Assignment 2 Time Series

Wesley Nderi(s3635870)

21/04/2018

Introduction

Contextual background

Coregonus hoyi (bloater) is a form of freshwater whitefish commonly described as a silvery coloured herring-like fish. The bloater is one of several white fish species that have become rare almost to the point of extinction.This can be majorly attributed to overfishing due to the increase of human population in the region. In addition, the intrusion of several invasive fish species such as the *Alewife* and *Petromyzon marinus*(sea lampreys) that had a similar diet to the Bloater. For species that could not adapt their diets disappeared, became smaller in size or declined in numbers. The bloater is one such species and as a result is listed as vulnerable to global extinction by the IUCN Red List (http://www.iucnredlist.org/details/5366/0).

Methodology

The **first task** in this assignment is to analyze yearly changes in the egg depositions of Lake Huron Bloaters recorded from 1981 to 1996. The **second task** is to find the best fitting trend model and the **third task** is to give a forecast of yearly changes for the next five years.

This dataset was provided by Mr.Haydar Demirhan but is also available in the **FSAdata** package as **BloaterLH**.

Task 1: Analysing egg depositions of Lake Huron Bloaters

```
# Packages required
library(tseries)
library(TSA)
library(fUnitRoots)
library(lmtest)
library(FitAR)
```

```
#Read the data into R
eggs<- read.csv("/Users/wes/Desktop/Rlesson/eggs.csv")
```

This dataset does not appear to be a time series object.

```
class(eggs)
```

```
## [1] "data.frame"
```

```
#Convert into a time series object
eggs.ts= ts(eggs$eggs,start = 1981)
```

```
class(eggs.ts)
```

```
## [1] "ts"
```

We can now plot a visualisation of the time series.

```
plot(eggs.ts,type='o',ylab='Egg depositions(in millions)', main='Yearly egg depositio
ns of Lake Huron Bloaters')
```

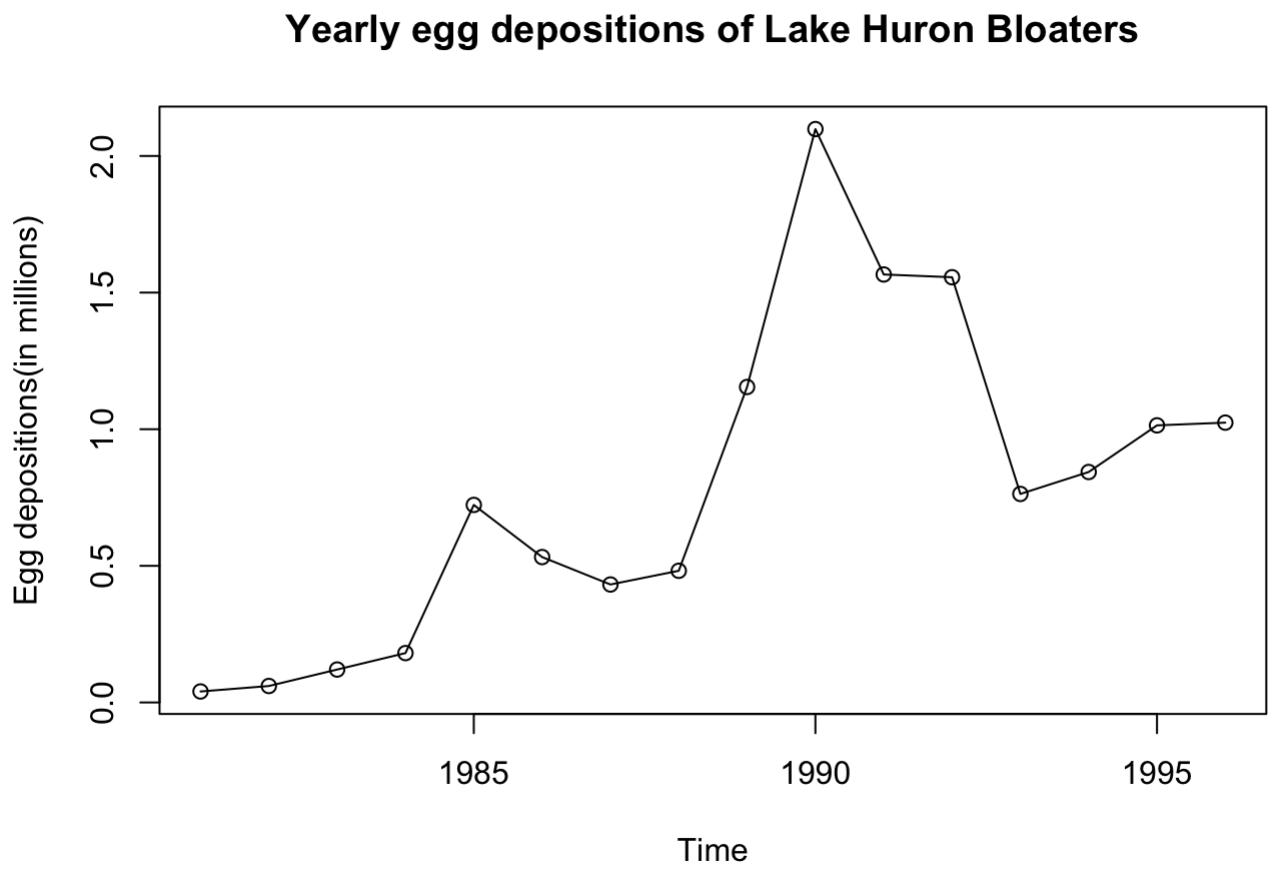

Figure 1: Visualisation of the time series

What can we observe from the above plot in terms of the following?

a)Trend: There does seem to be a general upward trend although it is not obvious.

b)Behaviour: It appears that the above series has an auto-regressive component as there are numerous succeeding points.

c)Seasonality: There is no obvious seasonality.

d)Changing variance: It is possible that there is a change in variance depicted by the movement from high values to low values.

Normality of the series

We can check for the normality of the series using a QQPlot and the Shapiro-Wilk test for normality.

Checking for normality using a QQ plot

Non-normality can be assessed using a quantile-quantile (QQ) plot. With normally distributed values, a QQ plot looks approximately like a straight line.

```
qqnorm(eggs.ts)
qqline(eggs.ts,col=2)
```

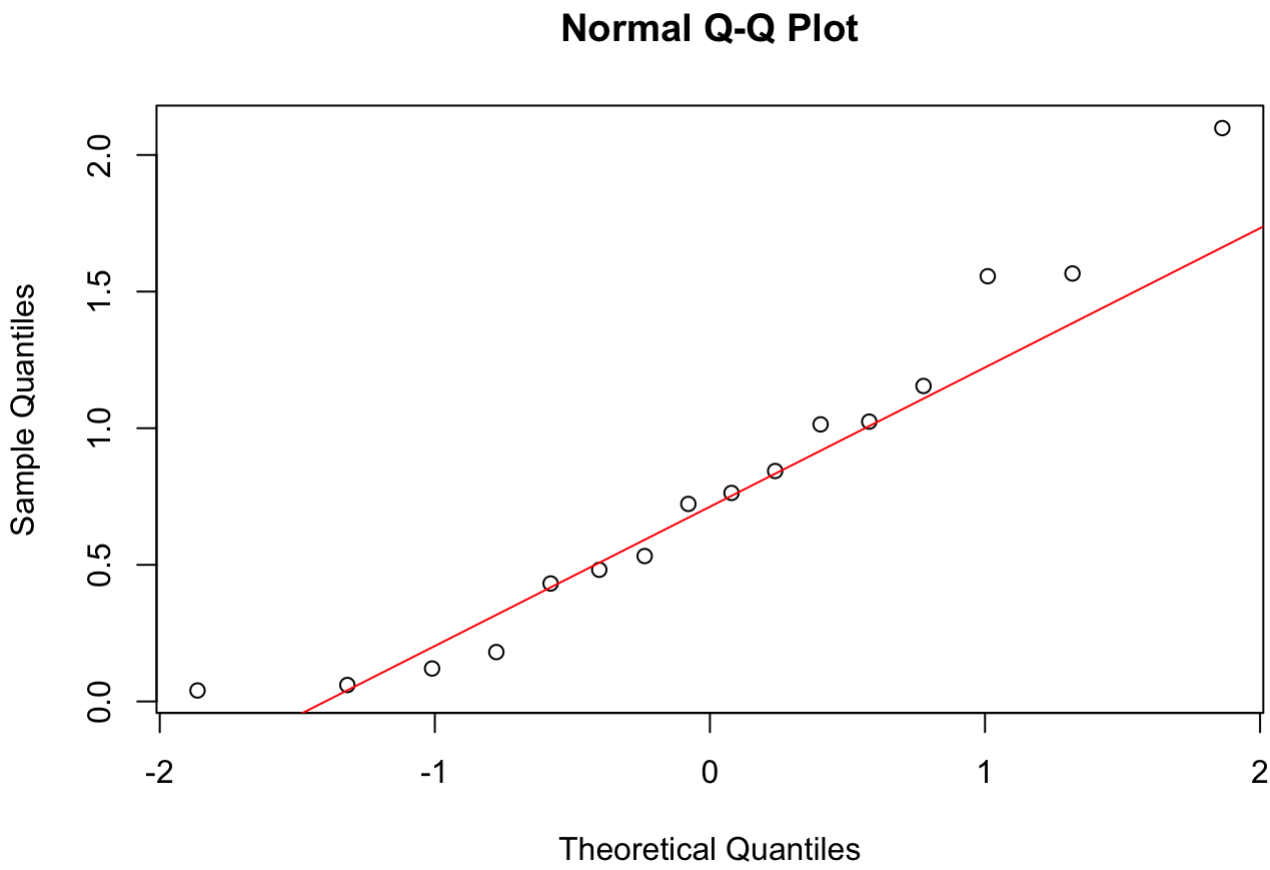

Figure 2: QQplot of the series

From the QQ plot above, we observe that a majority of the distribution of the series falls along the line of normality. However, we notice several outliers near the ends of the distribution that departure from the line of normality. These may or may not affect the normality of the data but this shall be confirmed using the Shapiro test as shown below.

Checking for normality using the Shapiro-Wilk test

This test calculates the correlation between the residuals and the corresponding normal quantiles. The lower the correlation, the lower the evidence of normality. Similarly, the higher the correlation, the more evidence of normality.

```
shapiro.test(eggs.ts)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  eggs.ts
## W = 0.94201, p-value = 0.3744
```

According to the Shapiro test, the series is normal. We do not have enough evidence to reject the assumption of normality.

