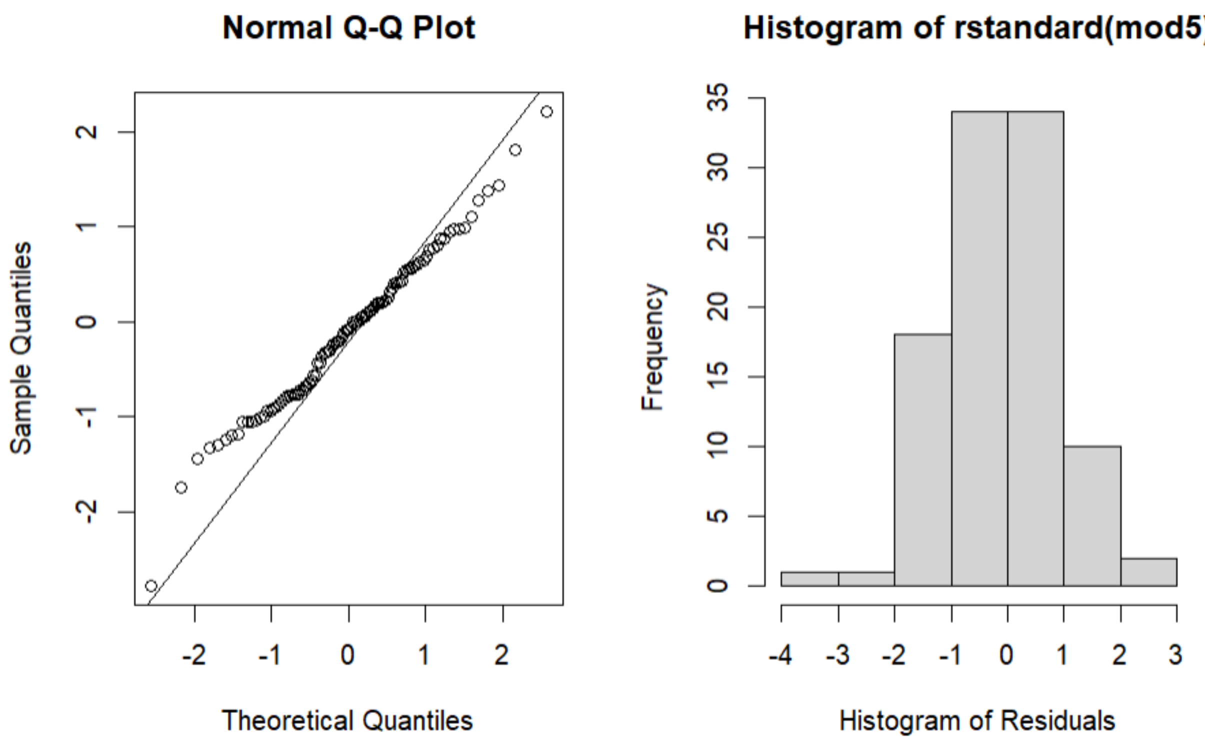
2. Check for residuals normality.

```
##Step3 - 2
##histograms, QQ plots, and Shapiro-Wilk's test

#ARIMA(2,1,4)
qqnorm(residuals(mod12)); qqline(rstandard(mod12))
hist(rstandard(mod12),xlab ="Histogram of Residuals")
shapiro.test(residuals(mod12))

#ARIMA(3,1,1)
qqnorm(residuals(mod5)); qqline(rstandard(mod5))
hist(rstandard(mod5),xlab ="Histogram of Residuals")
shapiro.test(residuals(mod5))

#ARIMA(2,1,2)
qqnorm(residuals(mod10)); qqline(rstandard(mod10))
hist(rstandard(mod10),xlab ="Histogram of Residuals")
shapiro.test(residuals(mod10))
```



