Analysis of Egg Depositions of Age-3 Lake Huron Bloaters (Coregonus Hoyi)

MATH1318 Time Series Analysis - Assignment 2

Sarthak Sirari (S3766477)

Introduction

Coregonus Hoyi (a.k.a. Bloater) is a kind of freshwater whitefish, belonging to the family Salmonidae. It could be found in Great Lakes of North America, near the depths of 30 to 198 metres. Its average length is 10 inches, but it could be as big as 14.6 inches and comfortable in the temperatures between 1.5 to 11.4 Celsius.

The population of Coregonus Hoyi is under decline and IUCN Red List has listed it as Vulnerable to global extinction. Its decline is due to predation by the species alewife and sea lamprey, and due to water pollution as well.

However, efforts by the activists to re-introduce Coregonus Hoyi in the Lake Ontario was a success and since then its numbers have increased. The plan is to increase its total population to 500,000 in Lake Ontario.

We will be observing the data of egg depositions of Age-3 Lake Huron Bloaters, performing several fitting operations, running some diagnostics to find the best fit model for the data.

Further, we will be predicting the values future values of egg depositions.

Problem Statement

The purpose of this assignment is to determine the best fit model among the available deterministic trend models (such as linear and quadratic) and the stochastic trends (like ARIMA model). Also, we have to provide the predictions of egg depositions for the next 5 years.

Data

Our data set is available in BloaterLH dataset of FSAdata package and also provided by our course coordinator and contains a total of 16 observations.

The dataset provided in BloaterLH dataset of FSAdata package consists of 3 columns namely year, eggs and age3, whereas the data provided by our course coordinator contains 2 columns, i.e. year and eggs. However, we need to use just the data of eggs column for analysis. The unit of the eggs column is millions.

Load Libraries and User Defined Functions

Load all the necssary libraries and all the necssary user defined functions

```
# Load all necessary libraries
library(TSA)
library(tseries)
library(FSAdata)
library(lmtest)
library(forecast)

# Load the user defined function for sorting AIC and BIC scores saved in external file
source('C:\\Users\\abhis\\OneDrive\\Desktop\\Master of Data Science\\Sem 3\\Time Series Analysis MATH1318\\Utility Functions\\sort.score.R')
# Load the user defined function for residual analysis saved in external
source('C:\\Users\\abhis\\OneDrive\\Desktop\\Master of Data Science\\Sem 3\\Time Series Analysis MATH1318\\Utility Functions\\residual.analysis.R')
```

Load Data Set

```
# Load the provided data set
data(BloaterLH)

# Check the data
head(BloaterLH)
```

	year <int>	eggs <dbl>	age3 <dbl>
1	1981	0.0402	5.143
2	1982	0.0602	154.286
3	1983	0.1205	65.143
4	1984	0.1807	102.857
5	1985	0.7229	102.857
6	1986	0.5321	200.571

6 rows

```
# Check the type of the egg depositions data
class(BloaterLH$eggs)
```

```
## [1] "numeric"
```

Convert Egg Depositions Data to Time Series

Now, we convert the converted egg depositions data to a time series.

```
# Convert the converted egg depositions data to time series data
eggs <- ts(BloaterLH$eggs, start=1981, end=1996, frequency=1)

# Check the type of the converted egg depositions data
class(eggs)
```

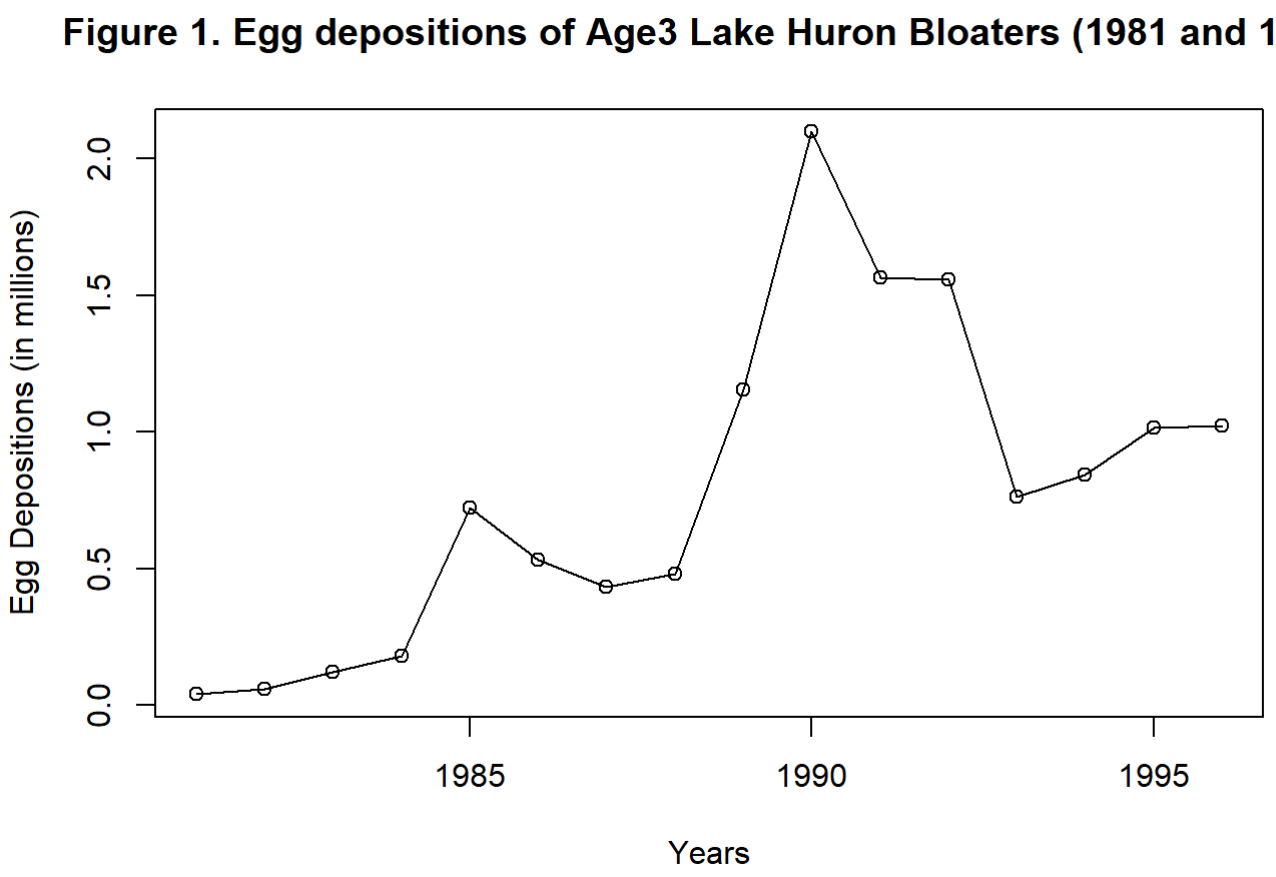
```
## [1] "ts"
```

Data Analysis

Time Series Plot

From the plot in Figure 1, following can be figured out: -
* Trend: There is an upward trend that can be clearly seen, with some with some change in trend around the year 1990.
* Seasonality: There no evidence of seasonality.
* Intervention: No significant intervention points can be noticed. However, there are some points around the years 1985 and 1990, where we can observe a sudden spike, but it is not apparent that these are the intervention points.
* Variance Change: With the two peaks around the years 1985 and 1990, the data seems to have minor change in variance.