ACF and PACF of the series

We can display the ACF and PACF plots of this series.

```
par(mfrow=c(1,2))
acf(eggs.ts,main='Sample ACF')
pacf(eggs.ts, main='Sample PACF')
```

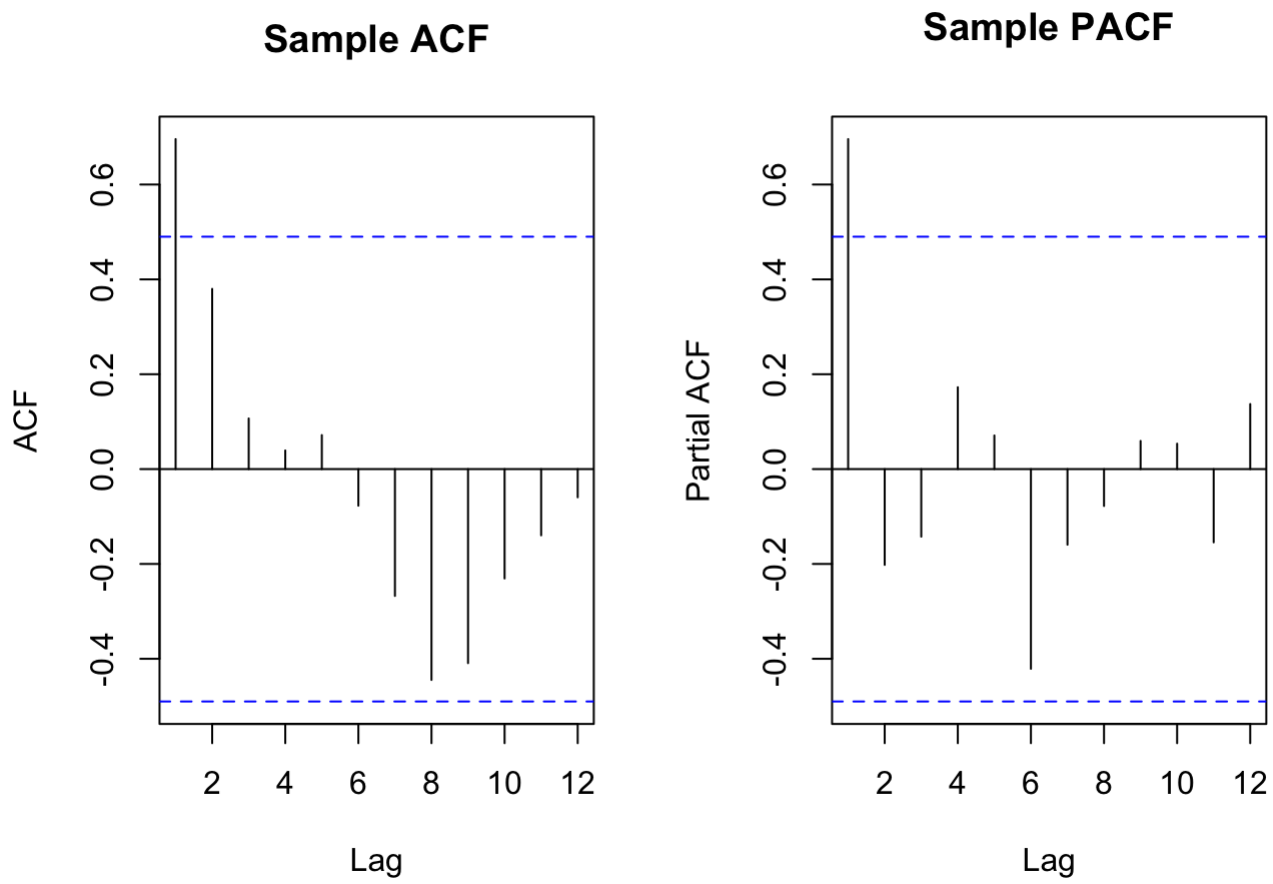

Figure 3: ACF and PACF of the series

From the ACF and PACF observe, a significant lag in the ACF and a significant lag in the PACF. It is also important to highlight the slight decaying trend in the ACF. Although the lags are not significant for us to consider that this is clearly evidence of a trend, it does signify that this may be highly probable.

We shall apply the ADF unit-root test to test the existence of non-stationarity within the series.

```
# Getting the order of lags
ar(diff(eggs.ts))
```

```
##
## Call:
## ar(x = diff(eggs.ts))
##
##
## Order selected 0  sigma^2 estimated as  0.1841
```

```
# ADF unit root test
adfTest(eggs.ts, lags = 0)
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 0
##   STATISTIC:
##     Dickey-Fuller: -0.4911
##   P VALUE:
##     0.452
##
## Description:
##  Sun May  6 13:10:58 2018 by user:
```

With a p-value of 0.452, we cannot reject the null hypothesis stating that the series is non-stationary. We conclude that the series is non-stationary at 5% kevel of significance.

```
kpss.test(eggs.ts,"Trend")
```

```
##
##  KPSS Test for Trend Stationarity
##
## data:  eggs.ts
## KPSS Trend = 0.19322, Truncation lag parameter = 0, p-value =
## 0.01854
```

The KPSS test has a null hypothesis that the series is trend stationary. The **BloaterLH** series has a p-value of 0.01854 further confirms that the series is non stationary.

We shall proceed to address the non stationary nature of the data by first transforming it and differencing as required.

Transformations

Box Cox transformation

```
eggs.ts.transform = BoxCox.ar(eggs.ts, method = c("yule-walker"))
```

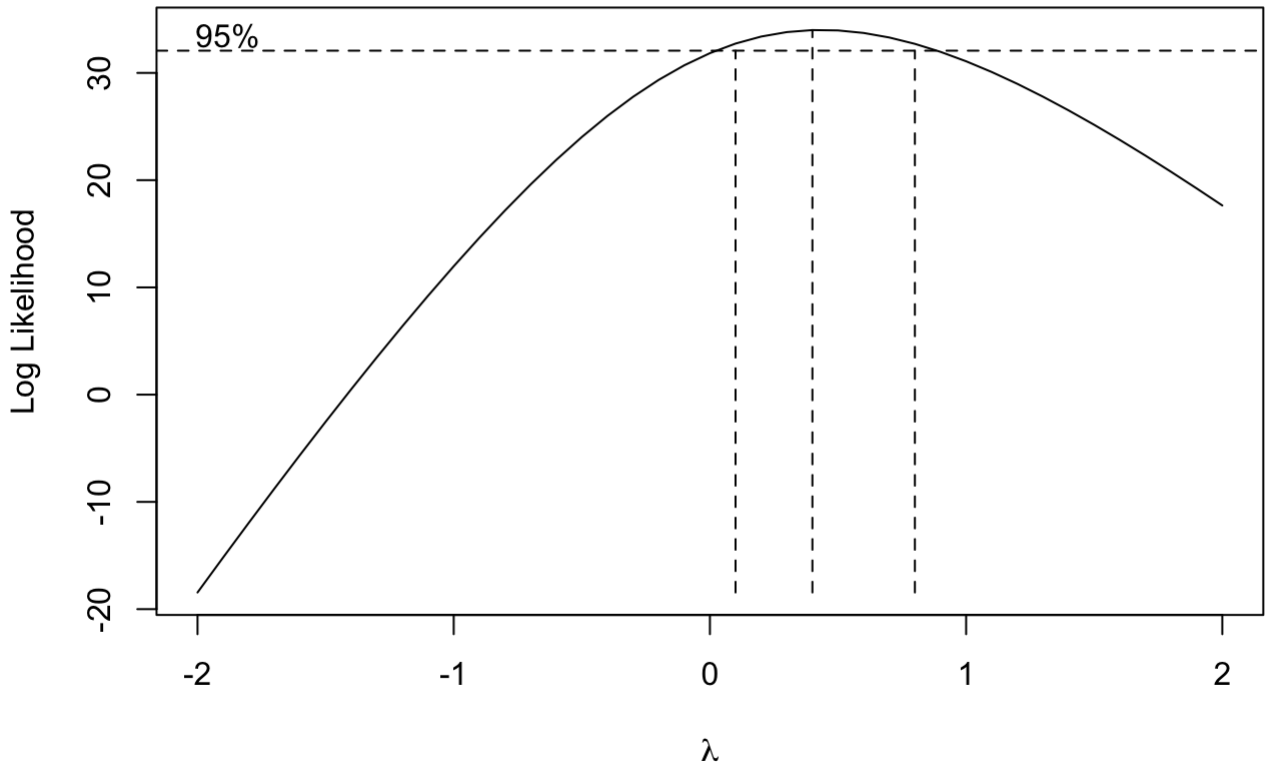

Figure 4: Box-Cox Transformation of the series

```
eggs.ts.transform$ci
```

```
## [1] 0.1 0.8
```

```
lambda = 0.45 # Value is far from the log transformation
BC.eggs = sqrt(eggs.ts)
```

We observe that the confidence intervals are 0.1 and 0.8 which gives a lambda midpoint value of 0.45. At this lambda value we observe the maximum log likelihood. As this is close to 0.5, we shall pick a square root transformation.

Let us now observe the effect of this transformation on the time series plot.

```
plot(BC.eggs,type='o',ylab='Egg depositions(in millions)', main='Yearly egg depositio
ns of Lake Huron Bloaters')
```

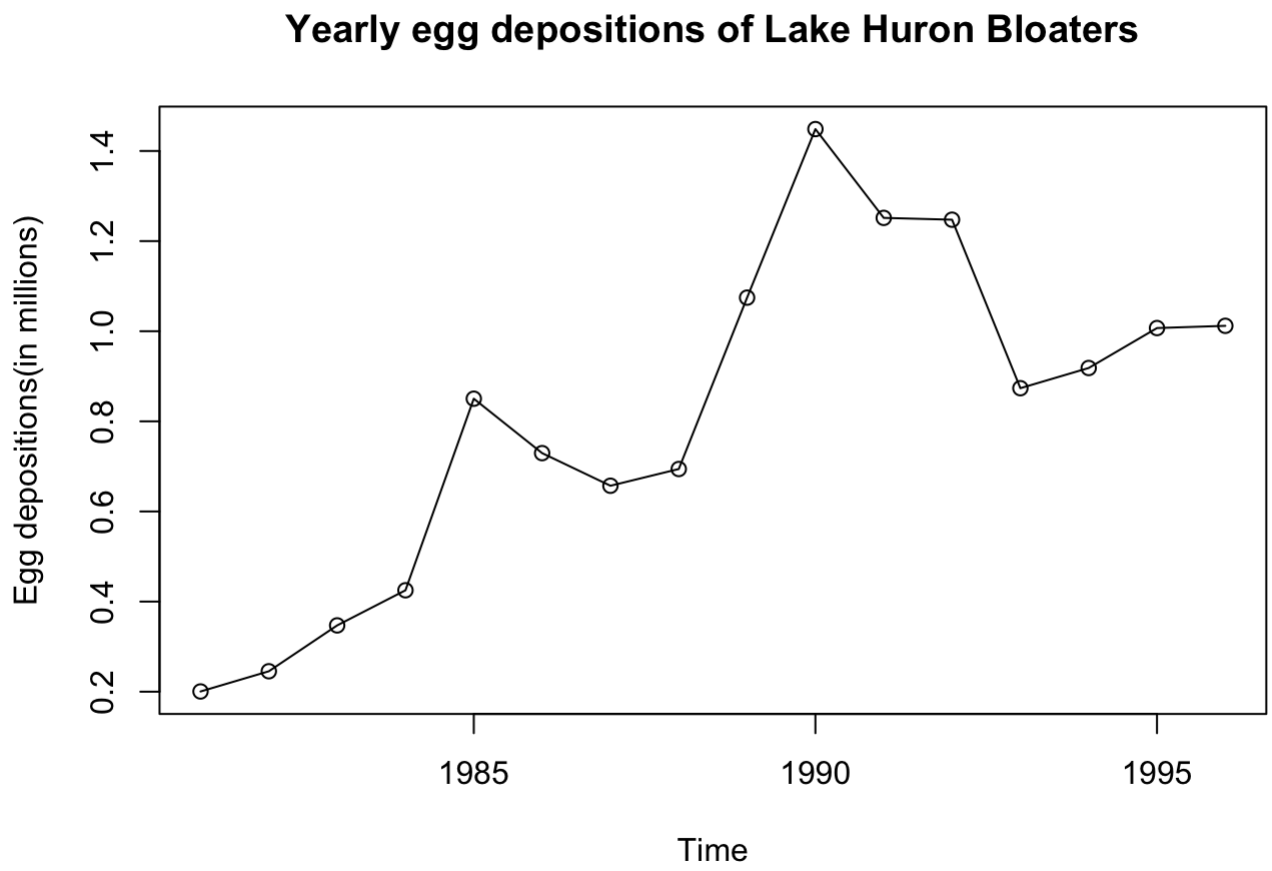

Figure 5: Visualisation of series after Box Cox transformation

It does not seem to have had a significant change in the plot although as we can see below from the Shapiro-Wilk test and QQplot below, it did improve the normality of the series.

```
shapiro.test(BC.eggs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  BC.eggs
## W = 0.96562, p-value = 0.7636
```

```
qqnorm(BC.eggs, main='QQ plot of transformed series')
qqline(BC.eggs,col=2)
```

