Model fitting and forecast of Egg deposition

ASSIGNMENT 2 : TIME SERIES ANALYSIS [MATH1318]

NAME : SYED WAJAHATH

STUDENT ID : S3750039

DATE : 17 – 04 - 2020

CONTENTS

INTRODUCTION

Bloater fish is a freshwater whitefish belonging to the family Salmonidae. It is commonly found in underwater slopes and great lakes.

The report discusses the **analysis of egg deposition of Lake Huron Bloasters** using different methods covered in the course. The report also discusses suitable elements of a given time series. The ultimate goal of the report is to forecast the deposition of eggs over the next five periods using the best model of a set of possible models. **The dataset used here depicts EGG DEPOSITION (IN MILLIONS) of AGE-3 from the year 1981 to 1996 and is available in FSAdata package.** The report discusses only the models suitable for STOCASTIC TREND.

The set of possible models to forecast is selected based on the following parameters;

- Normality test
- Transformations
- Differences
- ACF-PACF
- EACF
- BIC table
- Parameter estimations --- MaxLikelihood/Conditional sum of squares
- Model Diagnostics --- Residual Analysis

1. OBJECTIVE

The aim of the task is to forecast the deposition of eggs of Lake Huron Bloaters using the Parameter estimation methods and Model Diagnostics. The task uses the dataset listed under FSAdata package which has related data clustered over the period of 14 years from 1982 to 1996.

The first step of any time-related data is to load and visualize the obtained series by checking for 1. Trend 2. Seasonality 3. Variance 4. AR/MA behaviour 5. Intervention points

1.1. MODEL FITTING

The below figure1, represents the time series plot of the raw data for egg depositions.

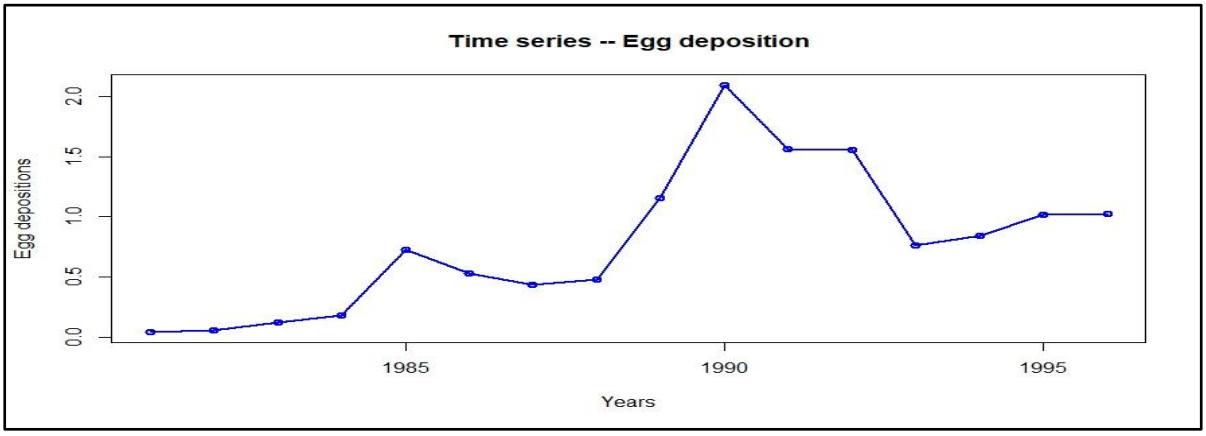


Figure 1 --- Time series plot

The characteristics of time series plot are as follows;

- **Trend:** It is clearly visible that time series follows a upward trend all the way with a gradual drop around 1993.
- **Seasonality:** Lack of repeated patterns confirms the absence of seasonality in the series.
- **AR/MA behaviour:** With depositions following each other, one could say series follows AR behaviour.
- **Change in Variance:** No signs of change in variance is seen.
- **Intervention point:** There are no intervention points in the series.

We then check for correlation of egg deposition with its previous year deposition. This is depicted using a scatter plot as in figure 2. A clear upward trend could be observed with data points aligned in a pattern. A correlation on the higher side is observed **0.7445**

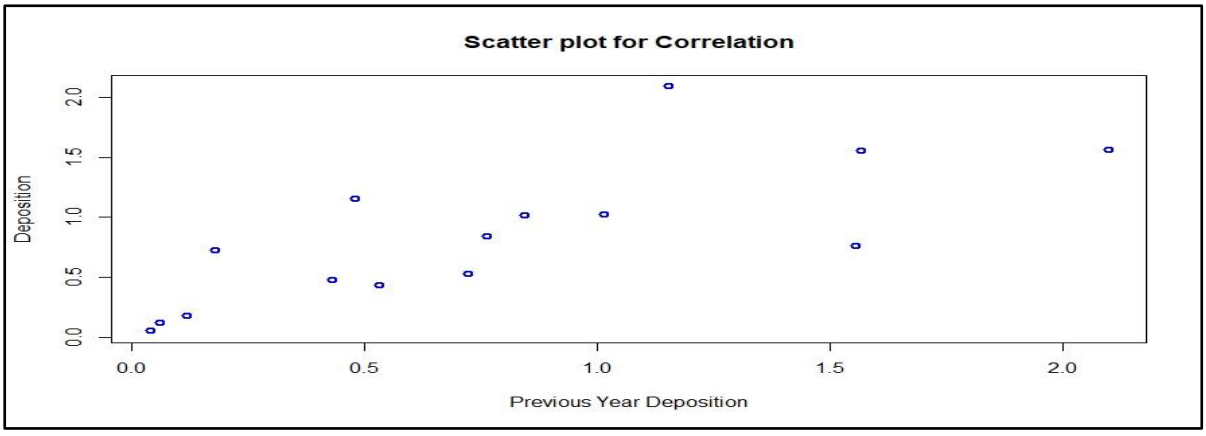


Figure 2 --- Correlation check

We then look for autocorrelation using ACF and PACF plots as in figure 3.