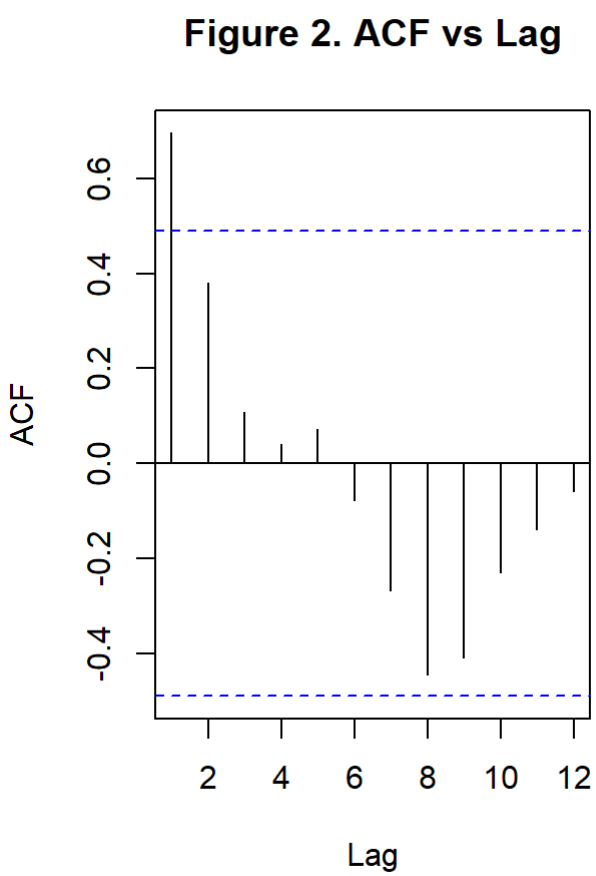
```
# Plot the time series plot
plot(eggs, ylab='Egg Depositions (in millions)', xlab='Years', type='o', main = 'Figure 1. Eg
g depositions of Age3 Lake Huron Bloaters (1981 and 1996)')
```

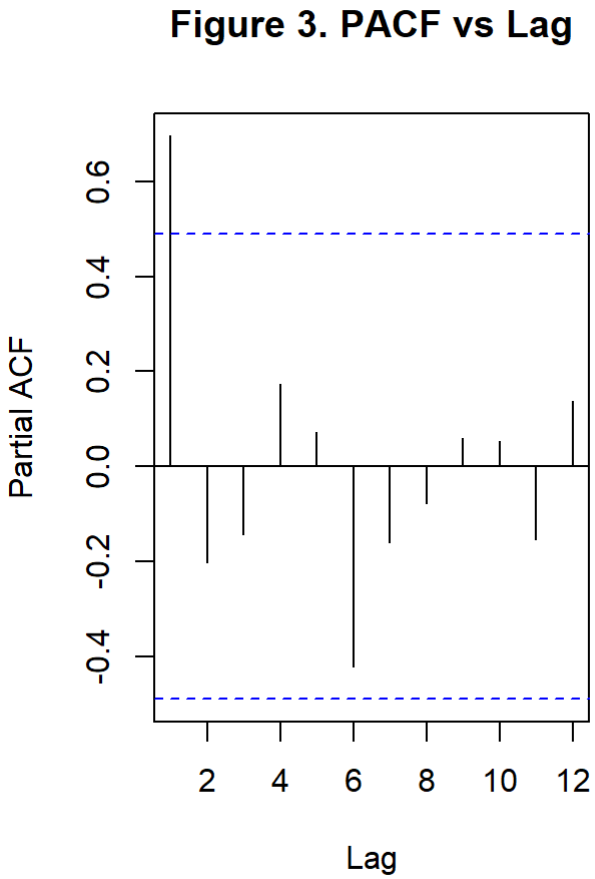


ACF and PACF plot

In the ACF plot in Figure 2 and PACF plot in Figure 3 of our time series data, we can see slowly decaying pattern in ACF and very high first correlation in PACF implies the existence of trend and non-stationarity.

```
# Plot the ACF and PACF plots of time series data
par(mfrow=c(1,2))
acf(eggs, main='Figure 2. ACF vs Lag')
pacf(eggs, main='Figure 3. PACF vs Lag')
```



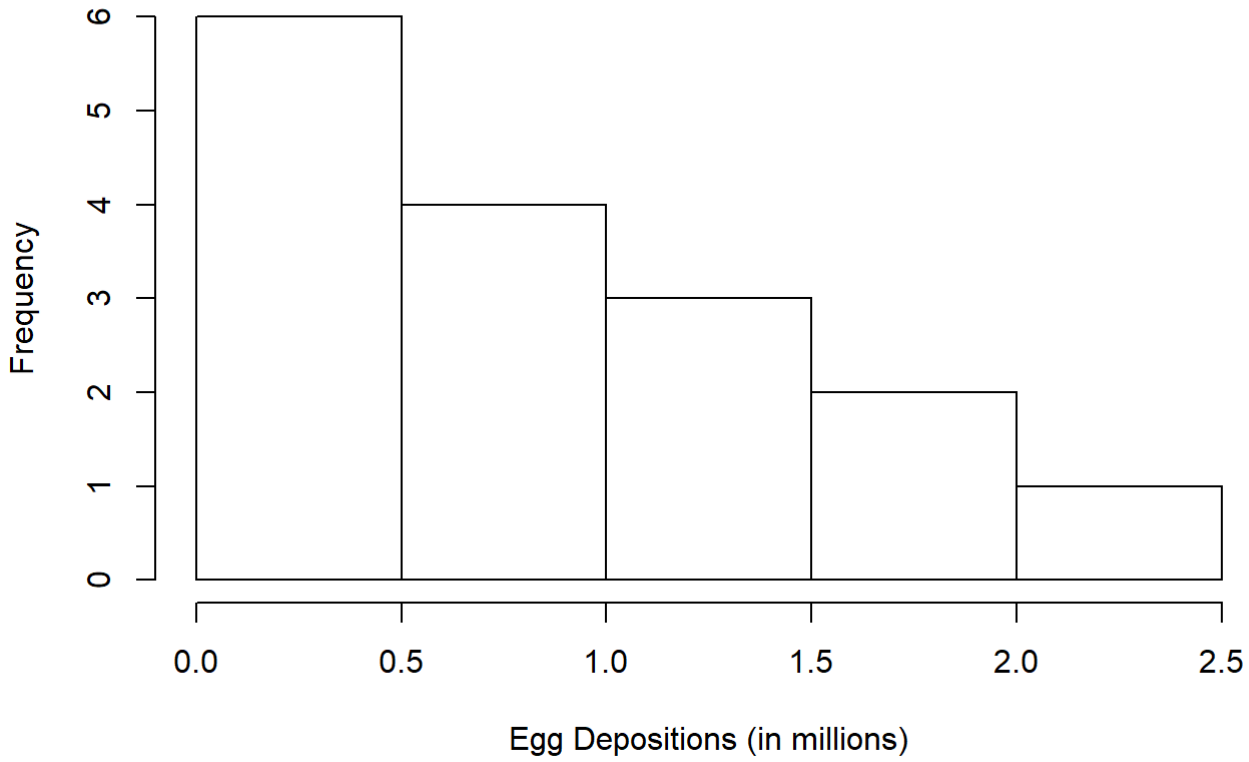


Histogram

The histogram shown in Figure 4 is skewed to the right, which means our time series data is not normally distributed.

```
# Plot the histogram of time series data
hist(eggs,xlab='Egg Depositions (in millions)', main = 'Figure 4. Histogram of the standardiz
ed residuals')
```