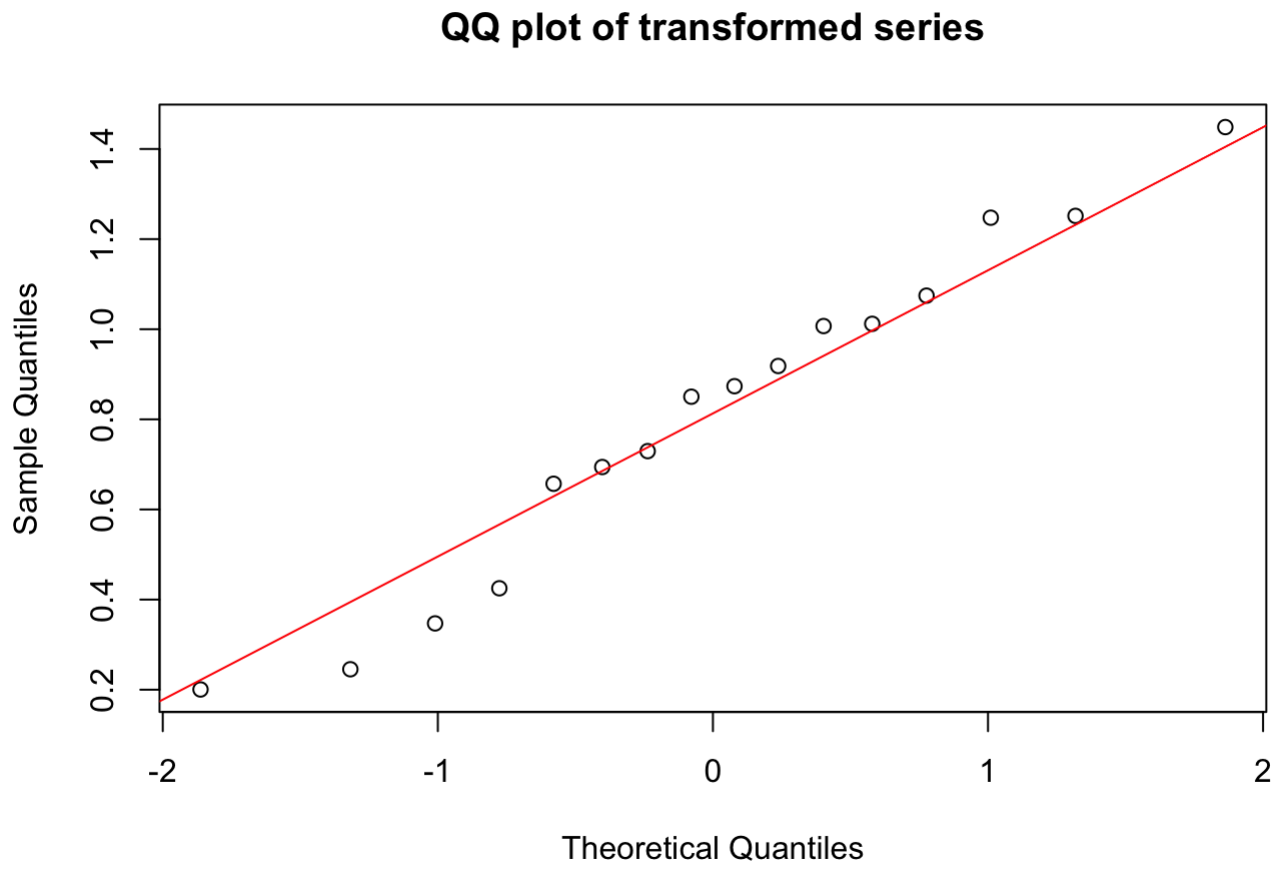

Figure 6: QQplot of series after Box Cox transformation

Unit Root test

We shall perform an Augmented Dickey-Fuller Test to observe whether the transformation was successful in eliminating the trend in the series. As observed in the plot of the series above, it does not appear to have done so.

```
ar(diff(BC.eggs))
```

```
##
## Call:
## ar(x = diff(BC.eggs))
##
##
## Order selected 0  sigma^2 estimated as  0.0464
```

Order for the test is obtained as 0.

```
adfTest(BC.eggs, lags = 0)
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 0
##   STATISTIC:
##     Dickey-Fuller: 0.192
##   P VALUE:
##     0.6689
##
## Description:
##  Sun May  6 13:10:59 2018 by user:
```

With a p-value of 0.6689 we can statistically conclude that the series is non stationary at 5% level of significance and we do not have enough evidence to reject the null hypothesis. The transformation did not address the trend present in the series.

We shall proceed to differencing because it is clear that there is still a trend in the series.

ACF and PACF of the transformed series

The ACF and PACF of the transformed series are very similar to the original series as shown below.

```
par(mfrow=c(1,2))
acf(BC.eggs,main='Sample ACF')
pacf(BC.eggs, main='Sample PACF')
```

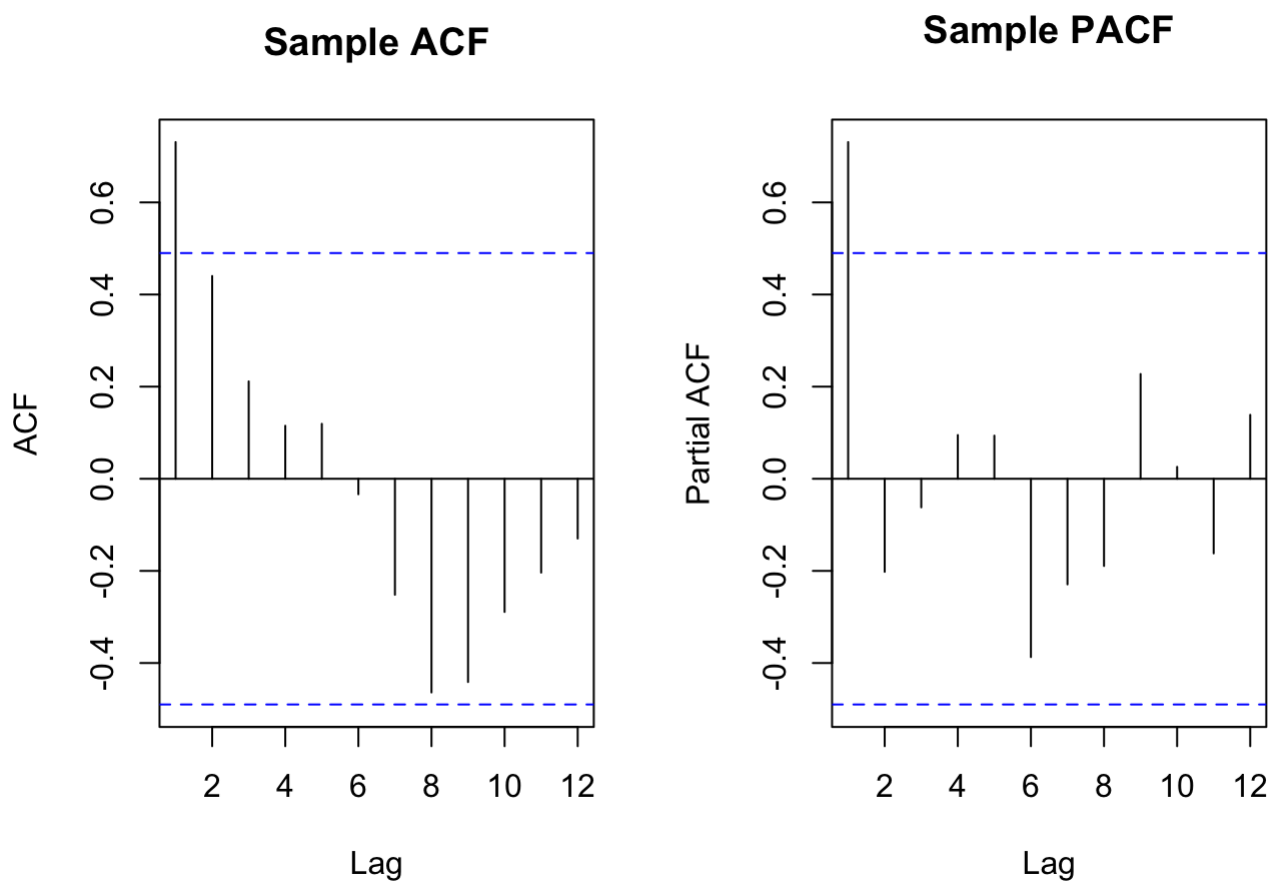

Figure 7: ACF and PACF of series after Box Cox transformation

We see one significant autocorrelation in ACF and 1 significant autocorrelation in PACF. Yet again we observe that the decaying pattern in the ACF has now become much clearer albeit the lags are not significant.

Differencing

```
#First difference
diff.BC.eggs = diff(BC.eggs, differences = 1)
```

```
plot(diff.BC.eggs, type = "o", main = "Plot of First difference of Lake Huron Bloater
s series")
```

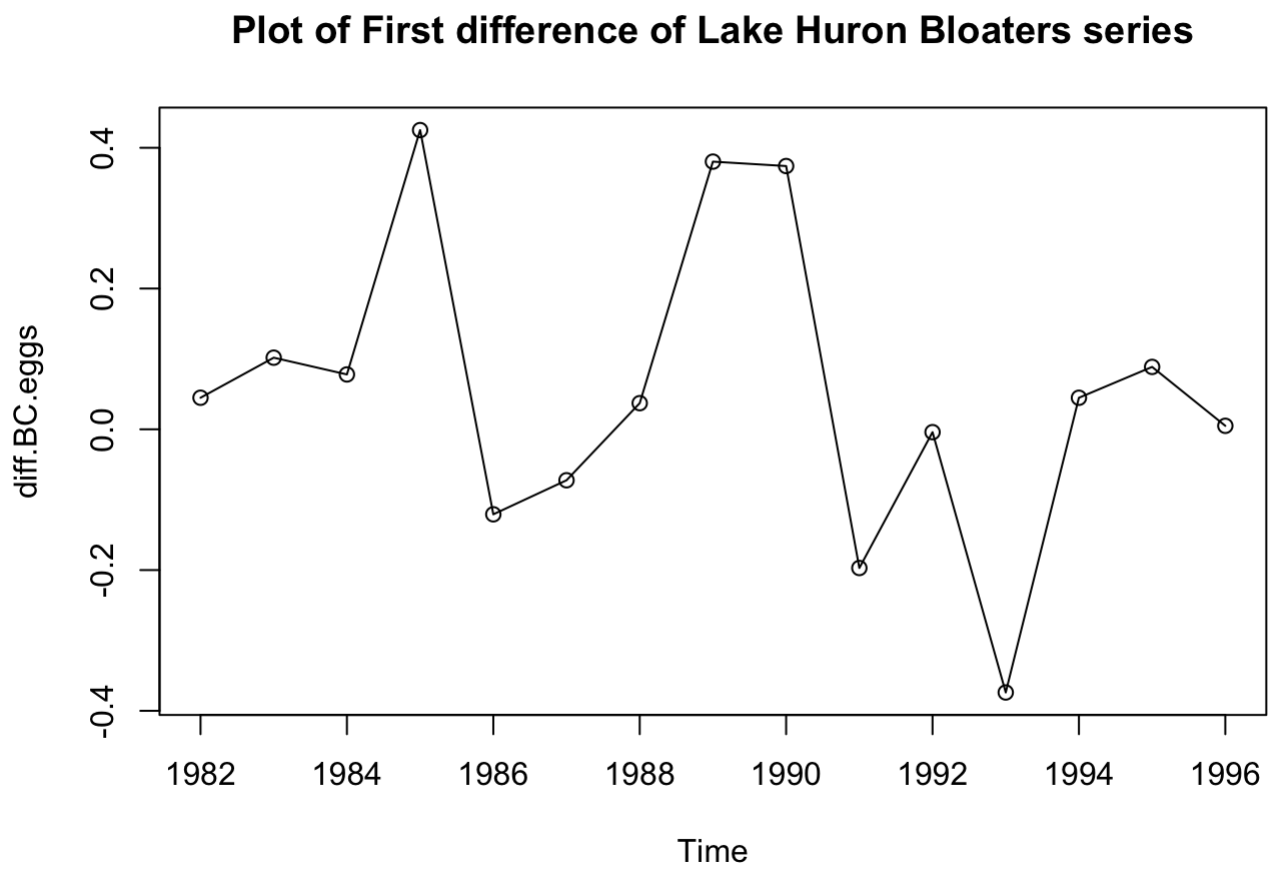

Figure 8:Series after first difference

The first difference seems to have addressed the trend in the series although we see that there are extreme highs and lows.