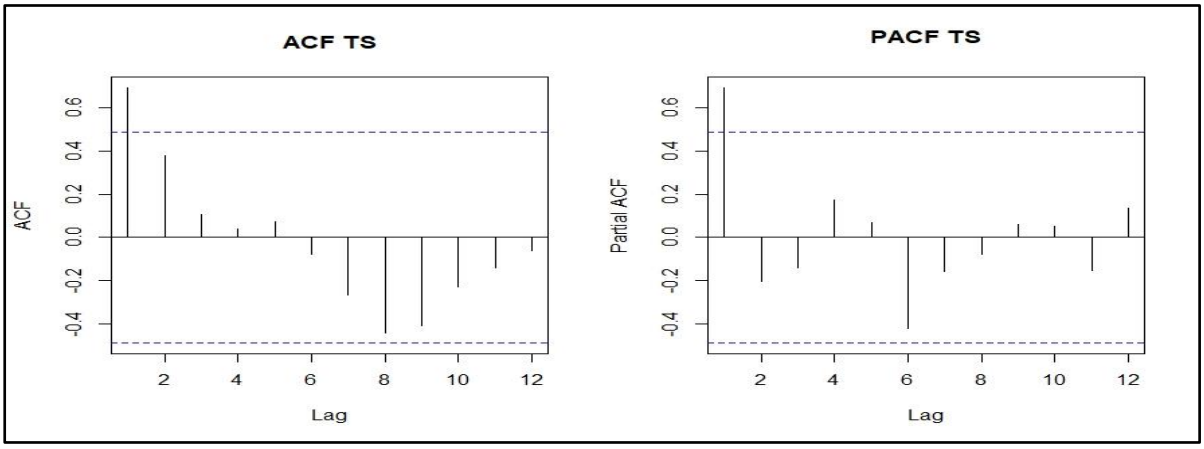


Figure 3 --- ACF & PACF of time series

- The slow decaying pattern of ACF and a tall line at lag 1 of PACF suggests the existence of non-stationarity in the series.
- Also, the visuals of time series suggests us the existence of trend. Thereby, we check for normality of the series.

1.2. NORMALITY AND STATIONARITY

As we find the series to be non-stationary, we perform a stationarity check using ADF test for confirmation and also a normality test using Shapiro-wilks test.

```
Shapiro-Wilk normality test

data:  eggs
W = 0.94201, p-value = 0.3744
```

```
Augmented Dickey-Fuller Test

data:  eggs
Dickey-Fuller = -2.0669, Lag order = 2, p-value = 0.5469
alternative hypothesis: stationary
```

Figure 4 --- Normality and Stationarity check

With p-value so less in normality test and higher in ADF test, it confirms us that data isn't normally distributed and series has non-stationarity. Therefore, we procced to perform transformation.

1.3. TRANSFORMATION

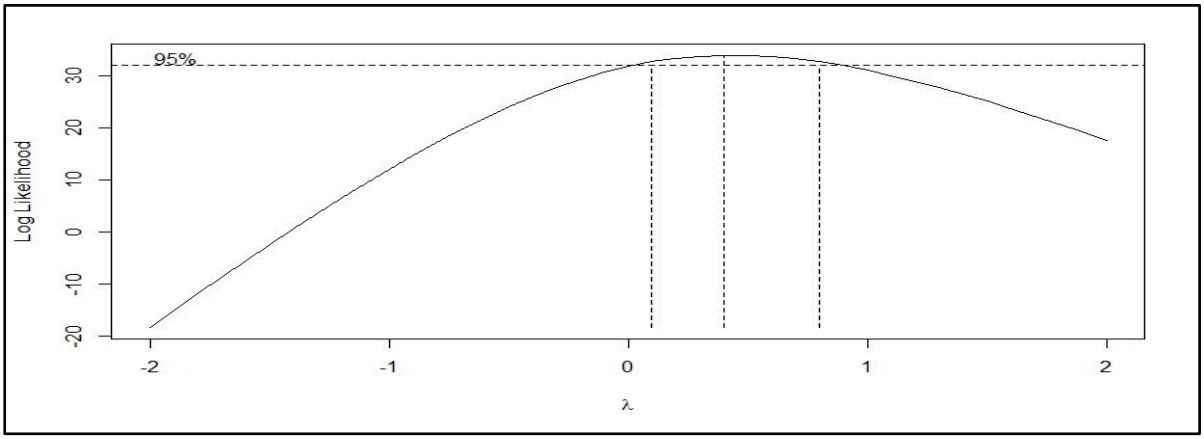


Figure 5 --- Confidence Interval

Transformations are of various forms which are applied to detrend the series. We used a box-cox transformation to de-trend the series. Figure 5, suggests the confidence interval to pick our lambda value. **`Yule-Walker`** method is used to transform.

The confidence interval obtained after yule-walker;s box-cox transformation is 0.1 – 0.9 which results in a midpoint of 0.45 which is near 0.5. So, we perform square root transformation. The transformed series still has some trend and is non-stationary as seen in figure 6.