Figure 4. Histogram of the standardized residuals

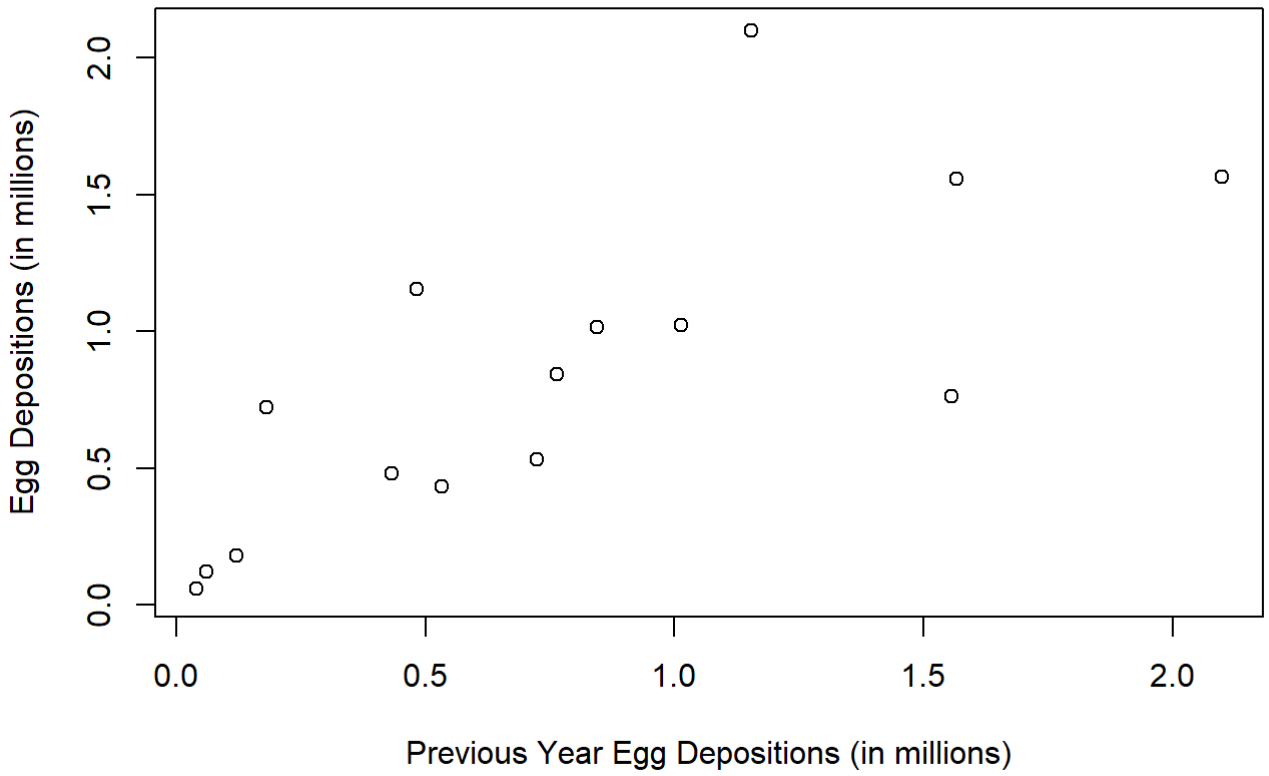


Scatter Plot

In the plot in Figure 5, a slight upward trend between Egg Depositions of consecutive years can be seen.

```
# Plot the scatter plot
plot(y=eggs,x=zlag(eggs),ylab='Egg Depositions (in millions)', xlab='Previous Year Egg Deposi
tions (in millions)' , main = 'Figure 5. Scatter plot of Egg Depositions change in Consecutiv
e years')
```

Figure 5. Scatter plot of Egg Depositions change in Consecutive years



Correlation

A high correlation factor between Egg Depositions of consecutive years of 0.74 can be observed, which supports our observation from PACF plot in Figure 3.

```
# Calculate correlation between the egg deposition change in the consecutive years
y = eggs
x = zlag(eggs)
index = 2:length(x)
cor(y[index],x[index]) # strong correlation 0.87
```

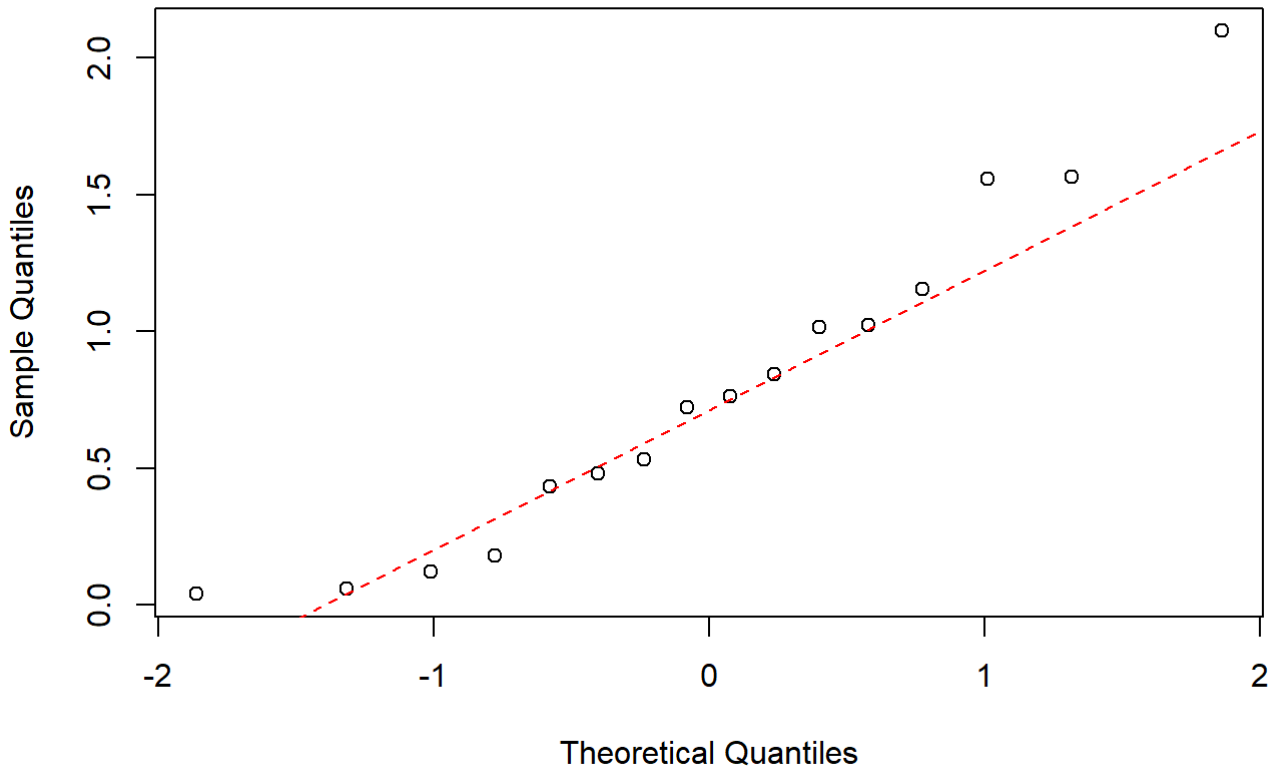
```
## [1] 0.7445657
```

Q-Q plot

In the Q-Q plot in Figure 6, we can observe that the tails of distribution is far from the normality and our time series data does not seems to be white noise.

```
# Plot the Q-Q plot of the standardized residuals of the Egg depositions
qqnorm(eggs, main = 'Figure 6. Normal Q-Q plot of Egg Depositions')
qqline(eggs, col = 2, lwd = 1, lty = 2)
```

Figure 6. Normal Q-Q plot of Egg Depositions



Shapiro-Wilk test

From the summary of Shapiro-Wilk normality test shown below, the p-value turn out to be 0.3744, which is greater than 0.05. So we fail to reject the null hypothesis of normal distribution.

```
# Apply the Shapiro-Wilk test of the Egg Depositions
shapiro.test(eggs)
```

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

Data Analysis Result

From the analysis of data performed above, we can observe that the data is non-stationary and does not following normal distribution. Also, the data shows Autoregressive as well as Moving Average behaviour (ARIMA Model). However, we will still try to fit a Linear and Quadratic model, just to be sure about it.

Data Modelling

In this part, we will attempt to find the best fit model for our time series data. Even though the data shows clear signs of being an ARIMA Model, we will still try to fit the Linear and Quadratic Models.

Linear Model

From the summary of the fitted Linear Model below, following can be observed: -
* The p-value of 0.004642 is less than the significance level of 0.05, hence we can reject the null hypothesis that the model fits our time series data.
* The R^2 value is 0.4074 (i.e. only 40.74% of variance can be explained by the model), which is less than the ideal 0.8, hence the Linear Model is not the best fit model for our data.
Therefore, we move forward to check the Quadratic Model.

```
# Fit Linear Trends Model
model.eggs.ln = lm(eggs~time(eggs))
summary(model.eggs.ln)
```

```
##
## Call:
## lm(formula = eggs ~ time(eggs))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4048 -0.2768 -0.1933  0.2536  1.1857
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -165.98275   49.58836  -3.347  0.00479 **
## time(eggs)     0.08387    0.02494   3.363  0.00464 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4598 on 14 degrees of freedom
## Multiple R-squared:  0.4469, Adjusted R-squared:  0.4074
## F-statistic: 11.31 on 1 and 14 DF,  p-value: 0.004642
```

Quadratic Model

From the summary of the fitted Quadratic Model below, following can be observed: -
* The p-value of 0.00289 is less than the significance level of 0.05, hence we can reject the null hypothesis that the model fits our time series data.
* The R^2 value is 0.5306 (i.e. only 53.06% of variance can be explained by the model), which is less than the ideal 0.8, hence the Quadratic Model is not the best fit model for our data.
Therefore, we move forward to check the ARIMA Model.

```
# Fit Quadratic Trends Model
t = time(eggs)
t2 = t^2
model.eggs.qa = lm(eggs ~ t + t2)
summary(model.eggs.qa)
```

```
##
## Call:
## lm(formula = eggs ~ t + t2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.50896 -0.25523 -0.02701  0.16615  0.96322
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.647e+04  2.141e+04  -2.170   0.0491 *
## t            4.665e+01  2.153e+01   2.166   0.0494 *
## t2          -1.171e-02  5.415e-03  -2.163   0.0498 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4092 on 13 degrees of freedom
## Multiple R-squared:  0.5932, Adjusted R-squared:  0.5306
## F-statistic: 9.479 on 2 and 13 DF,  p-value: 0.00289
```

ARIMA Model

After fitting our time series data in Linear and Quadratic models, we can conclude that the egg depositions data does not show Deterministic trend. Hence, we can conclude that this is a Stochastic trend.