We can proceed to conduct an ADF test to confirm whether the series is stationary.

```
ar(diff(diff.BC.eggs))
```

```
##
## Call:
## ar(x = diff(diff.BC.eggs))
##
## Coefficients:
##       1        2        3        4
## -0.7629  -0.5389  -0.7151  -0.5455
##
## Order selected 4  sigma^2 estimated as  0.06464
```

Order for the test is obtained as 4.

```
adfTest(diff.BC.eggs, lags = 4)
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 4
##   STATISTIC:
##     Dickey-Fuller: -0.8002
##   P VALUE:
##     0.3539
##
## Description:
##  Sun May  6 13:10:59 2018 by user:
```

With a p-value of 0.3539 we do not have enough evidence to reject the null hypothesis that the series is non-stationary. We can conclude non stationarity of the series at 5% level of significance. We shall apply the second difference.

```
#Second difference
diff.BC.eggs2 = diff(BC.eggs, differences = 2)
```

```
plot(diff.BC.eggs2, type = "o", main = "Plot of Second difference of Lake Huron Bloat
ers series")
```

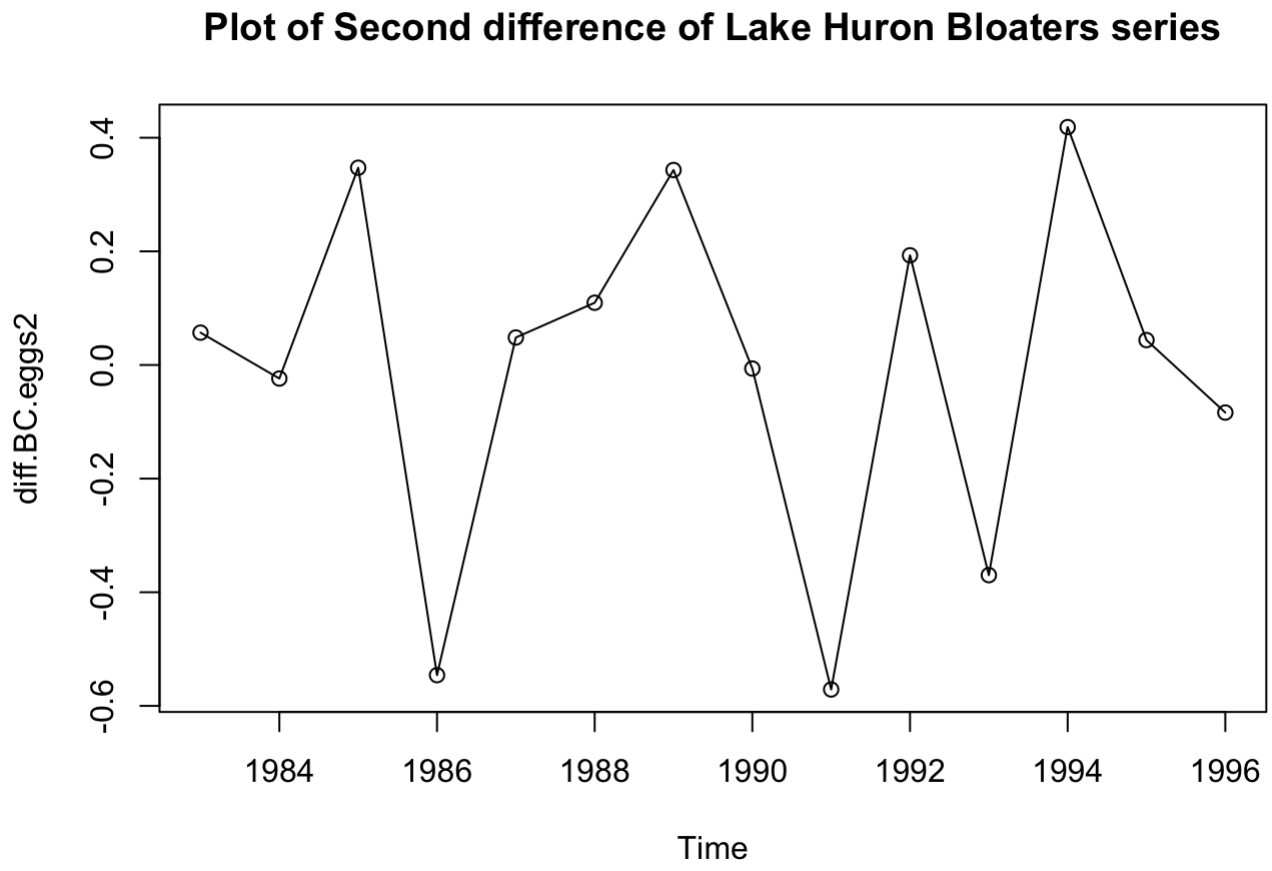

Figure 9:Series after second difference

The second difference looks better although we still see several extreme highs and lows. We shall apply the ADF unit root test to see if the series is stationary.

```
ar(diff(diff.BC.eggs2))
```

```
##
## Call:
## ar(x = diff(diff.BC.eggs2))
##
## Coefficients:
##       1        2        3        4
## -1.0729  -0.7936  -0.8347  -0.6104
##
## Order selected 4  sigma^2 estimated as  0.133
```

Order for the test is obtained as 4.

```
adfTest(diff.BC.eggs2,lags = 4)
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 4
##   STATISTIC:
##     Dickey-Fuller: -1.5305
##   P VALUE:
##     0.1221
##
## Description:
##  Sun May  6 13:10:59 2018 by user:
```

With a p-value of 0.1221 we can conclude non-stationarity of the series at 5% level of significance. We shall therefore apply the third difference.

```
#Third difference
diff.BC.eggs3 = diff(BC.eggs, differences = 3)
```

```
plot(diff.BC.eggs3, type = "o", main = "Plot of third difference of Lake Huron Bloate
rs series")
```

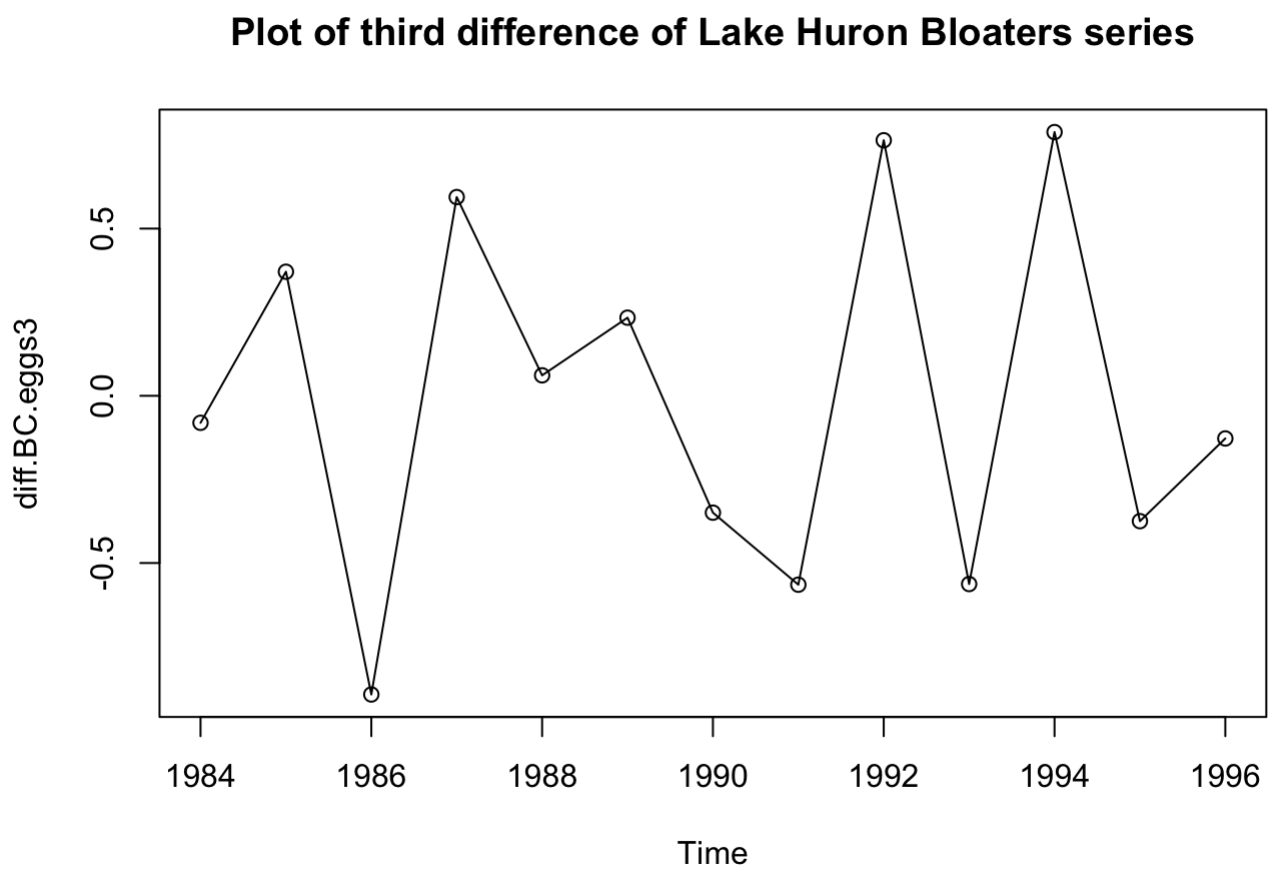

Figure 10:Series after third difference

This plot looks much better although as evidenced by the ADF unit root test below, the series is not yet stationary.

```
ar(diff(diff.BC.eggs3))
```

```
##
## Call:
## ar(x = diff(diff.BC.eggs3))
##
## Coefficients:
##       1        2        3        4
## -1.3533  -1.0853  -0.9144  -0.5759
##
## Order selected 4  sigma^2 estimated as  0.3188
```

Order for the test is obtained as 4.

```
adfTest(diff.BC.eggs3,lags = 4)
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 4
##   STATISTIC:
##     Dickey-Fuller: -1.2635
##   P VALUE:
##     0.2068
##
## Description:
##  Sun May  6 13:10:59 2018 by user:
```

With a p-value of 0.2068 we can conclude non-stationarity of the series at 5% level of significance. We shall therefore apply the fourth difference.

```
#Fourth difference
diff.BC.eggs4 = diff(BC.eggs, differences = 4)
ar(diff(diff.BC.eggs4))
```

```
##
## Call:
## ar(x = diff(diff.BC.eggs4))
##
## Coefficients:
##       1        2
## -1.3062  -0.5905
##
## Order selected 2  sigma^2 estimated as  1.079
```

Order for the test is obtained as 2.

```
adfTest(diff.BC.eggs4,lags = 2)
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 2
##   STATISTIC:
##     Dickey-Fuller: -2.2915
##   P VALUE:
##     0.02382
##
## Description:
##  Sun May  6 13:11:00 2018 by user:
```

With a p-value of 0.02382 we can now conclude stationarity of the series at 5% level of significance. We have enough evidence to reject the null hypothesis of stationarity.

```
plot(diff.BC.eggs4, type = "o", main = "Plot of the Fourth difference of Lake Huron B
loaters series")
```