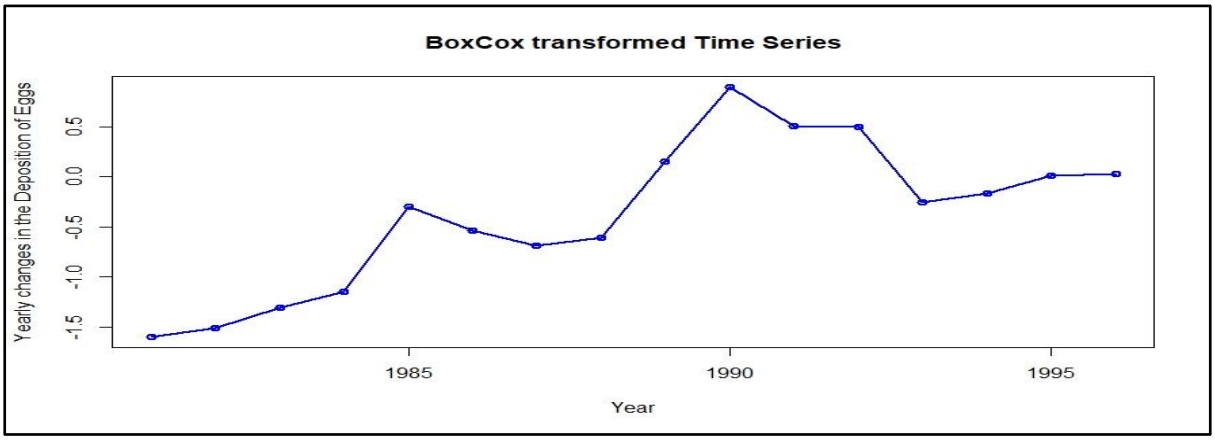
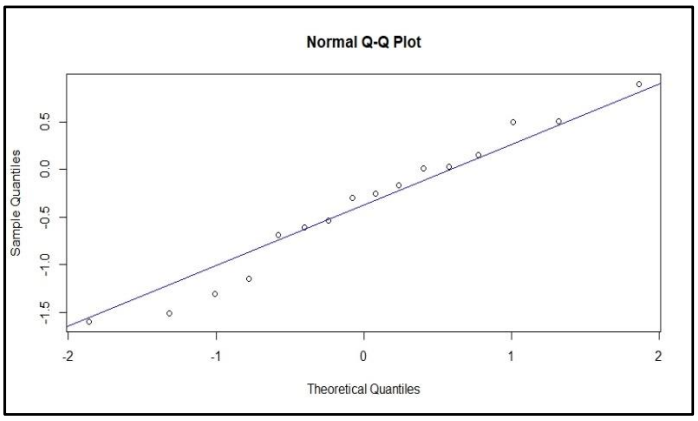


Figure 6 --- Transformed series

A qq-normal test and shapiro-wilk's test is considered for confirmation which affirmatively confirms our observation that series is not normal. The distributions towards the tail end of qq-plot do not lie with mean level. The normality p-value is neither good with 0.7636.(Figure 7) Therefore, we perform differencing.




```
Shapiro-Wilk normality
test

data:  eggs.trans
W = 0.96562, p-value =
0.7636
```

Figure 7 --- Normality test of transformed series

1.4. DIFFERENCING

As the series is still not de-trended and has non-stationarity, we perform differencing. We begin with a difference order of 1 which has quite a significant p-value of 0.0443 (under the significance of 0.05) resulting in stationarity of series. Curious to find a better series, we tried differencing of orders 2 and 3 which were terribly failed. Surprisingly, the p-value turned to be insignificant (0.1254 and 0.4242 respectively). Figure 8 represents a differenced series of order 1. Figure 9(a)(b)(c) represents the p-value of the ADF test conducted on the difference of order 1, 2 and 3 respectively.

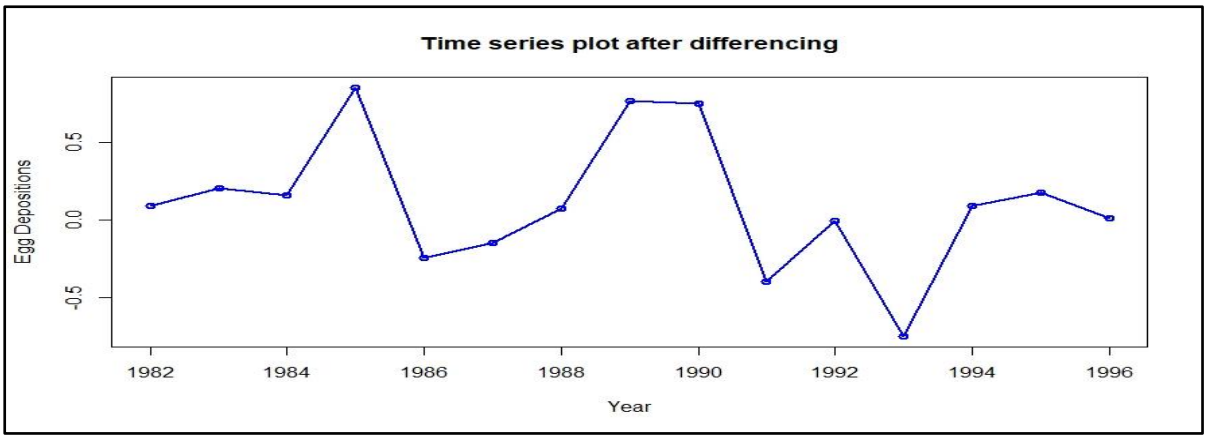


Figure 8 --- Differenced series

```
Augmented Dickey-Fuller Test

data:  diff.eggs.trans
Dickey-Fuller = -3.6096, Lag order = 2, p-value = 0.04931
alternative hypothesis: stationary
```

```
Augmented Dickey-Fuller Test

data:  diff.eggs.trans1
Dickey-Fuller = -3.1108, Lag order = 2, p-value = 0.1492
alternative hypothesis: stationary
```

```
Augmented Dickey-Fuller Test

data:  diff.eggs.trans2
Dickey-Fuller = -2.3442, Lag order = 2, p-value = 0.4412
alternative hypothesis: stationary
```

Figure 9 (a)(b)(c) --- ADF test results

The ACF and PACF of the series is complete white noise i,e without any significant lags. Thereby, resulting in no model prediction. Figure 10, represents ACF and PACF plots.

