Statistical Analysis of Ozone Layer Thickness

Anurag Dinesh Karmarkar

Data Description

The ozone layer is a region in the Earth's stratosphere which protects the Earth from the harmful ultraviolet rays emitted by the sun. In the recent years, it has been observed that the thickness of the ozone layer is rapidly decreasing. It is of utmost importance that we prevent the ozone layer from getting thinner. The ozone layer thickness is measured in Dobson Units. The dataset contains the measure of the ozone thickness from 1927 to 2016.

Problem Statement

There are 2 tasks in this problem statement. The first task is to analyse the provided data using various modelling techniques such as linear, quadratic, cyclic, seasonal and cosine and determine the best model to fit the data. We will then use the data to predict the dobson unit values for the next five years i.e from 2017 to 2021. The second task is to propose a set of ARIMA(p,d,q) models using various model specification techniques such as ACF-PACF, EACF, BIC.

Importing the libraries required for the task

```
library(TSA)
```

```
##
## Attaching package: 'TSA'
```

```
## The following objects are masked from 'package:stats':
##
##     acf, arima
```

```
## The following object is masked from 'package:utils':
##
##     tar
```

```
library(readr)
```

```
##
## Attaching package: 'readr'
```

```
## The following object is masked from 'package:TSA':
##
##     spec
```

```
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

Importing the data

```
ozonethickness_data <- read_csv("C:/Users/smart/OneDrive/Documents/Master of Data Science (RMI
T)/Semester 3/Time Series Analysis/Assignments/Assignment 1/data1.csv", col_names = FALSE)
```

```
## Parsed with column specification:
## cols(
##   X1 = col_double()
## )
```

```
head(ozonethickness_data)
```

X1 <dbl>
1.3511844
0.7605324
-1.2685573
-1.4636872
-0.9792030
1.5085675

6 rows

Analysis of Time Series Data (Solution of Task 1)

```
#Converting data to a time series object
rownames(ozonethickness_data) <- seq(from=1927, to=2016)
```

```
## Warning: Setting row names on a tibble is deprecated.
```

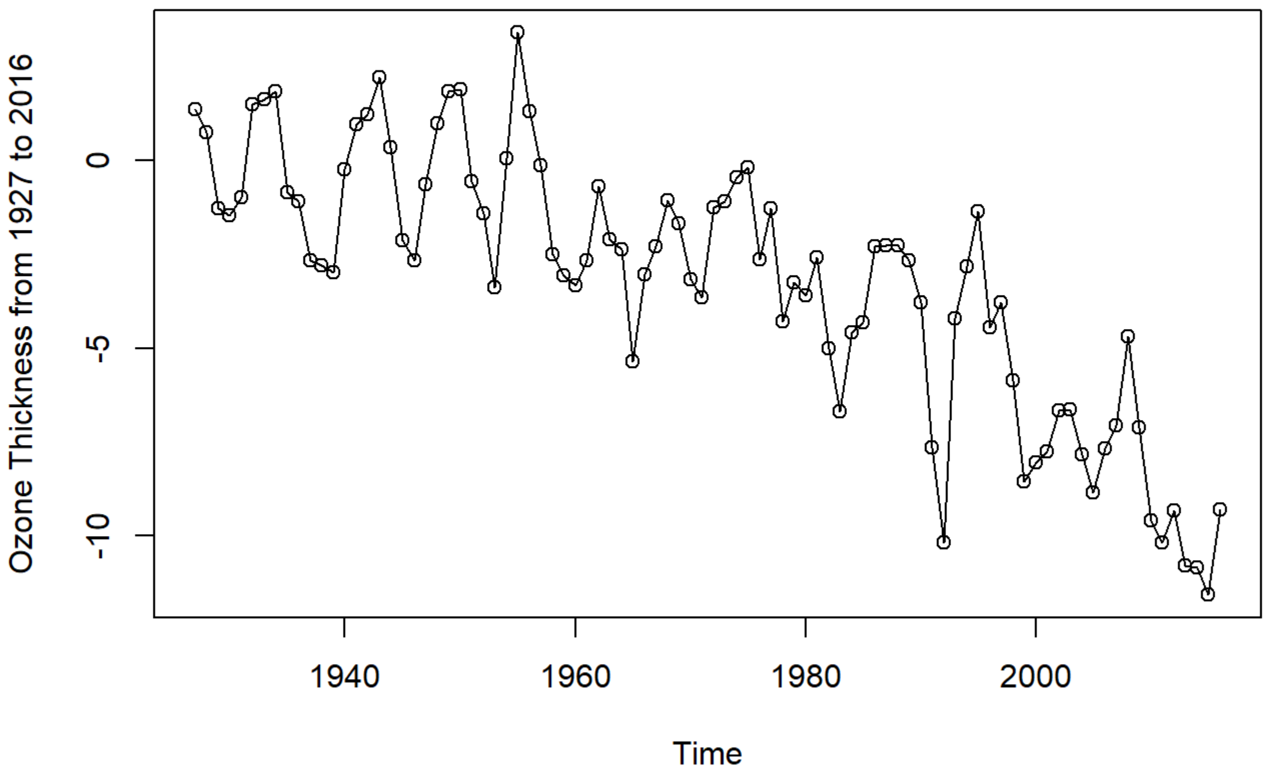
```
class(ozonethickness_data)
```

```
## [1] "spec_tbl_df" "tbl_df"      "tbl"         "data.frame"
```

```
ozonethickness <- ts(as.vector(ozonethickness_data), start=1927, end=2016)
head(ozonethickness)
```