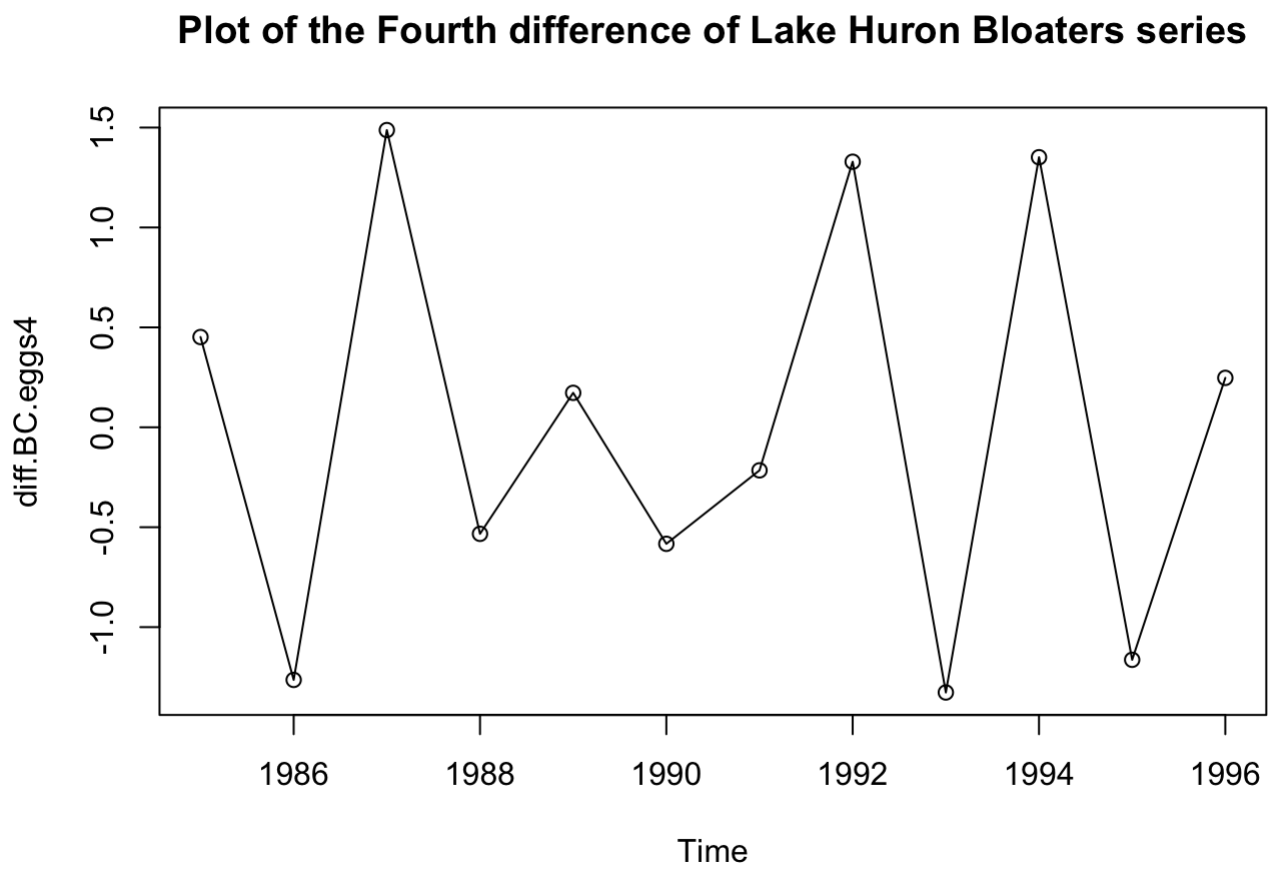

Figure 11:Series after fourth difference

This plot looks much better than the original series and we can see that the trend is no longer present. Although there are several extremes(both highs and lows) they do seem to be around a mean level.

Normality of the series after fourth differencing

```
qqnorm(diff.BC.eggs4,main='QQ plot of the series after fourth differencing')
qqline(diff.BC.eggs4,col=2)
```

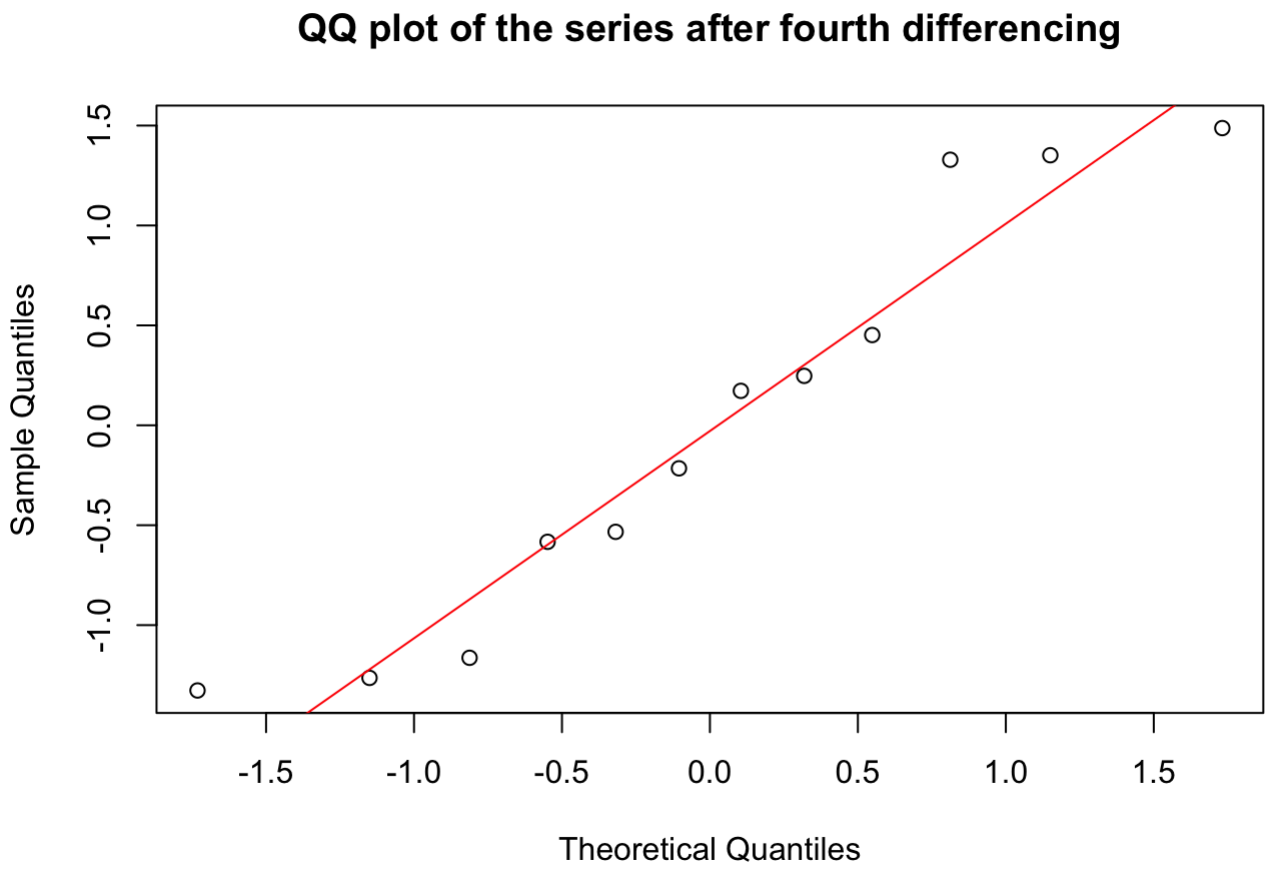

Figure 12:QQ plot of series after fourth difference

```
shapiro.test(diff.BC.eggs4)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  diff.BC.eggs4
## W = 0.91614, p-value = 0.2556
```

We observe from the QQ plot and Shapiro-Wilk test above that after the fourth difference, the series meets the assumption of normality. There is little departure from normality and with a p-value of 0.2556, we do not have enough evidence to reject the null hypothesis that the series is normally distributed.

We can observe a side by side QQ plot of normality after transformation and after differencing as shown below. It is clear that there is an improvement in normality after differencing.

```
par(mfrow=c(1,2))
qqnorm(diff.BC.eggs4,main='QQ plot of series after 4th diff')
qqline(diff.BC.eggs4,col=2)
qqnorm(BC.eggs, main='QQ plot of transformed series')
qqline(BC.eggs,col=2)
```

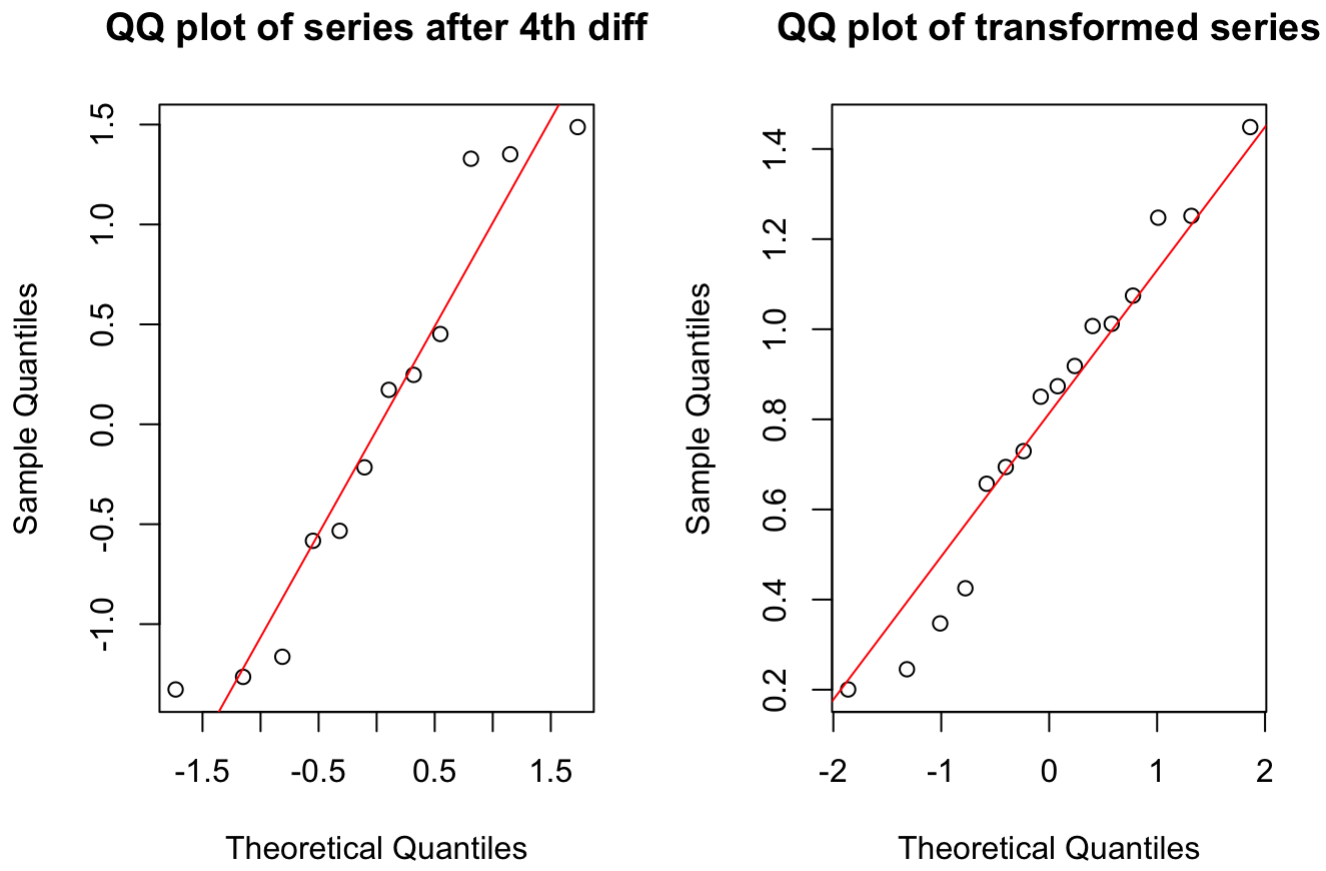

Figure 13:Comparison of normality

We see more observations closer to the trend line in the QQ plot after 4th diff and less outliers than in the QQ plot after transforming.