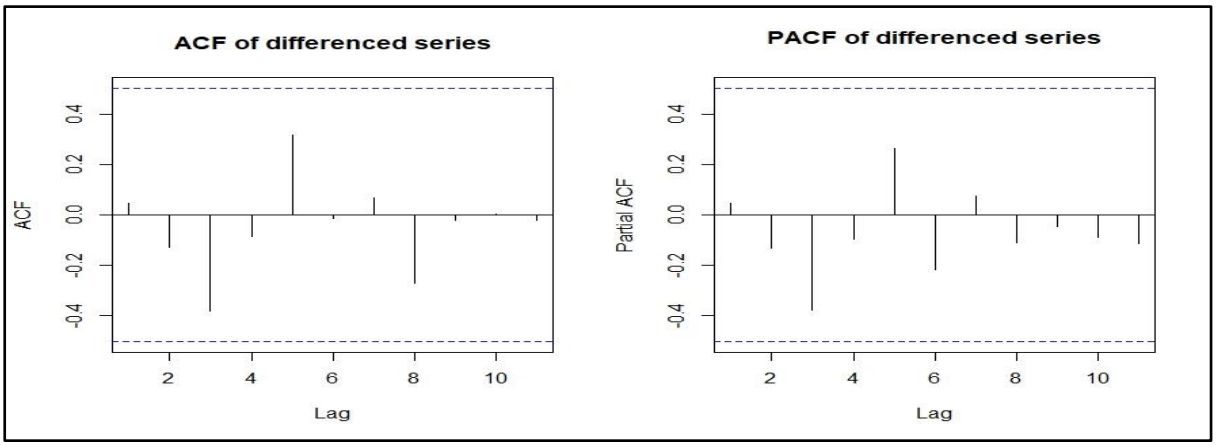


Figure 10 --- ACF & PACF of differenced series

1.5. EACF

```
AR/MA
  0 1 2 3
0 o o o o
1 o o o o
2 o o o o
3 o o o o
>
```

Figure 11 --- EACF

From the EACF matrix, the possible models are:
ARIMA(0,1,1),ARIMA(1,1,0),ARIMA(1,1,1), ARIMA(2,1,0),ARIMA(2,1,1),ARIMA(2,1,2), ARIMA(3,1,0),ARIMA(3,1,1),ARIMA(3,1,2)

1.6. BIC TABLE

From the below figure 12 which represents BIC table, model prediction seems to be easy. There seems no MA behaviour in our series. The models predicted are from the shaded columns. Therefore, predicted models from BIC are **ARIMA(1,1,0), ARIMA(2,1,0) and ARIMA(3,1,0).**

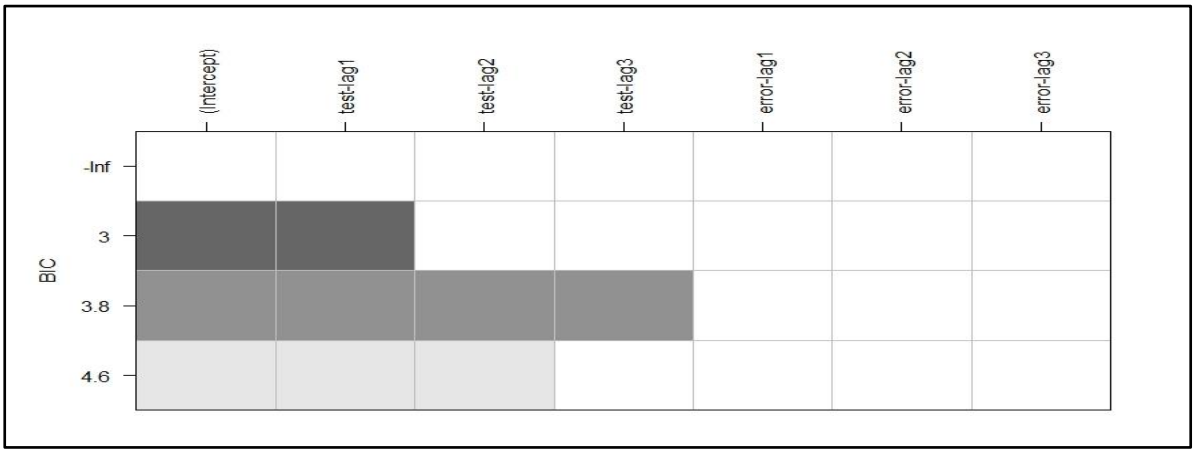


Figure 12 --- BIC table

With the possible models, we move on to predict the best model using the coefficients of parameters.

1.7. PARAMETER ESTIMATION ---- METHOD CSS AND ML

The below table represents the stats of significant models based on their parameters. With the majority of models with insignificant parameter values, we ignore them. We only have few significant models of our list. Off these models, only **mod_212_css** has all its parameters. Therefore, we consider this model for our further approximations.