Dickey-Fuller Unit-Root Test (ADF)

We perform this ADF test just be sure about the non-stationarity of our time series data.

From the test, we can see the p-value turn out to be 0.5469, which is greater than 0.05. So we have to we conclude that our time series data is non-stationary.

Hence, we move forward to apply Box-Cox transformation to our data.

```
# Perform Dickey-Fuller Unit-Root Test (ADF) Test on time series data
adf.test(eggs)
```

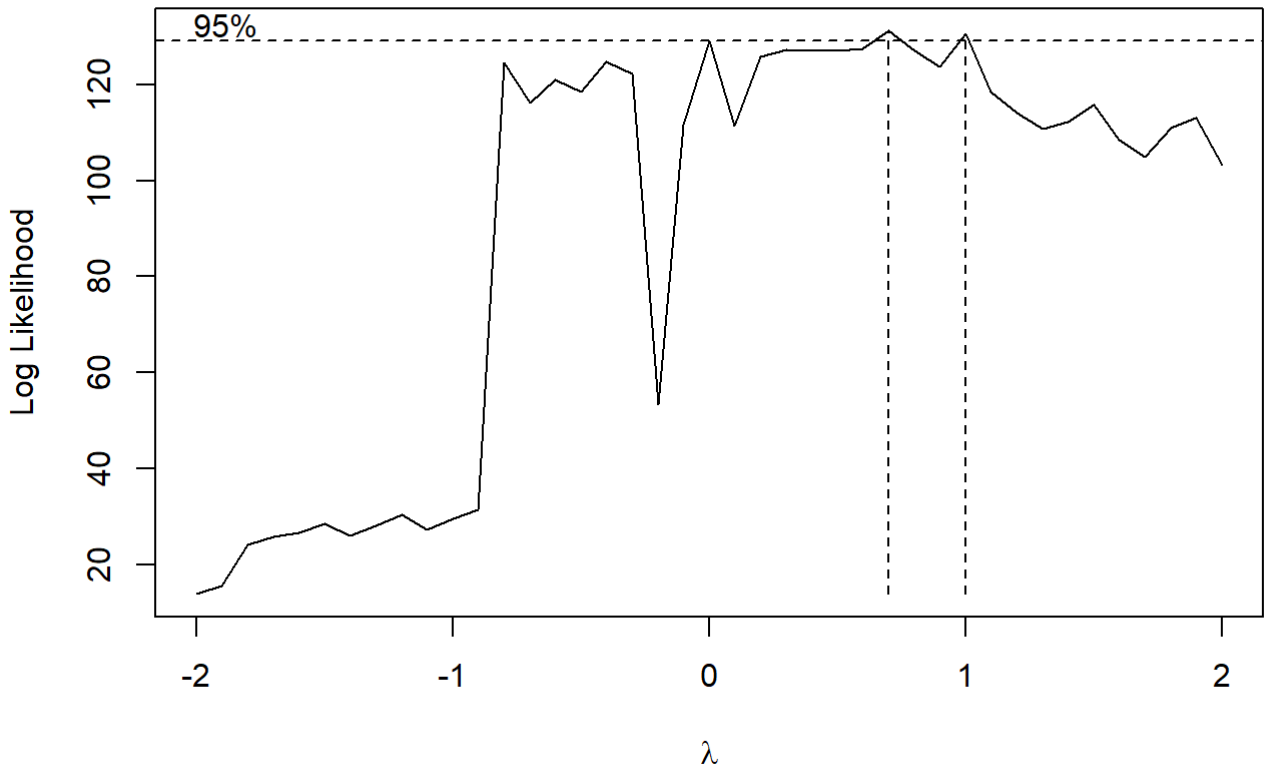
```
##
##  Augmented Dickey-Fuller Test
##
## data:  eggs
## Dickey-Fuller = -2.0669, Lag order = 2, p-value = 0.5469
## alternative hypothesis: stationary
```

Box-Cox transformation (Default)

We perform the Box-Cox Transformation (shown in Figure 7) on our time series data using the Default method, but since it is a short series, this method may not agree on lambda. Hence, we try the Yule Walker method for determining the value of lambda.

```
# Perform Box-Cox transformation using Default method
eggs.bc.transform = BoxCox.ar(eggs)
title(main = 'Figure 7. Log likelihood vs the values of lambda (Default)')
```

Figure 7. Log likelihood vs the values of lambda (Default)

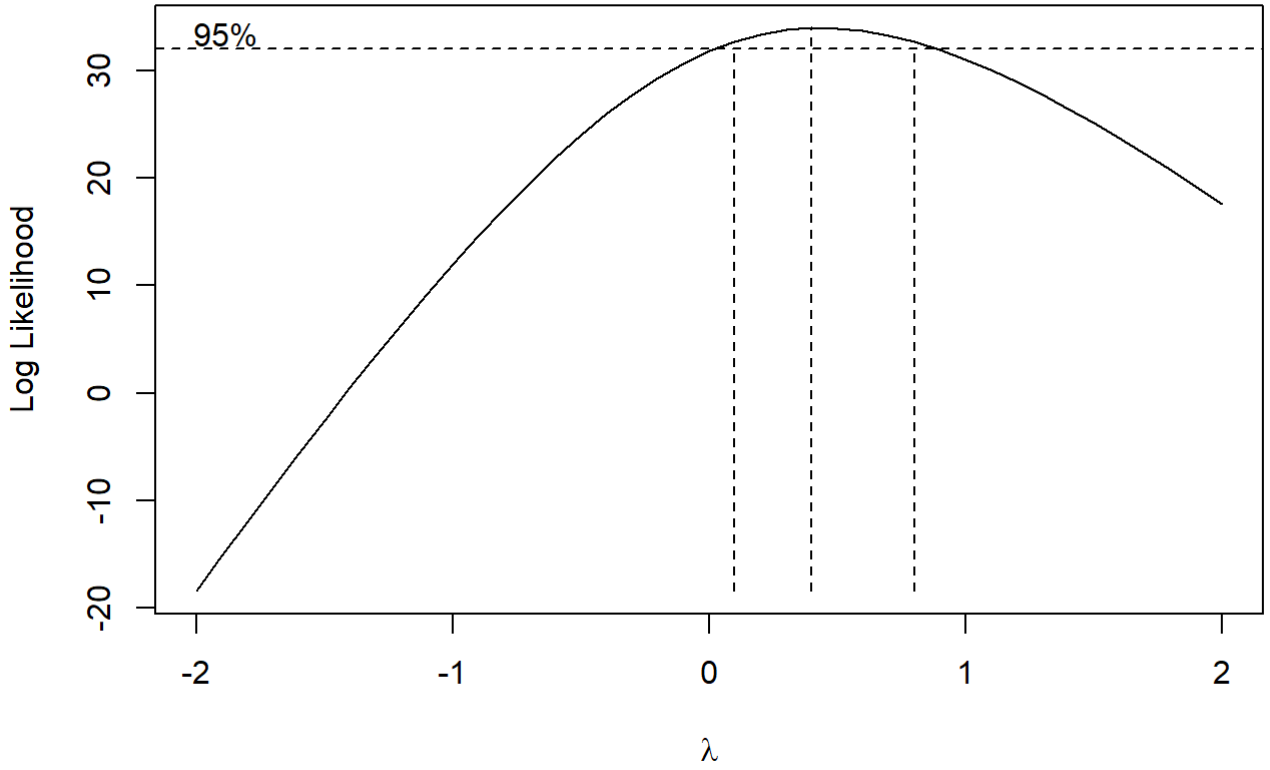


Box-Cox transformation (Yule Walker)

We perform the Box-Cox Transformation (shown in Figure 8) on our time series data using the Yule Walker method and check the confidence interval.

```
# Perform Box-Cox transformation using Yule Walker method
eggs.bc.transform = BoxCox.ar(eggs, method = "yule-walker")
title(main = 'Figure 8. Log likelihood vs the values of lambda (Yule Walker)')
```

Figure 8. Log likelihood vs the values of lambda (Yule Walker)




```
# Check confidence interval
eggs.bc.transform$ci
```

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

Box-Cox transformation of Data

To create the Box-Cox transformed data, we used the mid-point of the confidence interval as lambda value