ACF and PACF for the fourth differenced series

```
par(mfrow=c(1,2))
acf(diff.BC.eggs4, main='Sample ACF')
pacf(diff.BC.eggs4, main='Sample PACF')
```

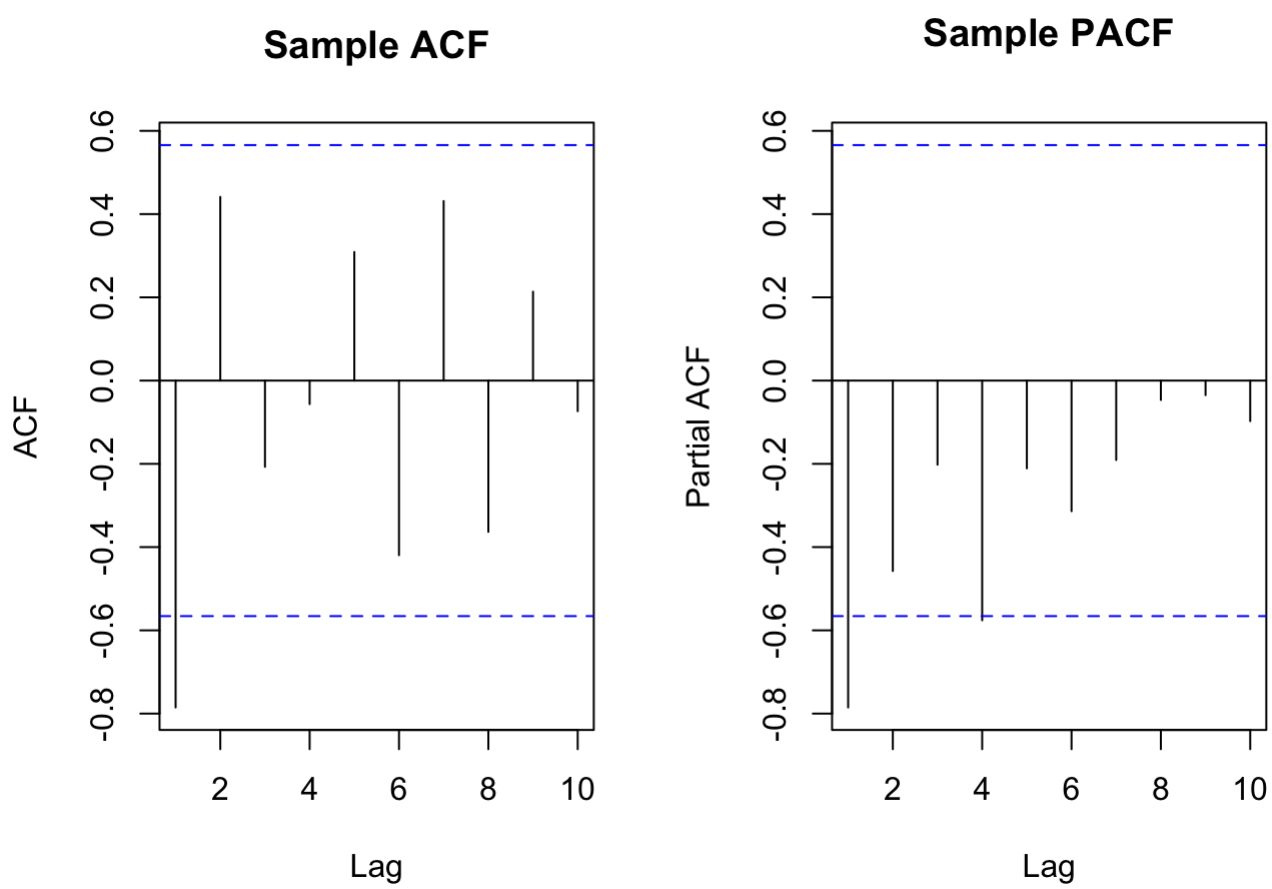

Figure 14:ACF and PACF after fourth difference

From ACF and PACF, we see one significant autocorrelation in ACF and one significant autocorrelation in PACF.So, possible models from here are { ARIMA(1,4,1), ARIMA(2,4,1), ARIMA(1,4,2)}.

We shall proceed with the EACF and BIC table to get other candidate models. We shall restrict the maximum number of AR and MA parameters in the EACF and BIC table because of the size of the series.

EACF

```
eacf(diff.BC.eggs4, ar.max = 2, ma.max = 3)
```

```
## AR/MA
##   0 1 2 3
## 0 x o o o
## 1 o o o o
## 2 o o o o
```

The top left ‘o’ symbol in EACF is located at the intersection of AR = 0 and MA= 1. In line with the downward vertex, AR would be 1 and 2 as well.The set of possible models become { ARIMA(0,4,1), ARIMA(1,4,1) and ARIMA(0,4,2)}.

BIC

```
par(mfrow=c(1,1))
res = armasubsets(y= diff.BC.eggs4,nar=4,nma=3,y.name='test',ar.method='ols')
plot(res)
```

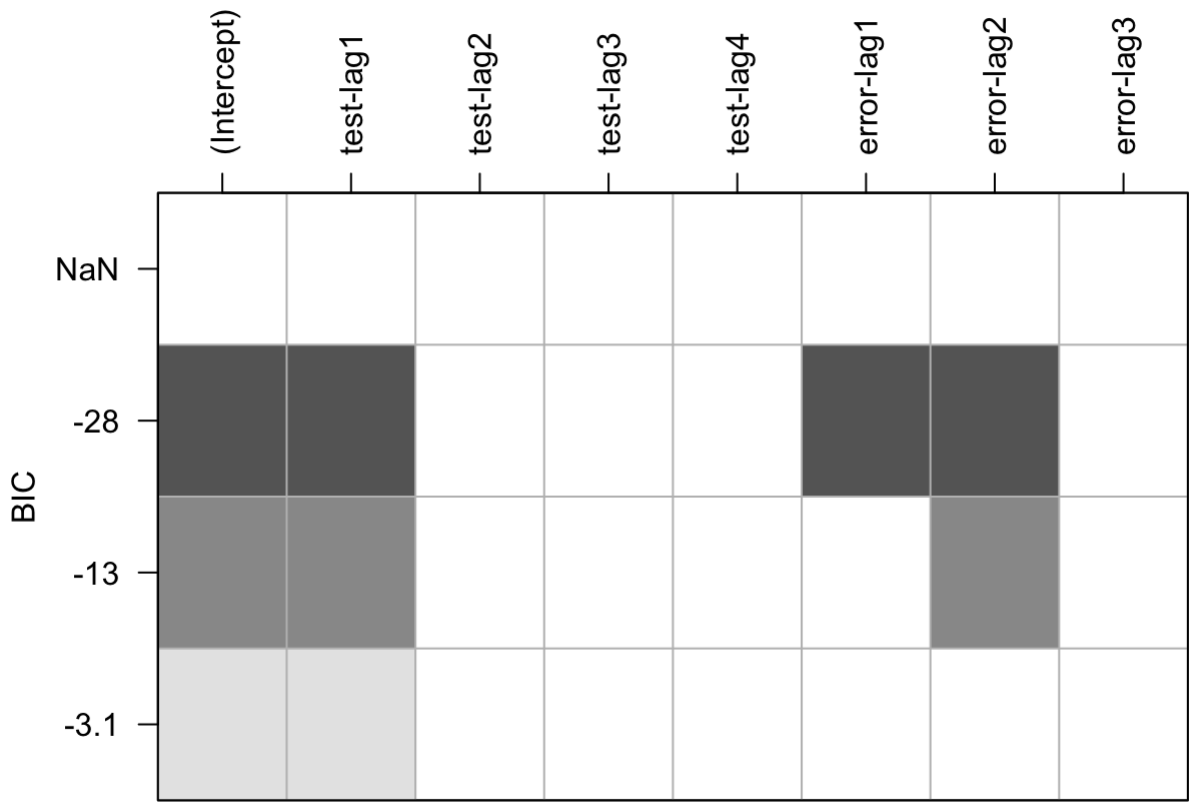

Figure 15: BIC Table

From the BIC table, we read the models are { AR(1), ARMA(1,1), ARMA(1,2), MA(1) and MA(2)}.In the context of the series, these should be read as { ARIMA(1,4,0), ARIMA(1,4,1), ARIMA(1,4,2), ARIMA(0,4,1) and ARIMA(0,4,2)} respectively.

In summary, our final candidate models are **{ ARIMA(1,4,0), ARIMA(1,4,1), ARIMA(1,4,2), ARIMA(2,4,1), ARIMA(0,4,1) and ARIMA(0,4,2)}**. We shall proceed to do the fit the models and find their parameter estimates.

Task 2: Finding the best fitting trend model of Lake Huron Bloaters series

ARIMA(1,4,0)

```
model140_ml = arima(eggs.ts, order = c(1,4,0), method = 'ML')
coeftest(model140_ml)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value  Pr(>|z|)
## ar1 -0.77523    0.14953 -5.1846 2.165e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
model_140_css = arima(eggs.ts,order=c(1,4,0),method='CSS')
coeftest(model_140_css)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value  Pr(>|z|)
## ar1 -0.82051    0.16507 -4.9708 6.669e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The ARIMA(1,4,0) model is significant with both the MLE and CSS.

ARIMA(0,4,1)

```
model041_ml = arima(eggs.ts, order = c(0,4,1), method = 'ML')
coeftest(model041_ml)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value  Pr(>|z|)
## ma1 -0.97878    0.20781   -4.71 2.477e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
model041_css = arima(eggs.ts, order = c(0,4,1), method = 'CSS')
coeftest(model041_css)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value  Pr(>|z|)
## ma1 -1.25372    0.10027 -12.504 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The ARIMA(0,4,1) model is significant with both the MLE and CSS.

ARIMA(0,4,2)

```
model042_ml = arima(eggs.ts, order = c(0,4,2), method = 'ML')
coeftest(model042_ml)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value  Pr(>|z|)
## ma1 -1.86751    0.32411 -5.7619 8.317e-09 ***
## ma2  0.94443    0.31996  2.9517  0.003161 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
model042_css = arima(eggs.ts, order = c(0,4,2), method = 'CSS')
coeftest(model042_css)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error  z value  Pr(>|z|)
## ma1 -2.29095    0.14287 -16.0351 < 2.2e-16 ***
## ma2  1.41267    0.16555   8.5331 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The ARIMA(0,4,2) is significant with both the MLE and CSS.

ARIMA(1,4,1)

```
model141_ml = arima(eggs.ts, order = c(1,4,1), method = 'ML')
coeftest(model141_ml)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value  Pr(>|z|)
## ar1 -0.63807    0.19490 -3.2739  0.001061 **
## ma1 -0.97188    0.23748 -4.0924 4.269e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
model141_css = arima(eggs.ts, order = c(1,4,1), method = 'CSS')
coeftest(model141_css)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value  Pr(>|z|)
## ar1 -0.71015    0.21880 -3.2456  0.001172 **
## ma1 -0.90595    0.12263 -7.3877 1.494e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Both the AR(1) and MA(1) component in the ARIMA(1,4,1) are significant.

ARIMA(1,4,2)

```
model142_ml = arima(eggs.ts, order = c(1,4,2), method = 'ML')
coeftest(model142_ml)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value  Pr(>|z|)
## ar1 -0.32703    0.27816 -1.1757   0.23972
## ma1 -1.82555    0.45083 -4.0493 5.136e-05 ***
## ma2  0.87597    0.46001  1.9042   0.05688 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
model142_css = arima(eggs.ts, order = c(1,4,2), method = 'CSS')
coeftest(model142_css)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value Pr(>|z|)
## ar1 -0.53170    0.36720 -1.4480 0.147627
## ma1 -1.30688    0.49188 -2.6569 0.007886 **
## ma2  0.43860    0.55530  0.7898 0.429623
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Only the MA(1) component in the ARIMA(1,4,2) is significant. The MLE and CSS both agree on this. This model may also be considered to be an overfit of the ARIMA(1,4,1) as we see several components become insignificant once the second MA component is added. We shall not include this model.

ARIMA(2,4,1)

```
model241_ml = arima(eggs.ts, order = c(2,4,1), method = 'ML')
coeftest(model241_ml)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value  Pr(>|z|)
## ar1 -0.80414    0.28944 -2.7782 0.0054656 **
## ar2 -0.21425    0.28764 -0.7448 0.4563716
## ma1 -0.96672    0.25713 -3.7596 0.0001702 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
model241_css = arima(eggs.ts, order = c(2,4,1), method = 'CSS')
coeftest(model241_css)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value  Pr(>|z|)
## ar1 -1.02602    0.30042 -3.4153 0.0006371 ***
## ar2 -0.38402    0.31022 -1.2379 0.2157511
## ma1 -0.79038    0.17250 -4.5818 4.609e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Only the AR(1) and MA(1) component in this model are significant. The MLE and CSS both arrive at this conclusion. Similarly, this model may also be considered to be an overfit of the ARIMA(1,4,1) as we see several components become insignificant once the second AR component is added. We shall not also include this model.

We will consider a ranking of the AIC and BIC values of the possible candidate models to decide the best one.

```
# AIC and BIC values
sort.score (AIC(model140_ml, model041_ml, model042_ml, model141_ml), score = "aic")
```

```
##             df      AIC
## model042_ml  3 42.84286
## model141_ml  3 44.37815
## model140_ml  2 48.87321
## model041_ml  2 49.07553
```

```
sort.score (BIC(model140_ml, model041_ml, model042_ml, model141_ml), score = "bic" )
```

```
##             df      BIC
## model042_ml  3 44.29758
## model141_ml  3 45.83287
## model140_ml  2 49.84302
## model041_ml  2 50.04535
```

We observe the smallest AIC with model ARIMA(0,4,2), which is 42.84286, and smallest BIC with model ARIMA(0,4,2), which is 44.29758. AIC and BIC agree on the model ARIMA(0,4,2).

Overfitting the model

We will try over-fitting with ARIMA(1,4,2) and ARIMA(0,4,3) models.

We already had ARIMA(1,4,2) but it was not promising as only the MA(1) component was significant.