Model	Method	AR(1)	AR(2)	AR(3)	MA(1)	MA(2)
ARIMA(2,1,1)	**CSS**	**<2e-16**	**0.3625**	**-**	**<2e-16**	**-**
ARIMA(2,1,2)	**CSS**	**5.913e-15**	**3.072e-15**	**-**	**0.001125**	**0.030193**
ARIMA(2,1,2)	**ML**	**0.005342**	**<2.2e-16**	**-**	**0.022622**	**NA**
ARIMA(3,1,2)	**CSS**	**2.519e-06**	**0.02976**	**0.77707**	**<2.2e-16**	**<2.2e-16**
ARIMA(3,1,2)	**ML**	**0.575602**	**0.003108**	**0.513640**	**0.635943**	**0.041246**

Table 1 ---- Significant models

1.8. AIC and BIC

Below are the AIC and BIC values of all the models using the parameter estimation method of Maximum Likelihood. **Mod_011_ml** has the lowest AIC and BIC values. Also, the coefficients of its parameters are insignificant. So we choose to ignore models based on ML.

```
            df      AIC
mod_011_ml   2 21.08226
mod_110_ml   2 21.09570
mod_210_ml   3 23.00931
mod_111_ml   3 23.08218
mod_310_ml   4 23.77666
mod_212_ml   5 24.54829
mod_211_ml   4 24.72960
mod_311_ml   5 25.77581
mod_312_ml   6 26.31545
>
```

```
            df      BIC
mod_011_ml   2 22.49836
mod_110_ml   2 22.51180
mod_210_ml   3 25.13346
mod_111_ml   3 25.20633
mod_310_ml   4 26.60887
mod_211_ml   4 27.56180
mod_212_ml   5 28.08854
mod_311_ml   5 29.31606
mod_312_ml   6 30.56375
>
```

Figure 13 --- AIC and BIC values

1.9. MODEL DIAGNOSTICS

1.9.1. OVERFITTING

With all the methods completed, **ARIMA(2,1,2)** seems to be the best model. Now, let us try overfitting the model by increasing the AR component by 1 and then MA by 1 i,e **ARIMA(2,1,3) and ARIMA(3,1,2). In ARIMA(3,1,2) we find AR(3) to be insignificant. Thus we ignore it. Now, with all the parameters having significant values in ARIMA(2,1,3), we consider both ARIMA(2,1,2) and ARIMA(2,1,3) for residual analysis.** Figure 14 represents the overfitting.

```
> # ARIMA(3,1,2)
> mod_312_css = arima(eggs.trans,order=c(3,1,2),method='CSS')
> coeftest(mod_312_css)

z test of coefficients:

     Estimate Std. Error  z value  Pr(>|z|)
ar1  1.227801   0.260868   4.7066 2.519e-06 ***
ar2 -0.776994   0.357515  -2.1733   0.02976 *
ar3  0.084906   0.299879   0.2831   0.77707
ma1 -1.794122   0.110744 -16.2007 < 2.2e-16 ***
ma2  1.515016   0.120668  12.5552 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> # ARIMA(2,1,3)
> mod_213_css = arima(eggs.trans,order=c(2,1,3),method='CSS')
> coeftest(mod_213_css)

z test of coefficients:

    Estimate Std. Error z value  Pr(>|z|)
ar1 -1.54767    0.21935 -7.0558 1.716e-12 ***
ar2 -0.81772    0.19783 -4.1335 3.573e-05 ***
ma1  2.09946    0.11247 18.6665 < 2.2e-16 ***
ma2  2.49814    0.22635 11.0368 < 2.2e-16 ***
ma3  1.32789    0.22253  5.9672 2.414e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> |
```

Figure 14 --- Overfitting

1.9.2. RESIDUAL ANALYSIS

In the residual analysis, we look for the distribution and variation of standardised residuals. With CSS models being considered, the normality of distribution doesn’t count in the process. ACF and PACF plots should be similar to that of a white noise series. Distributions below the guide line in QQ plot are considered significant. In the context, figure 15 represents residual analysis of ARIMA(2,1,2) and figure 16 respresents residual analysis of ARIMA(2,1,3).

From figure 15, one could observe the resdiuals arent distributed normally which however doesn't matter. The Time series plot however has distributions spread across due to change in variance. The ACF and PACF plots resemble white noise series.