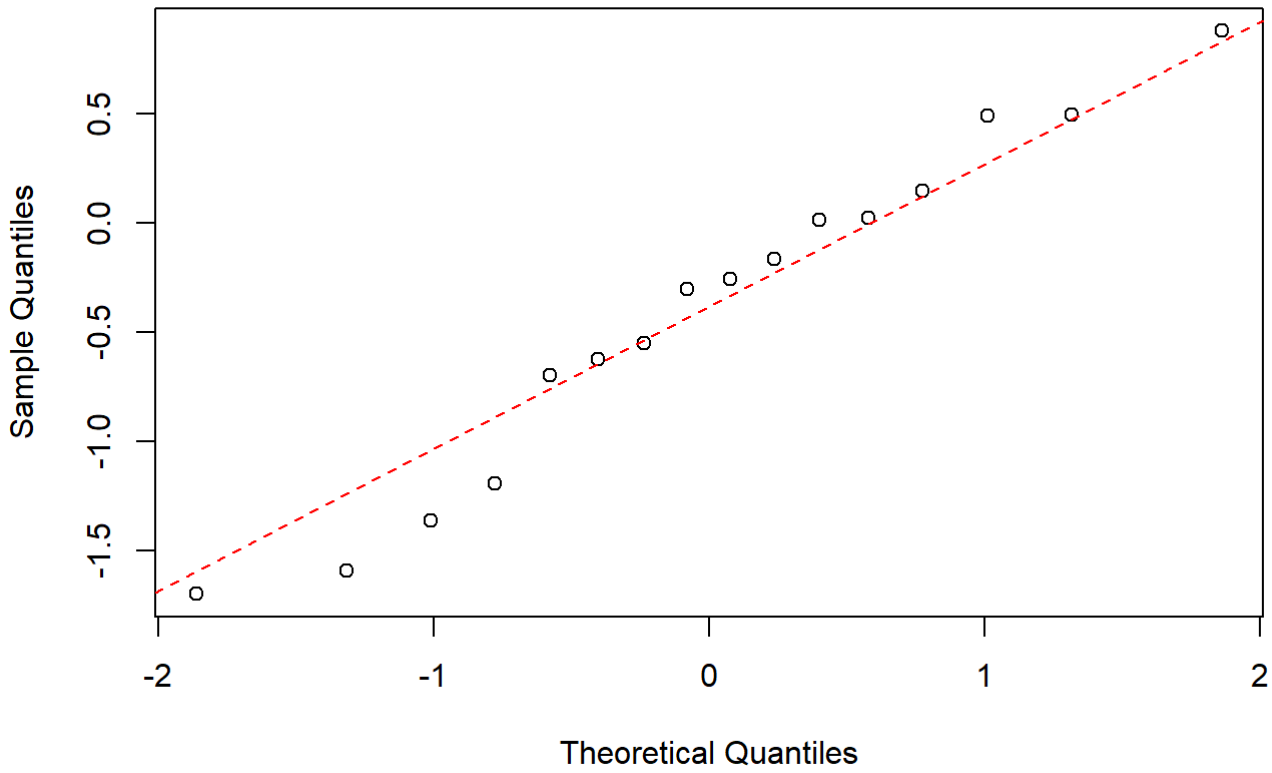
```
# Set lambda value to the mid-point of the confidence interval
lambda = 0.45

# Create Box-Cox transformed data
eggs.bc = (eggs^lambda-1)/lambda
```

Q-Q plot

In the Q-Q plot in Figure 9, we can observe that the tails of distribution are now nearer to the normal distribution line, hence, increasing the normality.

```
# Plot the Q-Q plot of the standardized residuals of the Box-Cox transformed data
qqnorm(eggs.bc, main = 'Figure 9. Normal Q-Q plot of Box-Cox transformed data')
qqline(eggs.bc, col = 2, lwd = 1, lty = 2)
```

Figure 9. Normal Q-Q plot of Box-Cox transformed data



Shapiro-Wilk test

From the summary of Shapiro-Wilk normality test shown below, the p-value has improved from 0.3744 to 0.7107, which is greater than 0.05. So we fail to reject the null hypothesis of normal distribution.

```
# Apply the Shapiro-Wilk test of the Box-Cox transformed data
shapiro.test(eggs.bc)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  eggs.bc
## W = 0.96269, p-value = 0.7107
```

Dickey-Fuller Unit-Root Test (ADF)

From the test, we can see the p-value turn out to be 0.6955, which is greater than 0.05. So we have to we conclude that our time series data is non-stationary.

Hence, we move forward to apply differencing of Box-Cox transformed data.

```
# Perform Dickey-Fuller Unit-Root Test (ADF) Test on time series data
adf.test(eggs.bc)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  eggs.bc
## Dickey-Fuller = -1.6769, Lag order = 2, p-value = 0.6955
## alternative hypothesis: stationary
```

First differencing of Box-Cox transformed data

We perform the first differencing below to Box-Cox transformed data

```
# Perform first differencing of Time Series data
eggs.bc.diff = diff(eggs.bc)
```

Plot the first differencing of Box-Cox transformed data

In the time series plot in Figure 10, we can see that the after applying first differencing to Box-Cox transformed data, the trend can still be seen.

```
# Plot the first differencing of Time Series data
plot(eggs.bc.diff, type='o', ylab='First differencing of Box-Cox transformed data', xlab='Yea
rs', main='Figure 10. Time Series of first differencing of Box-Cox transformed data')
```

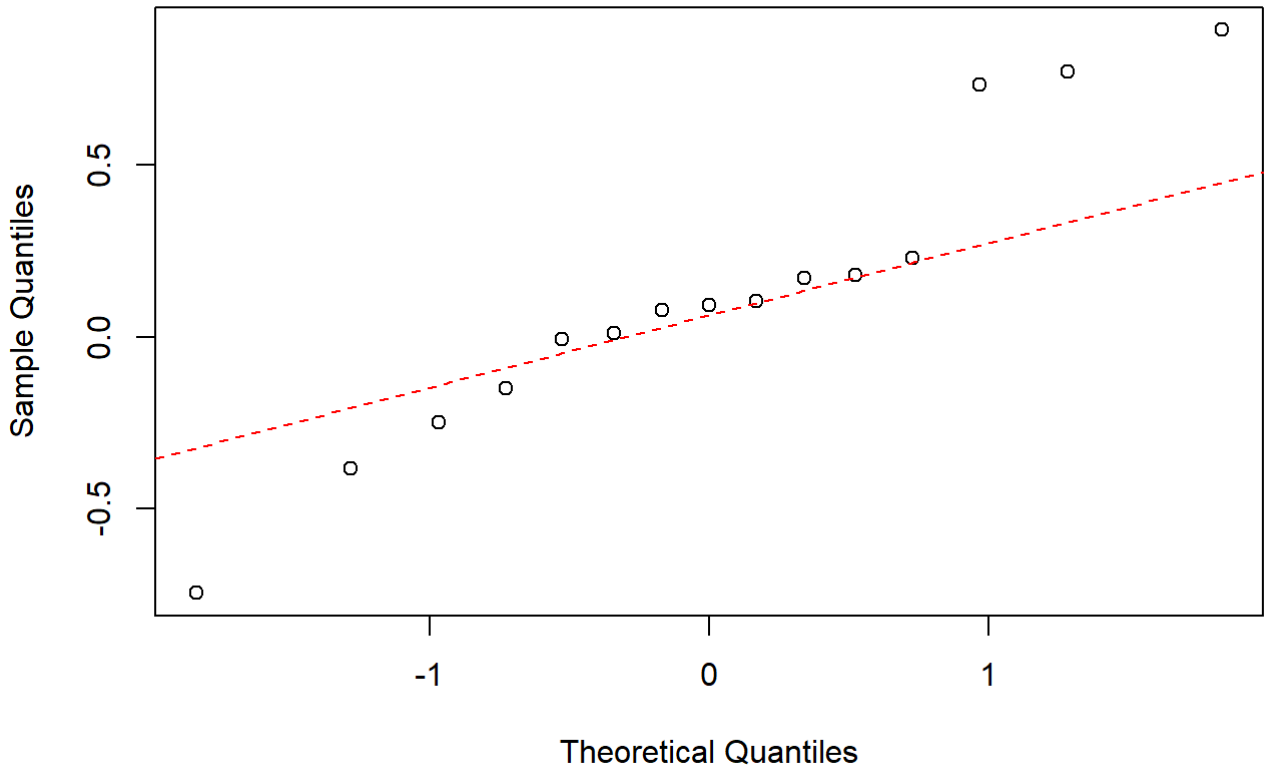
Figure 10. Time Series of first differencing of Box-Cox transformed dat

Q-Q plot

In the Q-Q plot in Figure 11, we can observe that the tails of distribution are a bit away from the normal distribution line around tails.

```
# Plot the Q-Q plot of the standardized residuals of the Box-Cox transformed data
qqnorm(eggs.bc.diff, main = 'Figure 11. Normal Q-Q plot of Box-Cox transformed data')
qqline(eggs.bc.diff, col = 2, lwd = 1, lty = 2)
```

Figure 11. Normal Q-Q plot of Box-Cox transformed data



Shapiro-Wilk test

From the summary of Shapiro-Wilk normality test shown below, the p-value has decreased from 0.7107 to 0.4086, but still it is greater than 0.05. So we fail to reject the null hypothesis of normal distribution.

```
# Apply the Shapiro-Wilk test of the Box-Cox transformed data
shapiro.test(eggs.bc.diff)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  eggs.bc.diff
## W = 0.94203, p-value = 0.4086
```

Dickey-Fuller Unit-Root Test (ADF)

From the test, we can see the p-value turn out to be 0.0443, which is less than 0.05, rejecting the null hypothesis of non-stationarity. So we have to we conclude that our time series data after first differencing of Box-Cox transformed data is stationary.