ARIMA(0,4,3)

```
model043_ml = arima(eggs.ts, order = c(0,4,3), method = 'ML')
coeftest(model043_ml)
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error z value Pr(>|z|)
## ma1 -1.945451   0.409099 -4.7555 1.98e-06 ***
## ma2  1.040986   0.591817  1.7590  0.07858 .
## ma3 -0.021509   0.296991 -0.0724  0.94226
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
model043_css = arima(eggs.ts, order = c(0,4,3), method = 'CSS')
coeftest(model043_css)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value  Pr(>|z|)
## ma1 -2.71216    0.31641 -8.5718 < 2.2e-16 ***
## ma2  2.48145    0.75750  3.2758  0.001054 **
## ma3 -0.72038    0.49810 -1.4463  0.148100
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The MLE method gives a possible convergence warning on this model and only MA(1) is significant. In the CSS method, only MA(1) and MA(2) are significant and so we should stop there.

Residual Analysis

We shall use the residuals of the series to test the goodness of fit of the candidate models. We shall begin with the ARIMA(0,4,2) as it is the most promising model.

ARIMA (0,4,2)

```
par(mfrow=c(1,1))
residual.analysis(model = model042_ml)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res.model
## W = 0.91548, p-value = 0.1426
```

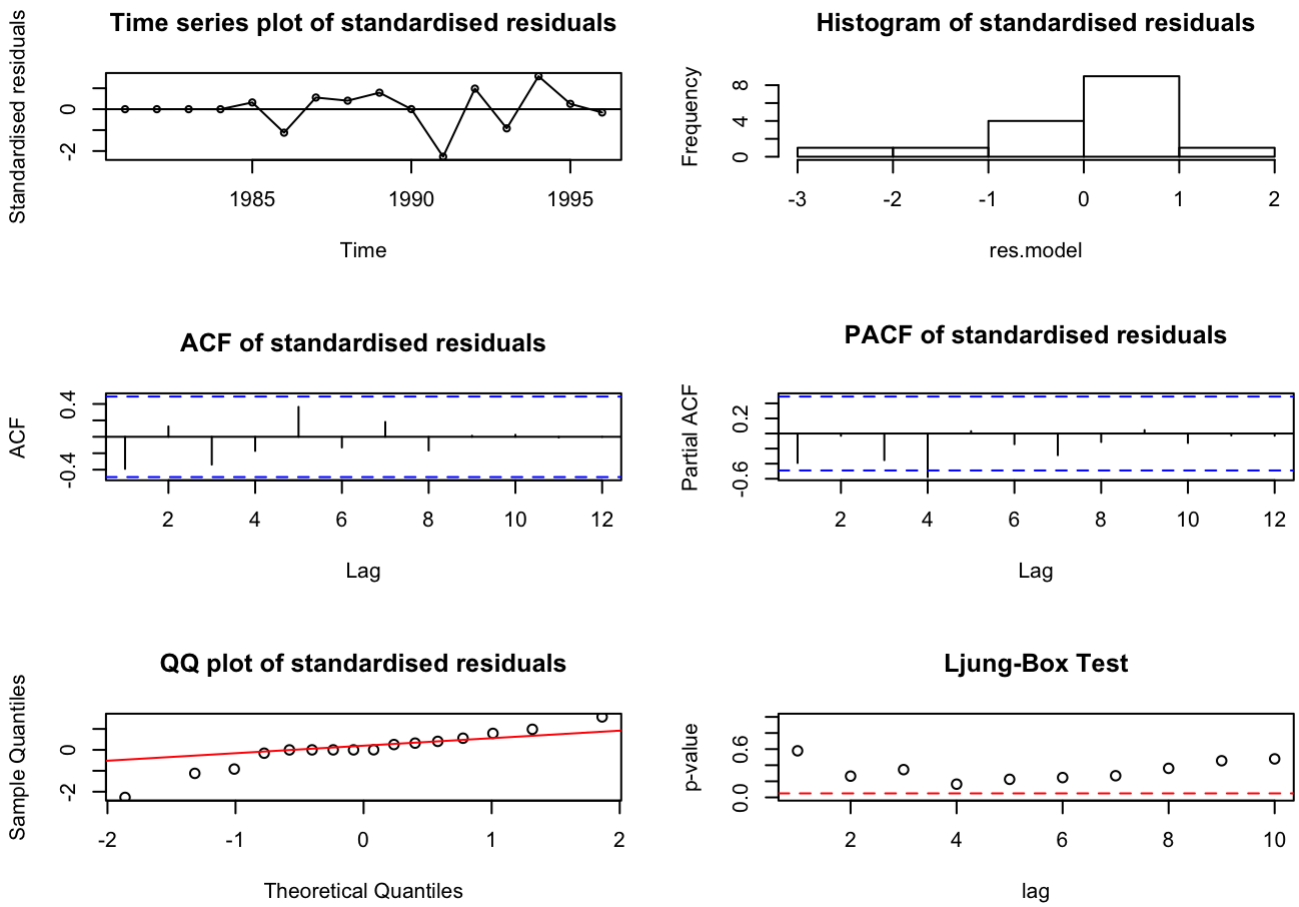

Figure 17:Residual analysis for ARIMA(0,4,2)

What are our observations from the above plots?

a)Plot of the residuals over time: There is no trend observed and we see the residuals scatter around the zero horizontal level. However, we see an increased variation towards the end of the series which is not ideal.

b)Histogram: The residuals follow a symmetric distribution as expected in a normal distribution. It is not perfect but this could be attributed to the small sample size.

c)ACF and PACF: These resemble a white noise series in general. We notice a distant significant autocorrelation in the PACF which is a problem.

c)QQplot: There are no serious departures from normality and most of the residuals fall along the straight line. However, we see that there are several outliers.

d)Ljung-box test: We have no evidence to reject the null hypothesis that the error terms are uncorrelated. However, we do note several observations very close to the significance value.

d)Shapiro-Wilk test: With a p value of 0.1426, we can conclude that the null hypothesis of normality is upheld.

ARIMA(1,4,0)

```
par(mfrow=c(1,1))
residual.analysis(model = model140_ml)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res.model
## W = 0.9185, p-value = 0.1595
```

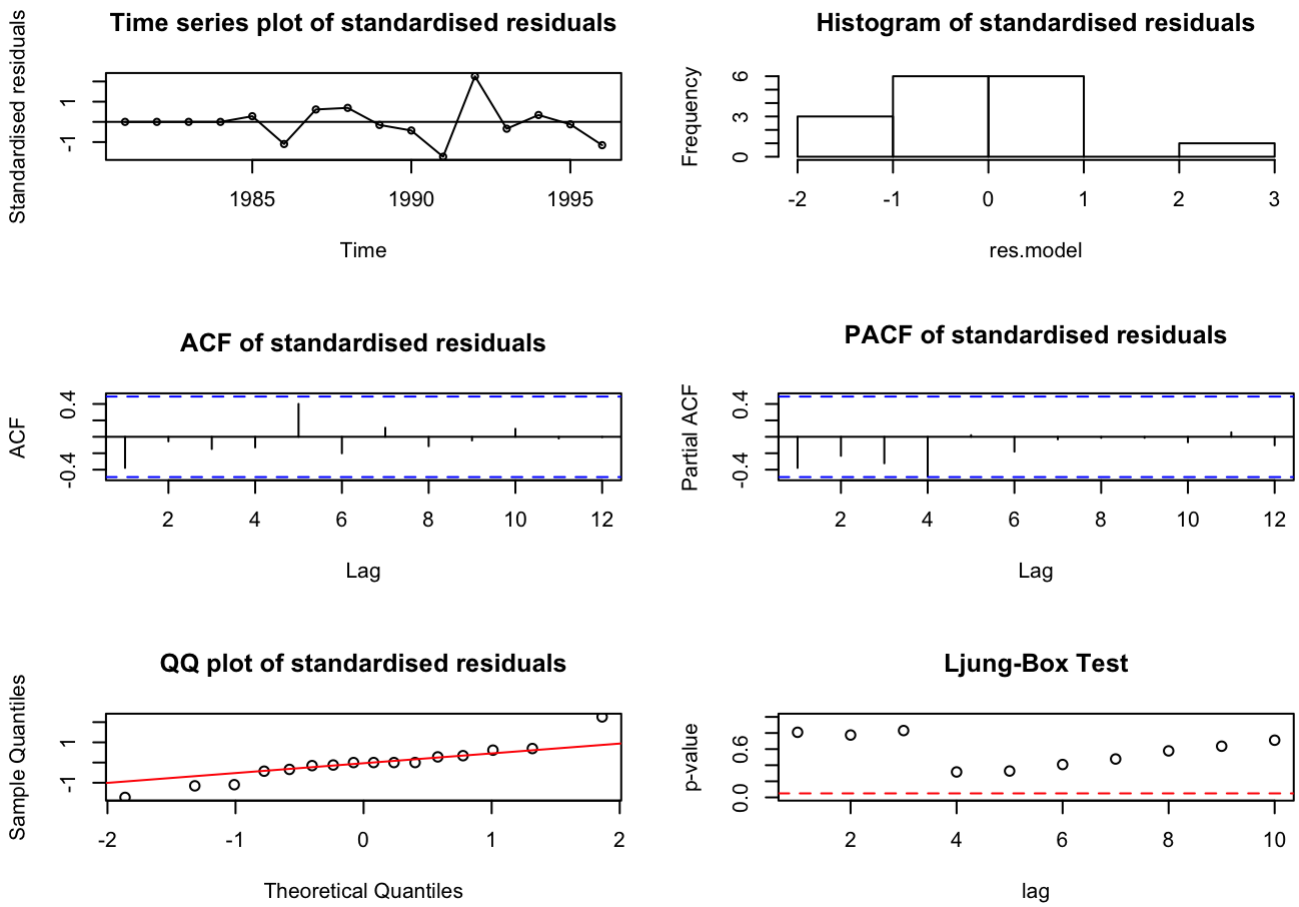

Figure 18:Residual analysis for ARIMA(1,4,0)

There seems to be a little variation in the plot of the residuals over time specifically in the middle and close to the end of the series which is not ideal. Trend is also absent. The histogram is not symmetric and could be inferred to be skewed to the left. The ACF and PACF resemble a white noise series although we notice an almost significant lag in the PACF. The normality of the plot is upheld as can be seen from the p-value of the Shapiro test, 0.1595, which is greater than the 0.05 significance level. The residuals from this ARIMA(1,4,0) stick close to the trend line with only several outliers.

ARIMA(0,4,1)