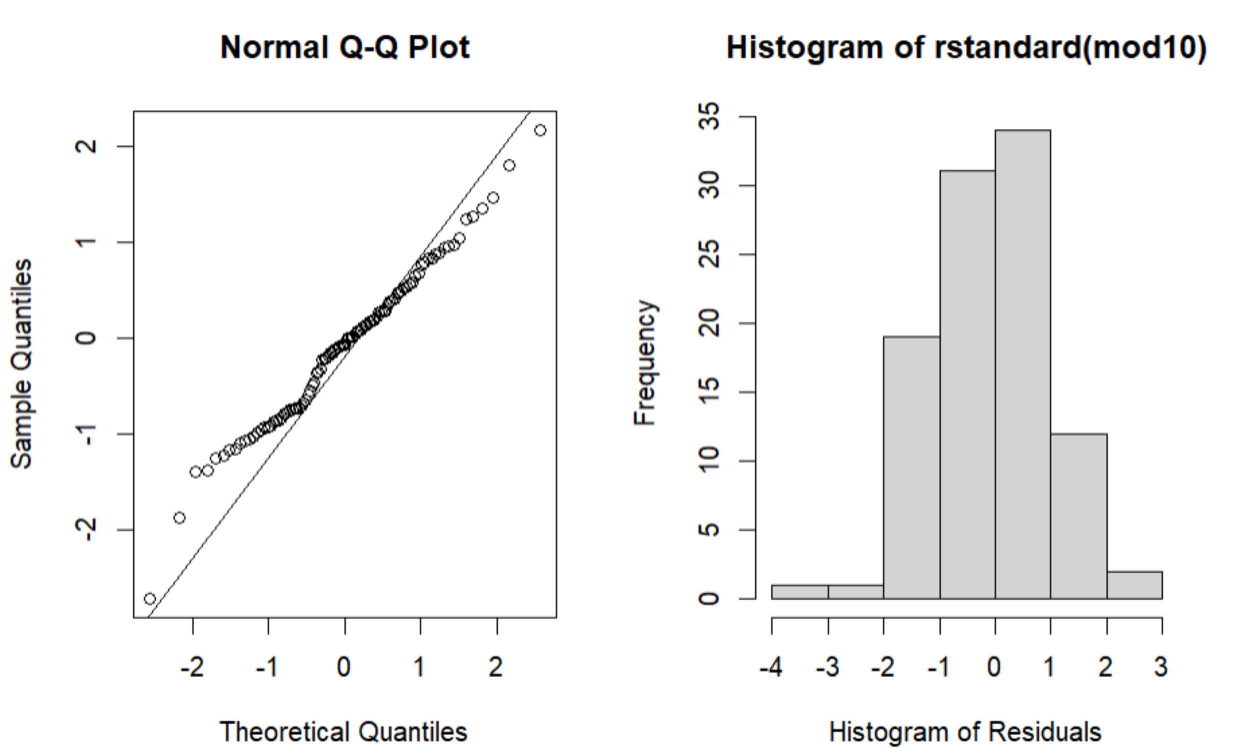

```
> shapiro.test(residuals(mod12))

        Shapiro-Wilk normality test

data:  residuals(mod12)
W = 0.98895, p-value = 0.581
```

In the ARIMA (2,1,4) model, the normal QQ plot and histogram depict that there's no problem with the normality of error terms. Since the p-value of the Shapiro test is greater than 0.05 we do not reject H0. This also implies that there's no problem with the normality of the error terms.




```
> shapiro.test(residuals(mod5))

        Shapiro-Wilk normality test

data:  residuals(mod5)
W = 0.98893, p-value = 0.5787
```

In the ARIMA (3,1,1) model, the normal QQ plot and histogram depict that there's no problem with the normality of error terms. Since the p-value of the Shapiro test is greater than 0.05 we do not reject H0. This also implies that there's no problem with the normality of the error terms.




```
> shapiro.test(residuals(mod10))

        Shapiro-Wilk normality test

data:  residuals(mod10)
W = 0.98889, p-value = 0.5757
```