Therefore, we can move forward to determine the order of the best fit ARIMA model.

```
# Perform Dickey-Fuller Unit-Root Test (ADF) Test on time series data
adf.test(eggs.bc.diff)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  eggs.bc.diff
## Dickey-Fuller = -3.6798, Lag order = 2, p-value = 0.0443
## alternative hypothesis: stationary
```

Data Modelling Result

After performing the data modelling, we conclude that our data gives best results for model fitting, when we do the first differencing of the Box-Cox transformed data.

Determine Best Fit ARIMA Model

ADF and PADF plot

In the ACF plot in Figure 12 and PACF plot in Figure 13 of our first differencing of Box-Cox transformed data, we can observe that both turn out to be white noise as there are no significant lags in both the plots. So we could not find any values of p and q.

Hence, we move forward to create EACF table.

```
# Plot the ADF and PADF plots for first differencing of Box-Cox transformed data
par(mfrow=c(1,2))
acf(eggs.bc.diff, ci.type = 'ma', main='Figure 12. ACF vs Lag')
pacf(eggs.bc.diff, main='Figure 13. ACF vs Lag')
```

Figure 12. ACF vs Lag

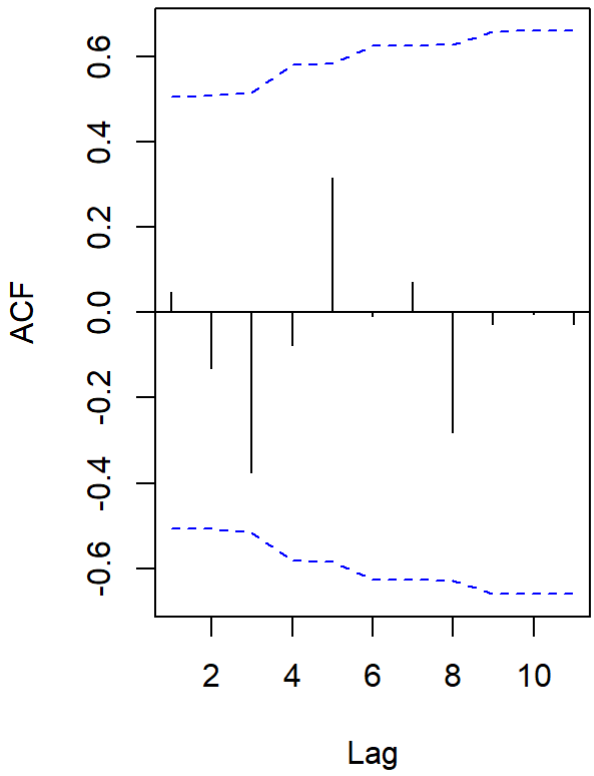
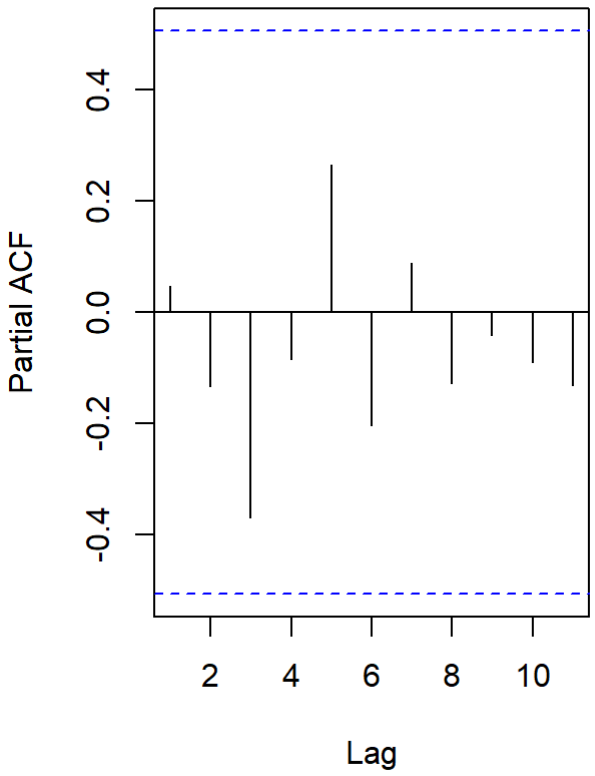


Figure 13. ACF vs Lag



Extended Autocorrelation Function (EACF)

In the EACF Table below, we can take p = 0 and q = 0 as our chosen vertex and include ARIMA(0,1,1), ARIMA(1,1,0) and ARIMA(1,1,1) models into the set of possible AIRMA models.

Now we moved forward to create Bayesian Information Criterion

```
# Create the Extended Autocorrelation Function (EACF) table
eacf(eggs.bc.diff, ar.max = 3, ma.max = 3)
```

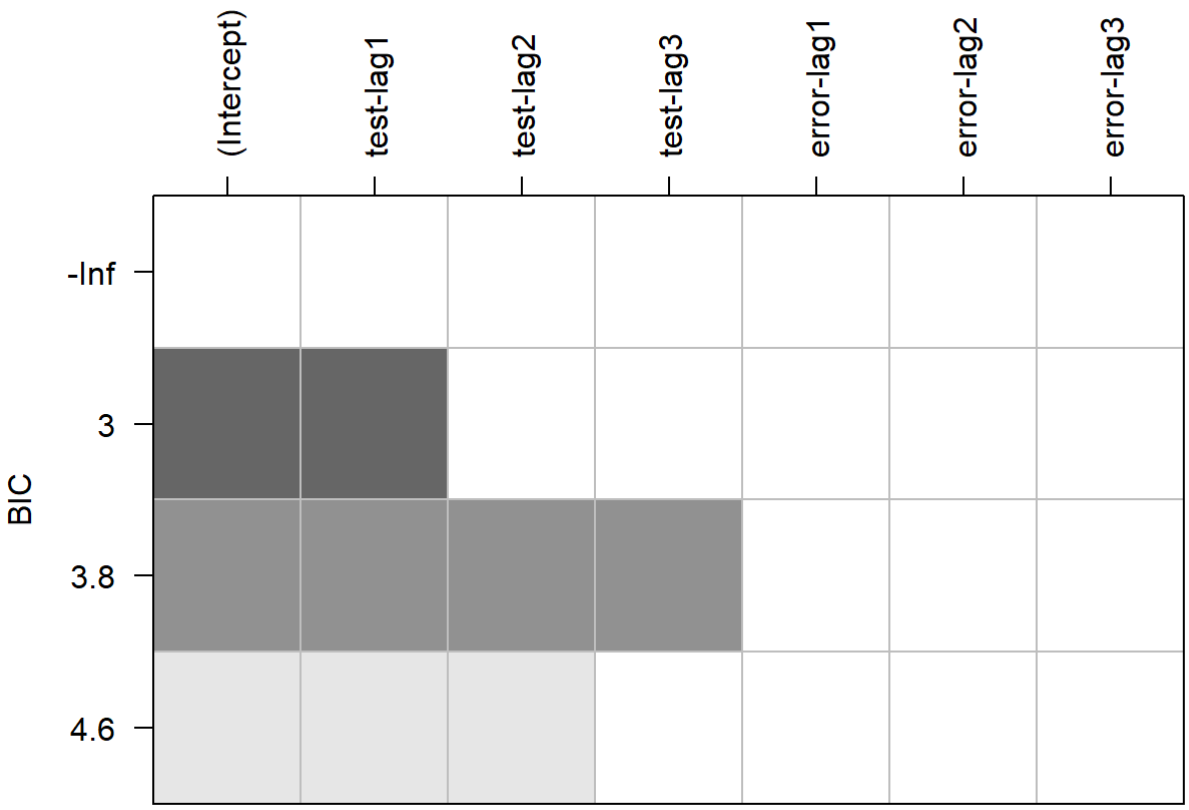
```
## AR/MA
##   0 1 2 3
## 0 o o o o
## 1 o o o o
## 2 o o o o
## 3 o o o o
```