```
par(mfrow=c(1,1))
residual.analysis(model = model041_ml)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res.model
## W = 0.95241, p-value = 0.5288
```

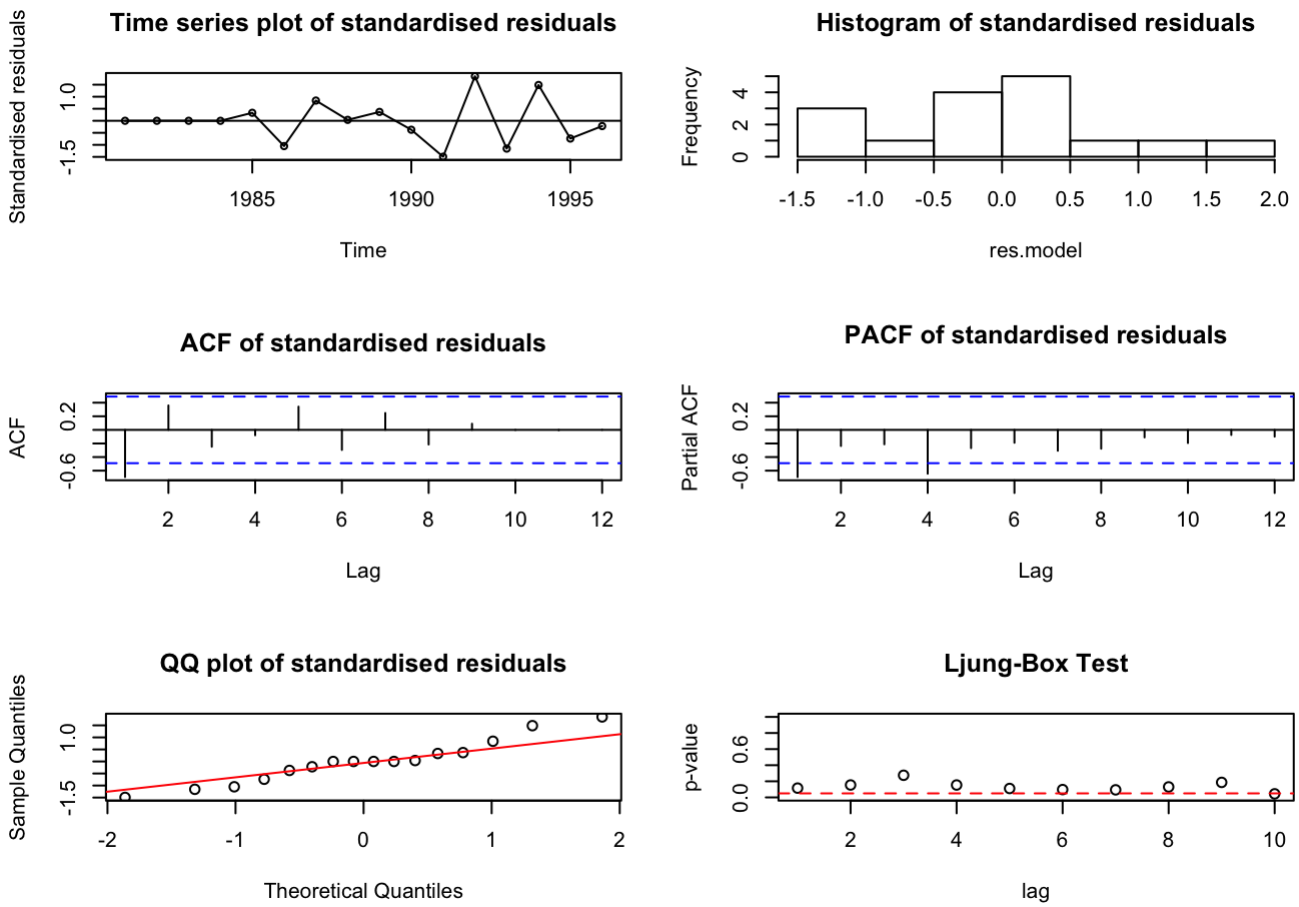

Figure 19:Residual analysis for ARIMA(0,4,1)

There does not seem to be a trend in the plot of the residuals over time although we see a little variation in the beginning and in the middle of the series. The histogram is not symmetric and most observations seem to be binned into the first half of the plot. The ACF and PACF do not resemble a white noise series as we observe one significant autocorrelation in the ACF and one significant autocorrelation in the PACF. The latter also has a distant significant autocorrelation. The normality of the plot is upheld as can be seen from the p-value of the Shapiro test, 0.5228, which is greater than the 0.05 significance level. We therefore cannot reject the null hypothesis that the residuals are normally distributed. The residuals from this ARIMA(0,4,1) stick close to the trend line with only several outliers. It appears that the residuals may have some correlation in the error terms as shown in the Ljung-Box test.

ARIMA(1,4,1)

```
par(mfrow=c(1,1))
residual.analysis(model = model141_ml)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res.model
## W = 0.8858, p-value = 0.04783
```

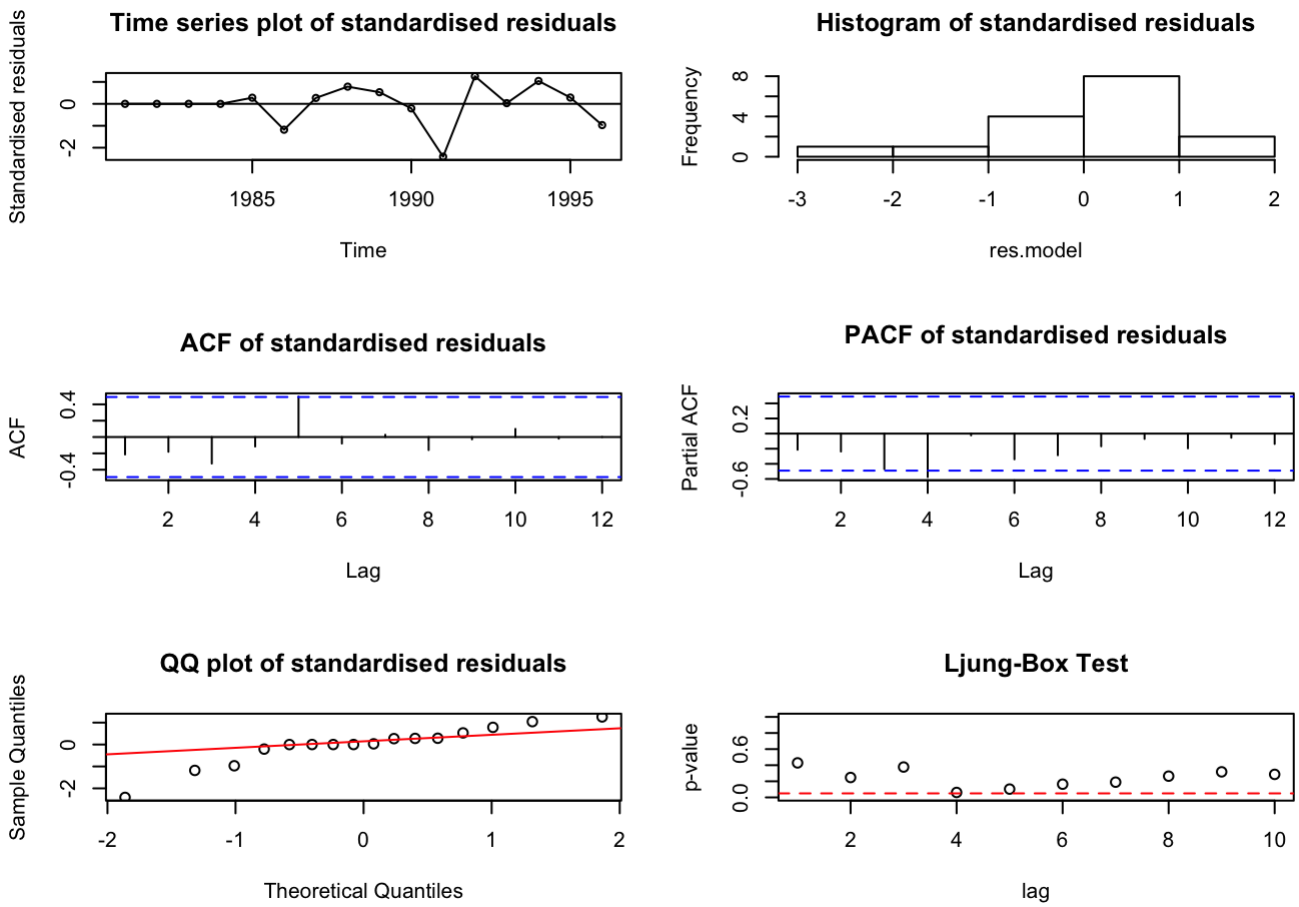

Figure 20:Residual analysis for ARIMA(1,4,1)

There does not seem to be a trend in the plot of the residuals over time although we see large variations in the middle of the series. The histogram is not symmetric and most observations seem to be binned into the second half of the plot. There seems to be a right skew in the data. The ACF and PACF almost resemble a white noise series as we observe distant significant autocorrelations in the ACF and PACF. The Shapiro test reveals a p-value of 0.04783 which is less than the 0.05 significance level. We therefore have enough evidence to statistically reject the null hypothesis that the residuals are normally distributed. The outliers that can be seen in the QQ plot seem to have affected the normality of the residuals. We also see a correlation in the error terms as shown in the Ljung-Box test. Based on the residual analysis, this model should no longer be considered.

Overal, each of the models above has an issue in its residuals. However, the ARIMA(0,4,2) is the most likely to be correct as its residuals behave roughly like independent, identically distributed normal variables with zero means and common standard deviations.It has its inadequacies but is the best in light of the above.

Task 3: Forecasts of egg depositions of Lake Huron Bloaters for the next 5 years

Using the ARIMA model(0,4,2) discussed earlier, we can estimate the egg depositions for the next 5 years based on the historical data available.

```
library(forecast)
```

```
##
## Attaching package: 'forecast'
```

```
## The following object is masked from 'package:FitAR':
##
##     BoxCox
```

```
## The following object is masked from 'package:nlme':
##
##     getResponse
```

```
fit = Arima(eggs.ts,c(0,4,2),lambda = 0.5)
plot(forecast(fit,h=10),xlim=c(1980,2000),ylim=c(-1,3),main = "Prediction of egg depo
sitions for the next five years", ylab = "Egg depositions(in millions)")
```

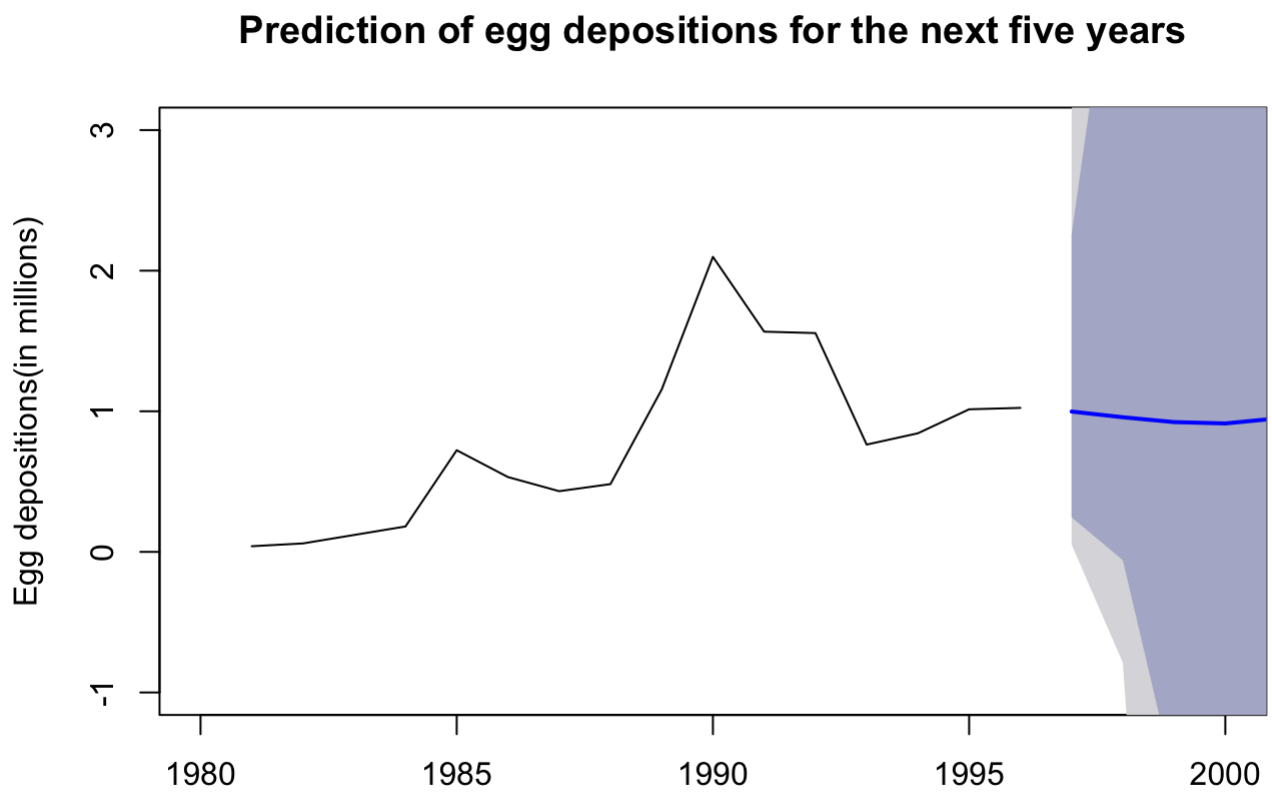

Figure 21:Forcast using ARIMA(0,4,2)

It is important to highlight that although the confidence bound seems to blow out, this is not a problem because the uncertainty increases exponentially as we move far from the last observed time point.

Conclusion

The ARIMA(0,4,2) model is the best fitting model according to our observations above. It has a good goodness of fit and was better than all the other candidate models in terms of their AIC and BIC scores. The model shows some inadequacy during residual analysis although we observe that it is better than the other candidate models. In addition, it is a considerably large model. It is possible that the limited observations negatively impact finding the best model. In closing, although this is the best model with the data at hand, there is definitely room for improvement.