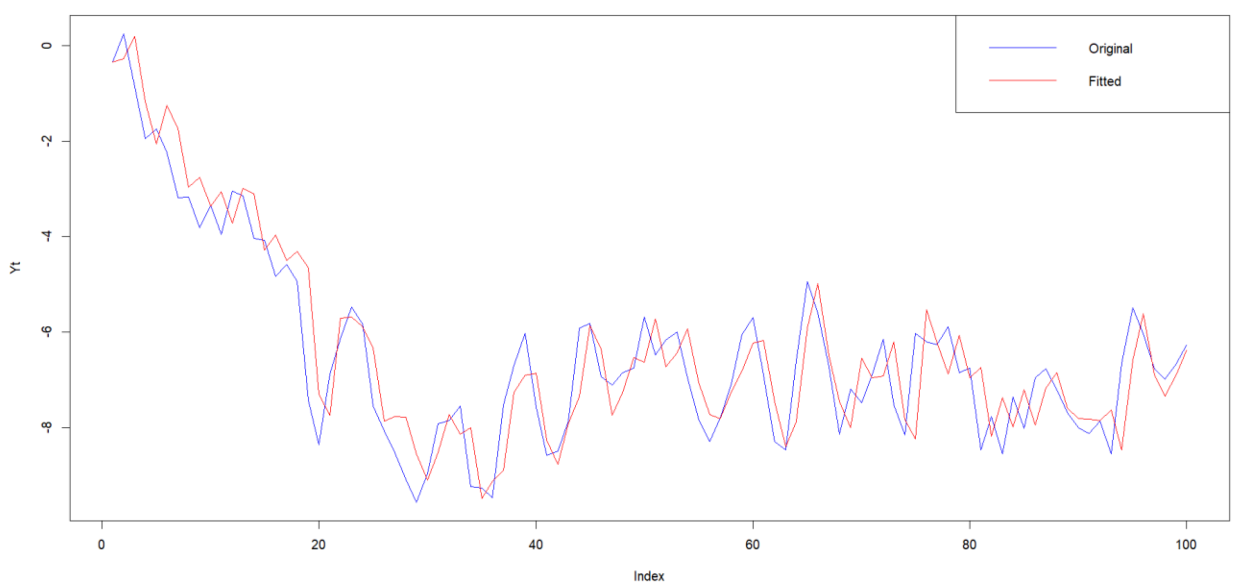
In the ARIMA (2,1,2) model, the normal QQ plot and histogram depict that there's no problem with the normality of error terms. Since the p-value of the Shapiro test is greater than 0.05 we do not reject H0. This also implies that there's no problem with the normality of the error terms.

3. According to the previous results of the selected 3 models, ARIMA (2,1,4), ARIMA (3,1,1), and ARIMA (2,1,2), there's not much significant difference between the AIC and BIC values. Also, the residuals of all 3 models show the normality. Hence when selecting the model, the principle of parsimony has been considered.

Since the ARIMA (2,1,4) model has more parameters than the other 2 it has not been considered to go further. Then from the rest of the two models, by considering the number of MA components, the ARIMA (3,1,1) model has been selected as the final model.

4. Original time series and best-fitted model

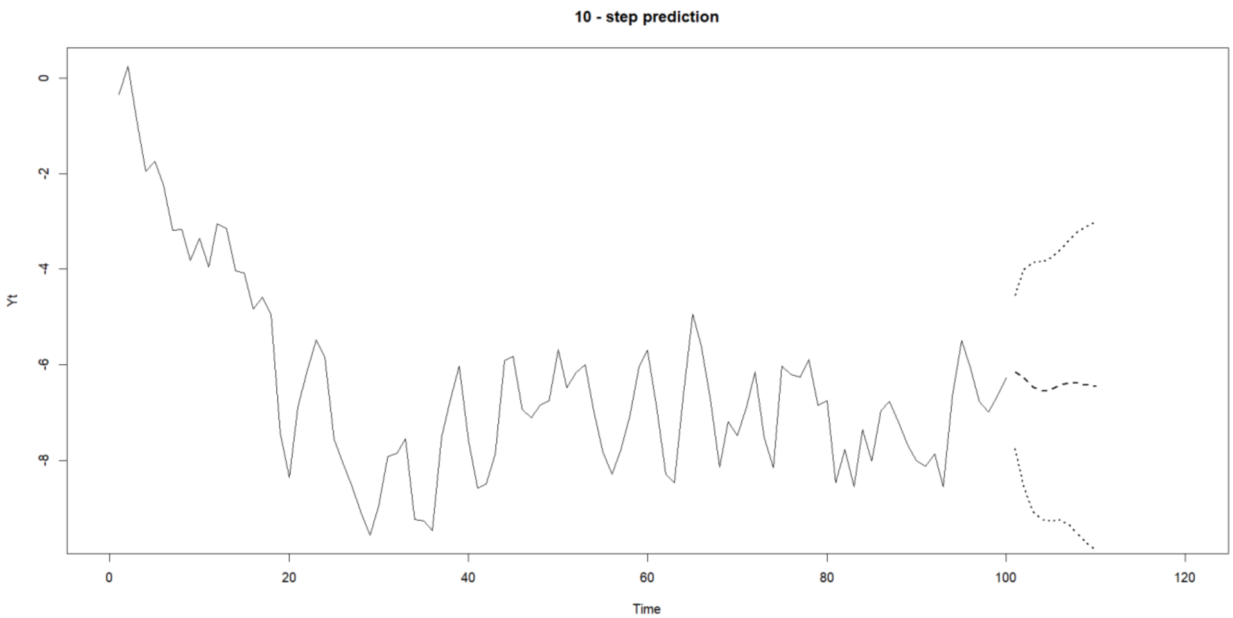
```
##Step3 - 4
# Extract fitted values and residuals
fitted_values <- fitted(mod5)

# Plot the original dataset, fitted values, and residuals
plot(Yt, type = "l", col = "blue", ylim = range(Yt, fitted_values))
lines(fitted_values, col = "red")

# Add legend
legend("topright", legend = c("Original", "Fitted"), col = c("blue", "red"), lty = 1)
```