Bayesian Information Criterion (BIC)

In the BIC table in Figure 14, we check the shaded columns and conclude that we can include ARIMA(1,1,0), ARIMA(2,1,0) and ARIMA(3,1,0) models in the set of our candidate models.

```
# Plot the Bayesian Information Criterion table
bic = armasubsets(y = eggs.bc.diff, nar = 3, nma = 3, y.name = 'test', ar.method = 'ols')
plot(bic)
title(main = 'Figure 14. Bayesian Information Criterion table', line = 6)
```

Figure 14. Bayesian Information Criterion table



Best Fit ARIMA Model Result

So to conclude we have the final set of candidate models as ARIMA(0,1,1), ARIMA(1,1,0), ARIMA(1,1,1), ARIMA(2,1,0) and ARIMA(3,1,0).

Parameter Estimation

Here we will perform the Parameter Estimation for each of the final selected ARIMA models.

ARIMA(0,1,1)

The p-value of MA(1) component is less than 0.05 in CSS method, which means it is highly significant, whereas, the p-value of MA(1) component is greater than 0.05 in ML method, which means it is not significant.

```
# Perform significance test using conditional sum of squares method
model_011_css = arima(eggs.bc.diff,order=c(0,1,1),method='CSS')
coeftest(model_011_css)
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error z value  Pr(>|z|)
## ma1 -1.093338   0.056532  -19.34 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Perform significance test using maximum likelihood estimation method
model_011_ml = arima(eggs.bc.diff,order=c(0,1,1),method='ML')
coeftest(model_011_ml)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value Pr(>|z|)
## ma1 -0.99994    0.66766 -1.4977   0.1342
```

ARIMA(1,1,0)

The p-value of AR(1) component is greater than 0.05 in CSS method, which means it is not significant, and the p-value of AR(1) component is greater than 0.05 in ML method, which means it is also not significant.

```
# Perform significance test using conditional sum of squares method
model_110_css = arima(eggs.bc.diff,order=c(1,1,0),method='CSS')
coeftest(model_110_css)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value Pr(>|z|)
## ar1 -0.40609    0.24460 -1.6602  0.09687 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Perform significance test using maximum likelihood estimation method
model_110_ml = arima(eggs.bc.diff,order=c(1,1,0),method='ML')
coeftest(model_110_ml)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value Pr(>|z|)
## ar1 -0.38056    0.23493 -1.6199   0.1053
```

ARIMA(1,1,1)

The p-value of AR(1) component is greater than 0.05 in CSS method, which means it is not significant, and the p-value of AR(1) component is greater than 0.05 in ML method, which means it is also not significant.

The p-value of MA(1) component is less than 0.05 in CSS method, which means it is highly significant, and the p-value of MA(1) component is also less than 0.05 in ML method, which means it is highly significant.

```
# Perform significance test using conditional sum of squares method
model_111_css = arima(eggs.bc.diff,order=c(1,1,1),method='CSS')
coeftest(model_111_css)
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error z value  Pr(>|z|)
## ar1  0.030947   0.290460  0.1065    0.9152
## ma1 -1.021947   0.201926 -5.0610 4.171e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Perform significance test using maximum likelihood estimation method
model_111_ml = arima(eggs.bc.diff,order=c(1,1,1),method='ML')
coeftest(model_111_ml)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value Pr(>|z|)
## ar1  0.11474    0.26992  0.4251 0.670764
## ma1 -1.00000    0.33205 -3.0116 0.002599 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ARIMA(2,1,0)

The p-value of AR(1) component is greater than 0.05 in CSS method, which means it is not significant, and the p-value of AR(1) component is greater than 0.05 in ML method, which means it is also not significant.

The p-value of AR(2) component is greater than 0.05 in CSS method, which means it is not significant, and the p-value of AR(2) component is greater than 0.05 in ML method, which means it is also not significant.

```
# Perform significance test using conditional sum of squares method
model_210_css = arima(eggs.bc.diff,order=c(2,1,0),method='CSS')
coeftest(model_210_css)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value Pr(>|z|)
## ar1 -0.46896    0.26455 -1.7727  0.07629 .
## ar2 -0.15595    0.26434 -0.5899  0.55523
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Perform significance test using maximum likelihood estimation method
model_210_ml = arima(eggs.bc.diff,order=c(2,1,0),method='ML')
coeftest(model_210_ml)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value Pr(>|z|)
## ar1 -0.43885    0.25669 -1.7096  0.08734 .
## ar2 -0.13640    0.24779 -0.5505  0.58200
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ARIMA(3,1,0)

The p-value of AR(1) component is less than 0.05 in CSS method, which means it is significant, and the p-value of AR(1) component is also less than 0.05 in ML method, which means it is also significant.

The p-value of AR(2) component is greater than 0.05 in CSS method, which means it is not significant, and the p-value of AR(2) component is greater than 0.05 in ML method, which means it is also not significant.

The p-value of AR(3) component is less than 0.05 in CSS method, which means it is significant, and the p-value of AR(3) component is also less than 0.05 in ML method, which means it is also significant.

```
# Perform significance test using conditional sum of squares method
model_310_css = arima(eggs.bc.diff,order=c(3,1,0),method='CSS')
coeftest(model_310_css)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value Pr(>|z|)
## ar1 -0.53415    0.21650 -2.4672  0.01362 *
## ar2 -0.41735    0.24223 -1.7229  0.08490 .
## ar3 -0.51729    0.23907 -2.1637  0.03049 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Perform significance test using maximum likelihood estimation method
model_310_ml = arima(eggs.bc.diff,order=c(3,1,0),method='ML')
coeftest(model_310_ml)
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value Pr(>|z|)
## ar1 -0.51836    0.22606 -2.2931  0.02184 *
## ar2 -0.36849    0.24360 -1.5127  0.13036
## ar3 -0.47004    0.23356 -2.0125  0.04417 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Sort Score w.r.t. AIC and BIC values

Here we sort the AIC and BIC scores of the chosen ARIMA models using maximum likelihood estimation method.

As per the AIC and BIC tables shown below, we can observe that ARIMA(0,1,1) has smallest AIC and BIC values both.

```
sort.score(AIC(model_011_ml,model_110_ml,model_111_ml,model_210_ml,model_310_ml), score = "aic")
```

	df <dbl>	AIC <dbl>
model_011_ml	2	23.12446
model_111_ml	3	24.94140
model_110_ml	2	27.05798
model_310_ml	4	27.58888
model_210_ml	3	28.75996

5 rows

```
sort.score(BIC(model_011_ml,model_110_ml,model_111_ml,model_210_ml,model_310_ml), score = "bic" )
```

	df <dbl>	BIC <dbl>
model_011_ml	2	24.40258
model_111_ml	3	26.85857
model_110_ml	2	28.33609
model_310_ml	4	30.14511
model_210_ml	3	30.67714

5 rows

Parameter Estimation Result

From the tables, we conclude that ARIMA(0,1,1) is the best fit model.

Residual Analysis

Now we perform the Residual Analysis of our best fit ARIMA(0,1,1) model and following are the observations from the same: -

* Time Series Plot in Figure 14 shows no trends and also there is no variance that can be observed, which mean our chosen model could be the best fit.
* Histogram in Figure 15 appears to be normally distributed, which means our chosen model could fit the normal distribution, shows little to no skewness.
* The ACF of standardised residuals in Figure 16 appears to be white noise, which means our chosen model fits the data.
* The PACF of standardised residuals in Figure 17 also appears to be white noise, which means our chosen model fits the data.
* The Q-Q Plot shown in Figure 18 depicts that most of the points are near to the normal distribution line, which

means our chosen model could fit the normal distribution.
* The Ljung-Box Test in Figure 19 shows all the points at each lag are above the 5% dashed line, this means we fail to reject the null hypothesis that the error terms are uncorrelated.
* From the summary of Shapiro-Wilk normality test, the p-value turn out to be 0.653, which is greater than 0.05. So we fail to reject the null hypothesis of normal distribution.