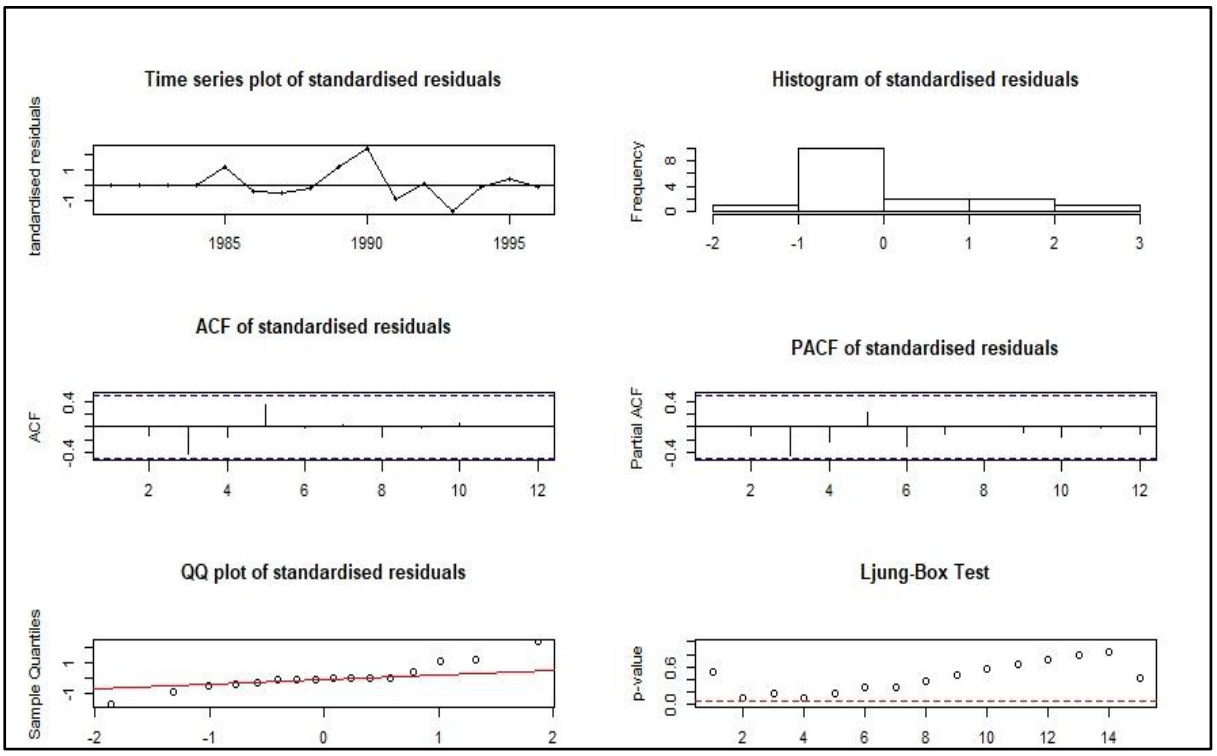


Figure 15 --- Residual Analysis ARIMA(2,1,2)

From figure 16, distribution of ARIMA(2,1,3) could be observed which are almost the same as ARIMA(2,1,2)

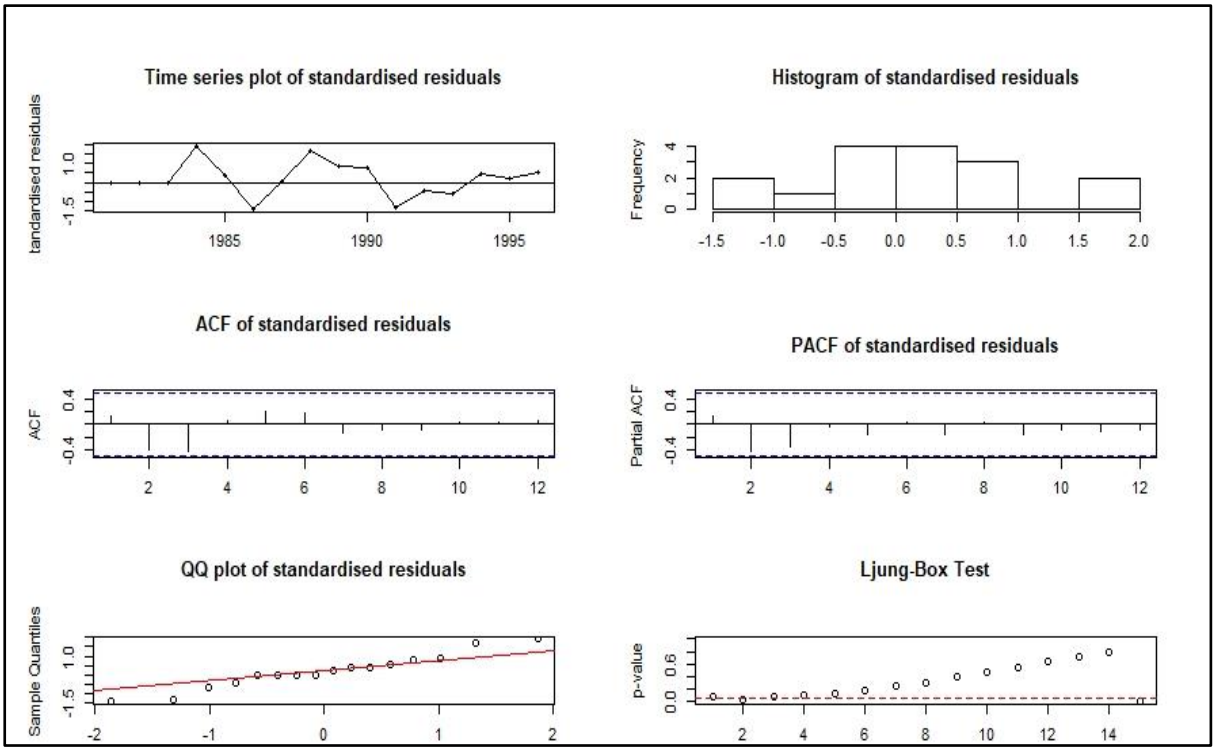


Figure 16 --- Residual Analysis ARIMA(2,1,3)

Therefore, based on principle of parsimony, we choose ARIMA(2,1,2) for forecasting the egg deposition.

2. MODEL SPECIFICATION

Figure 17, represents the forecast of Egg deposition over next five years using ARIMA(2,1,2).