Step 4: Forecast

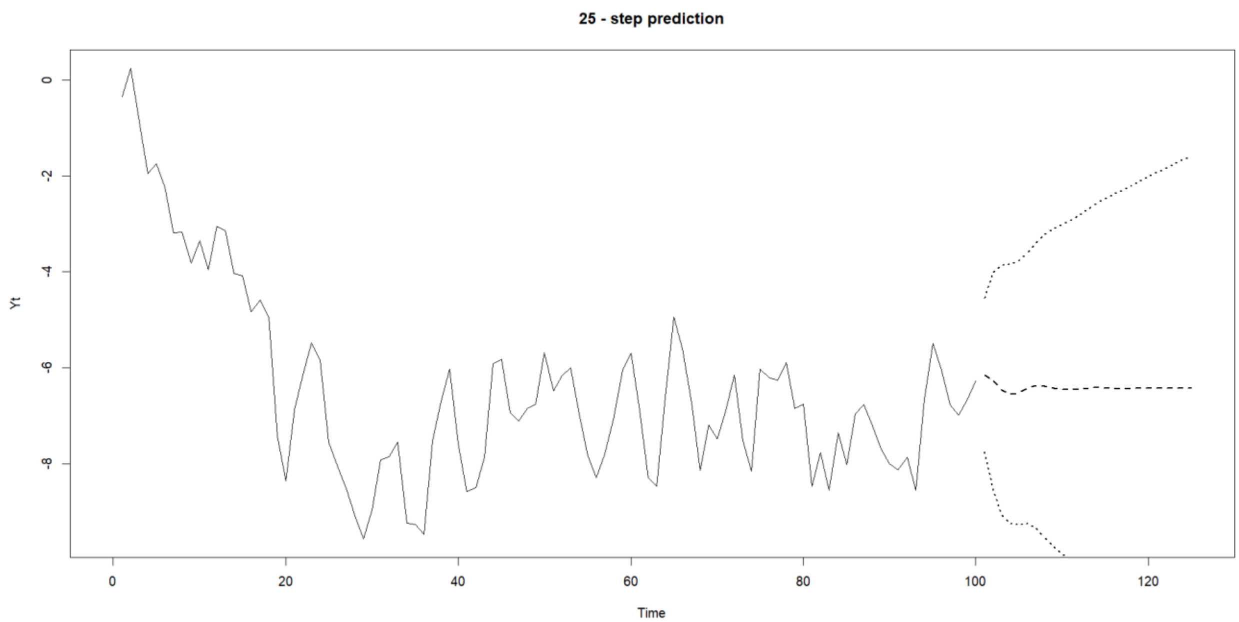
1. When h = 10

```r
#When h = 10
pred <- predict(mod5,10)

plot.ts(Yt,xlim =c (0 ,120), main = '10 - step prediction')

#Plotting the forecasted data with 95% CI
lines(pred$pred,lty=2,lwd=2)
lines(pred$pred-1.96*pred$se,lty=3,lwd=2)
lines(pred$pred+1.96*pred$se,lty =3,lwd =2)
```



2. When h=25

```r
#When h = 25
pred <- predict(mod5,25)

plot.ts(Yt,xlim =c (0 ,125), main = '25 - step prediction')

#Plotting the forecasted data with 95% CI
lines(pred$pred,lty=2,lwd=2)
lines(pred$pred-1.96*pred$se,lty=3,lwd=2)
lines(pred$pred+1.96*pred$se,lty =3,lwd =2)
```


25 - step prediction
Yt
0
-2
-4
-6
-8
Time
0
20
40
60
80
100
120