```
# Call the function residual.analysis()
residual.analysis(model = model_011_ml)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res.model
## W = 0.95773, p-value = 0.653
```

Figure 14. Time series plot of standardised residuals

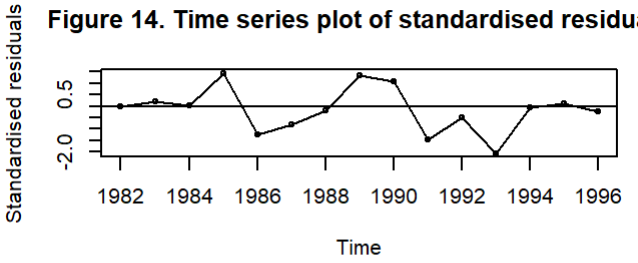


Figure 15. Histogram of standardised residuals

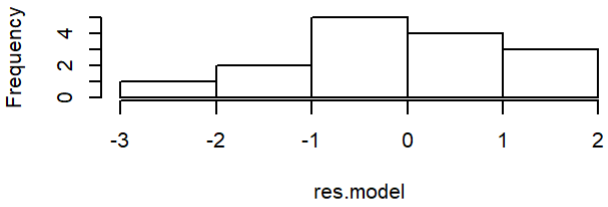


Figure 16. ACF of standardised residuals

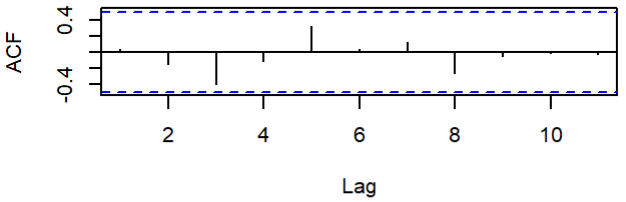


Figure 17. PACF of standardised residuals

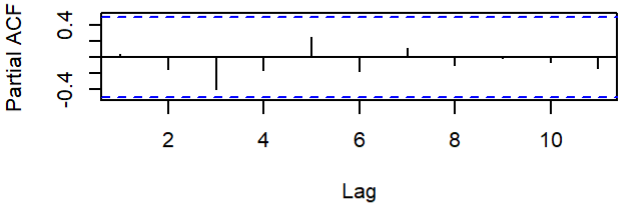


Figure 18. QQ plot of standardised residuals

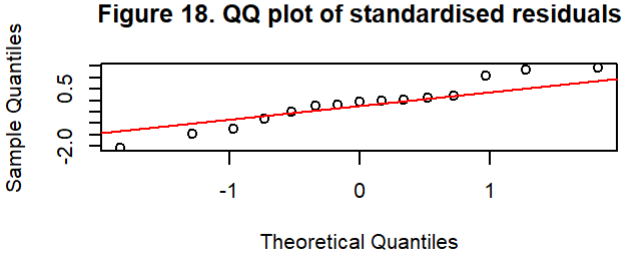
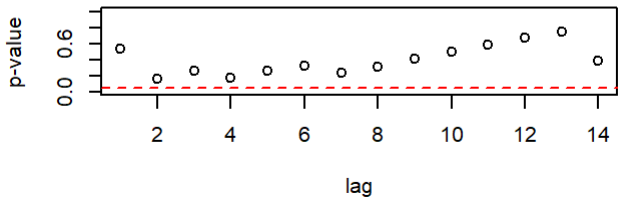


Figure 19.Ljung-Box Test .



Residual Analysis Result

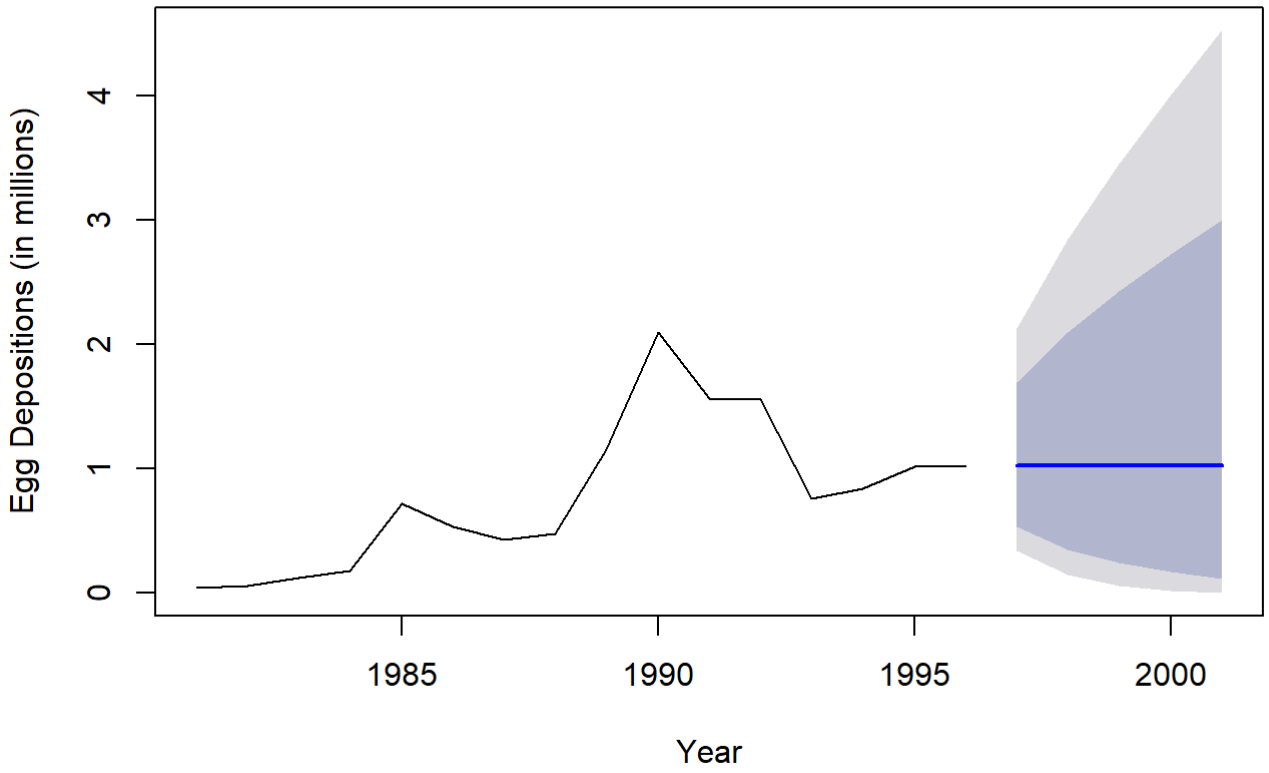
To conclude all the observations from the Residual Analysis performed above, we can confidently say that our chosen model ARIMA(0,1,1) is the best fit model for the data of egg depositions of age-3 Lake Huron Bloaters (1981 and 1996).

Forecasting

Now, finally we perform predicting of the egg depositions (in millions) of age-3 Lake Huron Bloaters for the next 5 year, which can be seen in Figure 20.

```
# Plot the time series plot of the egg deposition with the forecast for next 5 years
fitted_model = Arima(eggs, lambda = 0.45, c(0,1,1))
plot(forecast(fitted_model, h = 5), xlab = 'Year', ylab = 'Egg Depositions (in millions)', ma
in = 'Figure 20. Egg depositions of Age3 Lake Huron Bloaters with Forecast')
```

Figure 20. Egg depositions of Age3 Lake Huron Bloaters with Forecast



Conclusion

- We analysed the data egg depositions of Lake Huron Bloaters between the years 1981 and 1996 and found the data to exhibit Autoregressive as well as Moving Average behaviour (ARIMA Model), yet we try to fit a Linear and Quadratic models, just to be sure.
- We performed the data modelling and concluded that our data gives best results for model fitting, when we do the first differencing of the Box-Cox transformed data.
- Further, we have proposed a set of possible ARIMA models, using each and every model specification tool such as ACF, PACF, EACF, BIC table and other tests as **ARIMA(0,1,1), ARIMA(1,1,0), ARIMA(1,1,1), ARIMA(2,1,0) and ARIMA(3,1,0)**.
- Then we performed Parameter Estimation test, which conclude that **ARIMA(0,1,1)** is the best fit model.
- Going forward, we performed Residual Analysis and conclude that our chosen model **ARIMA(0,1,1)** is the best fit model.
- Finally, we performed the prediction the egg depositions (in millions) of age-3 Lake Huron Bloaters for the next 5 year, by using the best fit **ARIMA(0,1,1)** model.

References

- MATH1318 Time Series Analysis notes prepared by Dr. Haydar Demirhan.
- Time Series Analysis with application in R by Cryer and Chan.