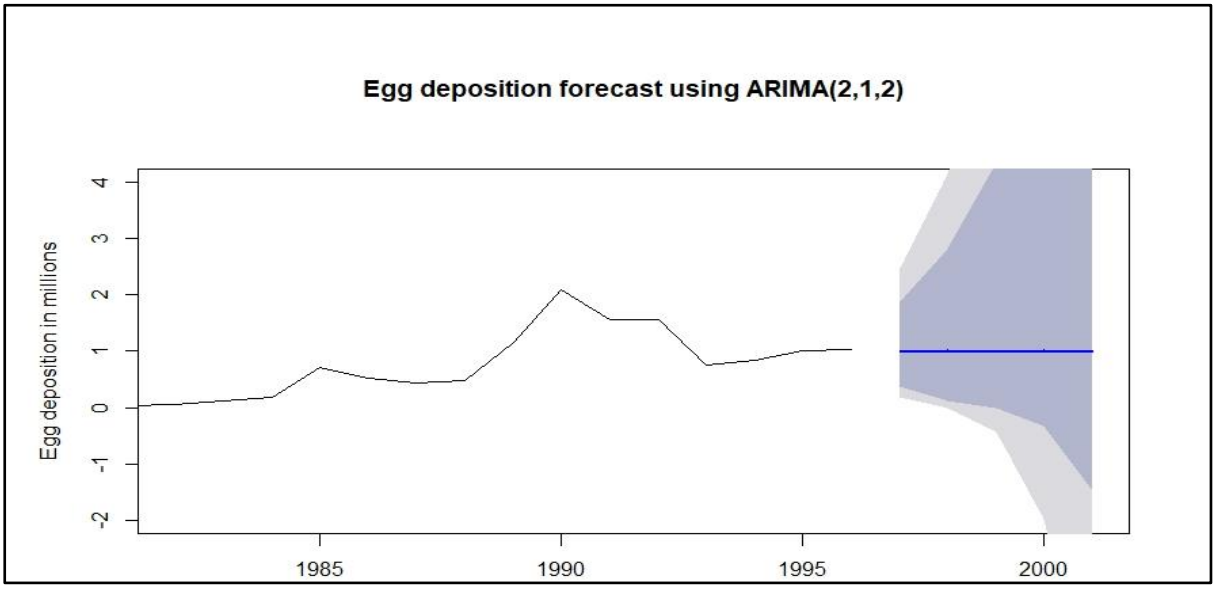


Figure 17 --- Forecast

3. DISCUSSION

- **The given series is stochastic and very short, Therefore, did not check model fitting for any of the deterministic models.**
- **With series being very short, change in variance could not be dealt properly.**
- **The ACF and PACF plots resembled white noise with lack of observations.**
- **EACF suggested every possible model with matrix being 0 at every point.**
- **However, BIC table ruled out any possibility of MA models.**
- **The parameter estimation techniques ruled out the possible models because of insignificant coefficients.**
- **ARIMA(2,1,2) using CSS was found to be the optimal of all possible models.**
- **Utility functions --- sort.score, and residual.analysis has been used for the assignment.**

4. CONCLUSION

Based on all the plots and tables, we find ARIMA(2,1,2) model to be the best in order to forecast the deposition of Bloater eggs.

APPENDICES

1) Import required libraries

```
rm(list=ls())
library(TSA)
library(fUnitRoots)
library(lmtest)
library(tseries)
library(FitAR)
library(forecast)
setwd('C:/Users/Wajahath/Desktop/Analytics/Sem 3/Time Series/Assignment 2')
source('C:/Users/Wajahath/Desktop/Analytics/Sem 3/Time Series/Tasks/Module 6/sort.score.R')
source('C:/Users/Wajahath/Desktop/Analytics/Sem 3/Time Series/Tasks/Module 6/residual.analysis.R')
```

2) Import data

```
eggs <- read.csv('eggs.csv',header=FALSE, skip = 1)$V2 #READ SECOND COL
class(eggs)
eggs = ts(as.vector(eggs), start = 1981, end = 1996) #TIME SERIES
class(eggs)
eggs
plot(eggs, type='o', ylab='Egg depositions', col = 'Blue', lwd = 2,     xlab = 'Years', main = 'Time series --
Egg deposition')
```

3) Correlation check

```
#Scatter plot
plot(y=eggs,x=zlag(eggs),ylab='Deposition', xlab='Previous Year Deposition',     col = 'Blue', lwd =
2,main= 'Scatter plot for Correlation')
#Correlation check
y = eggs
x = zlag(eggs)
index = 2:length(x)
cor(y[index],x[index])
```

4) Autocorrelation

```
par(mfrow=c(1,2))
acf(eggs, main='ACF TS')
pacf(eggs, main='PACF TS')
shapiro.test(eggs)
#ADF test for stationarity
adf.test(eggs)
#SERIES IS NON-STATIONARY. Therefore, transformation required
```

5) Transformation

```
par(mfrow=c(1,1))
eggs.trans = BoxCox.ar(eggs)
eggs.trans = BoxCox.ar(eggs, method = "yule-walker")
eggs.trans$ci
lambda = 0.5
eggs.trans = (eggs^lambda-1)/lambda
plot(eggs.trans,type='o',ylab='Yearly changes in the Deposition of Eggs',
   xlab ='Year',col = 'Blue', lwd = 2, main = 'BoxCox transformed Time Series')
qqnorm(eggs.trans)
qqline(eggs.trans, col = 'Blue')
shapiro.test(eggs.trans)
```

6) Differencing

```
diff.eggs.trans = diff(eggs.trans)
par(mfrow=c(1,1))
plot(diff.eggs.trans,type='o',ylab='Egg Depositions', xlab='Year', col = 'Blue', lwd = 2,main='Time
series plot after differencing')
adf.test(diff.eggs.trans)
# diff.eggs.trans1 = diff(eggs.trans, difference = 2)
# adf.test(diff.eggs.trans1)
# diff.eggs.trans2 = diff(eggs.trans, difference = 3)
# adf.test(diff.eggs.trans2)
par(mfrow=c(1,2))
acf(diff.eggs.trans, main='ACF of differenced series')
pacf(diff.eggs.trans,main='PACF of differenced series')
```

7) EACF

```
par(mfrow=c(1,1))
eacf(diff.eggs.trans, ar.max = 3, ma.max = 3)
```

8) BIC table

```
res = armasubsets(y=diff.eggs.trans,nar=3,nma=3,y.name='test',ar.method='ols')
plot(res)
```

9) Model fitting

```
#ARIMA(0,1,1)
mod_011_css = arima(eggs.trans, order = c(0,1,1), method = 'CSS')
coeftest(mod_011_css)
mod_011_ml = arima(eggs.trans, order = c(0,1,1), method = 'ML')
coeftest(mod_011_ml)
```

```
#ARIMA(1,1,0)
mod_110_css = arima(eggs.trans, order = c(1,1,0), method = 'CSS')
coeftest(mod_110_css)
mod_110_ml = arima(eggs.trans, order = c(1,1,0), method = 'ML')
coeftest(mod_110_ml)
#ARIMA(1,1,1)
mod_111_css = arima(eggs.trans, order = c(1,1,1), method = 'CSS')
coeftest(mod_111_css)
mod_111_ml = arima(eggs.trans, order = c(1,1,1), method = 'ML')
coeftest(mod_111_ml)
#ARIMA(2,1,0)
mod_210_css = arima(eggs.trans, order = c(2,1,0), method = 'CSS')
coeftest(mod_210_css)
mod_210_ml = arima(eggs.trans, order = c(2,1,0), method = 'ML')
coeftest(mod_210_ml)
#ARIMA(2,1,1)
mod_211_css = arima(eggs.trans, order = c(2,1,1), method = 'CSS')
coeftest(mod_211_css)
mod_211_ml = arima(eggs.trans, order = c(2,1,1), method = 'ML')
coeftest(mod_211_ml)
#ARIMA(2,1,2)
mod_212_css = arima(eggs.trans, order = c(2,1,2), method = 'CSS')
coeftest(mod_212_css)
mod_212_ml = arima(eggs.trans, order = c(2,1,2), method = 'ML')
coeftest(mod_212_ml)
#ARIMA(3,1,0)
mod_310_css = arima(eggs.trans, order = c(3,1,0), method = 'CSS')
coeftest(mod_310_css)
mod_310_ml = arima(eggs.trans, order = c(3,1,0), method = 'ML')
coeftest(mod_310_ml)
#ARIMA(3,1,1)
mod_311_css = arima(eggs.trans, order = c(3,1,1), method = 'CSS')
coeftest(mod_311_css)
mod_311_ml = arima(eggs.trans, order = c(3,1,1), method = 'ML')
coeftest(mod_311_ml)
#ARIMA(3,1,2)
mod_312_css = arima(eggs.trans, order = c(3,1,2), method = 'CSS')
coeftest(mod_312_css)
mod_312_ml = arima(eggs.trans, order = c(3,1,2), method = 'ML')
coeftest(mod_312_ml)
```

10) AIC and BIC

```
sort.score(AIC(mod_011_ml,mod_110_ml,mod_111_ml, mod_210_ml, mod_211_ml,
mod_212_ml, mod_310_ml,mod_311_ml,mod_312_ml), score = 'aic')

sort.score(BIC(mod_011_ml,mod_110_ml,mod_111_ml, mod_210_ml, mod_211_ml,
mod_212_ml, mod_310_ml,mod_311_ml,mod_312_ml), score = 'bic')
```

11) Overfitting

```
# ARIMA(3,1,2)
mod_312_css = arima(eggs.trans,order=c(3,1,2),method='CSS')
coeftest(mod_312_css)
# ARIMA(2,1,3)
mod_213_css = arima(eggs.trans,order=c(2,1,3),method='CSS')
coeftest(mod_213_css)
```

12) Residual Analysis

```
residual.analysis(mod_212_css)
residual.analysis(mod_213_css)
```

13) Forcasting

```
install.packages('forecast')
library(forecast)
par(mfrow=c(1,1))
fit = Arima(eggs, model = mod_212_css, lambda=0.5)
plot(forecast(fit,h=5), ylab = 'Egg deposition in millions', ylim = c(-2,4), xlim = c(1982,2001),
type = 'l', main = 'Egg deposition forecast using ARIMA(2,1,2)')
```

REFERENCES

Rmit.instructure.com. 2020. *Myapps Portal*. [online] Available at: <https://rmit.instructure.com/courses/67182/files/10725972?module_item_id=2056170&fd_cookie_set=1> [Accessed 10 May 2020]. --- Module 6

Rmit.instructure.com. 2020. *Myapps Portal*. [online] Available at: <https://rmit.instructure.com/courses/67182/files/10725987?module_item_id=2056178&fd_cookie_set=1> [Accessed 10 May 2020]. --- Module 7

Rmit.instructure.com. 2020. *Myapps Portal*. [online] Available at: <https://rmit.instructure.com/courses/67182/files/12082056?module_item_id=2372831> [Accessed 10 May 2020]. --- Soln Task 7

Otexts.com. 2020. *8.6 Estimation And Order Selection | Forecasting: Principles And Practice*. [online] Available at: <https://otexts.com/fpp2/arima-estimation.html> [Accessed 10 May 2020].