```
## [1]  1.3511844  0.7605324 -1.2685573 -1.4636872 -0.9792030  1.5085675
```

```
#Plotting the time series data
plot(ozonethickness, type='o', ylab='Ozone Thickness from 1927 to 2016')
```



We have converted the data into a time series object and plotted the time series. We can observe that there is a negative trend in the series. The ozone layer has thinned drastically from 1980. The change from 1940 to 1960 is not drastic but gradual. The series appears to be non-cyclic, non-seasonal and non-stationary. However, we will check for these properties in the analysis.

Creating a function for Residual Analysis

We have created a function for the residual analysis which includes plotting the fitted model, histogram, QQ plot, autocorrelation, partial autocorrelation and performing the Shapiro-Wilkins Test. We will use this function to analyze all the models and find the best fit for forcasting the values for the next 5 years.

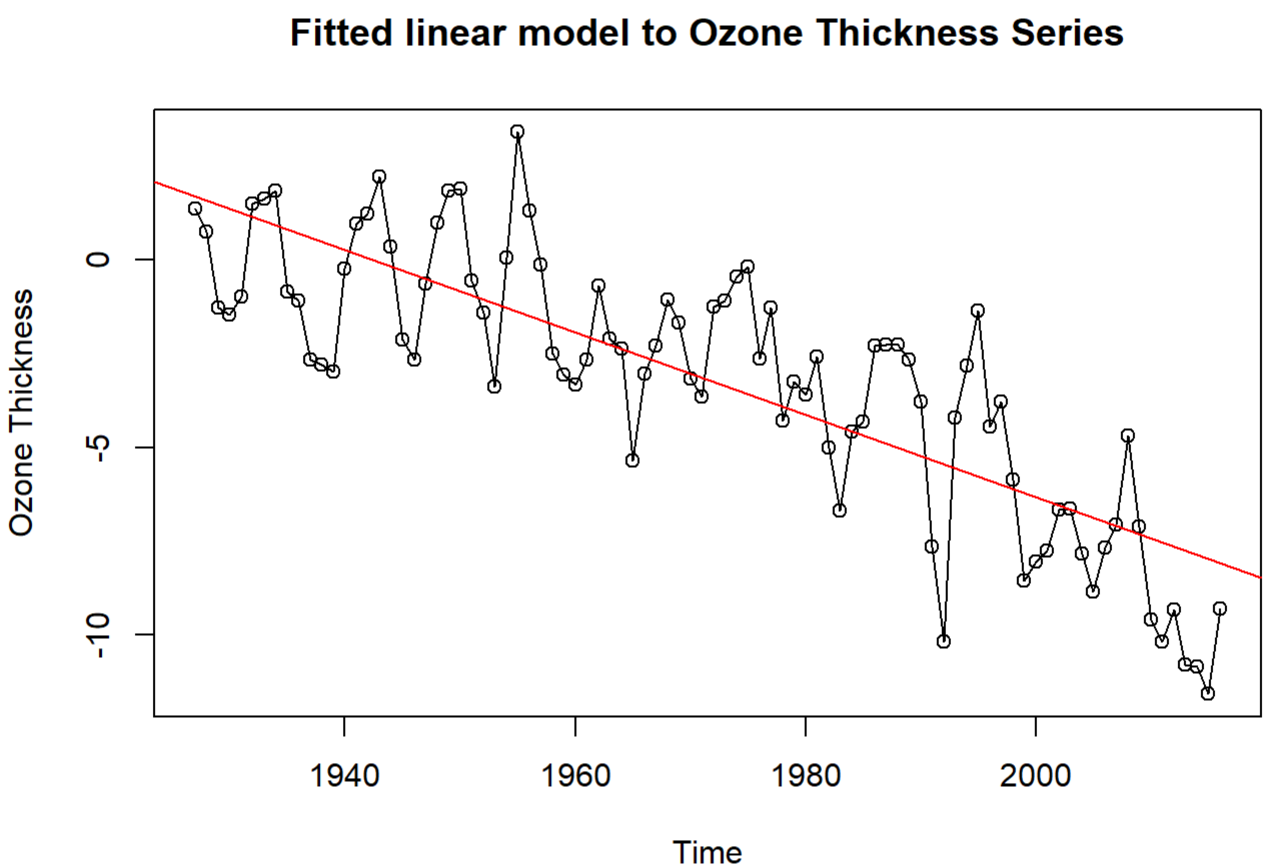
```
residual_analysis <- function(model_used, timeseries) {
  res_model_used = rstudent(model_used)
  pt = plot(y = res_model_used,
            x = as.vector(time(timeseries)),
            xlab = 'Time',
            ylab='Standardized Residuals',
            type='l',
            main = "Standardised residuals from the model")
  abline(h=0)
  his = hist(res_model_used,xlab='Standardized Residuals', main = "Histogram of standardised res
iduals")
  qqn = qqnorm(y=res_model_used, main = "QQ plot of standardised residuals")
  qql = qqline(y=res_model_used, col = 2, lwd = 1, lty = 2)
  sha = shapiro.test(res_model_used)
  autocorr = acf(res_model_used, main = "ACF of standardized residuals")
  par_acf = pacf(res_model_used, main = "PACF of standardized residuals")
  return(his)
  return(qqn)
  return(qql)
  return(sha)
  return(autocorr)
  return(par_acf)
}
```

Linear Model

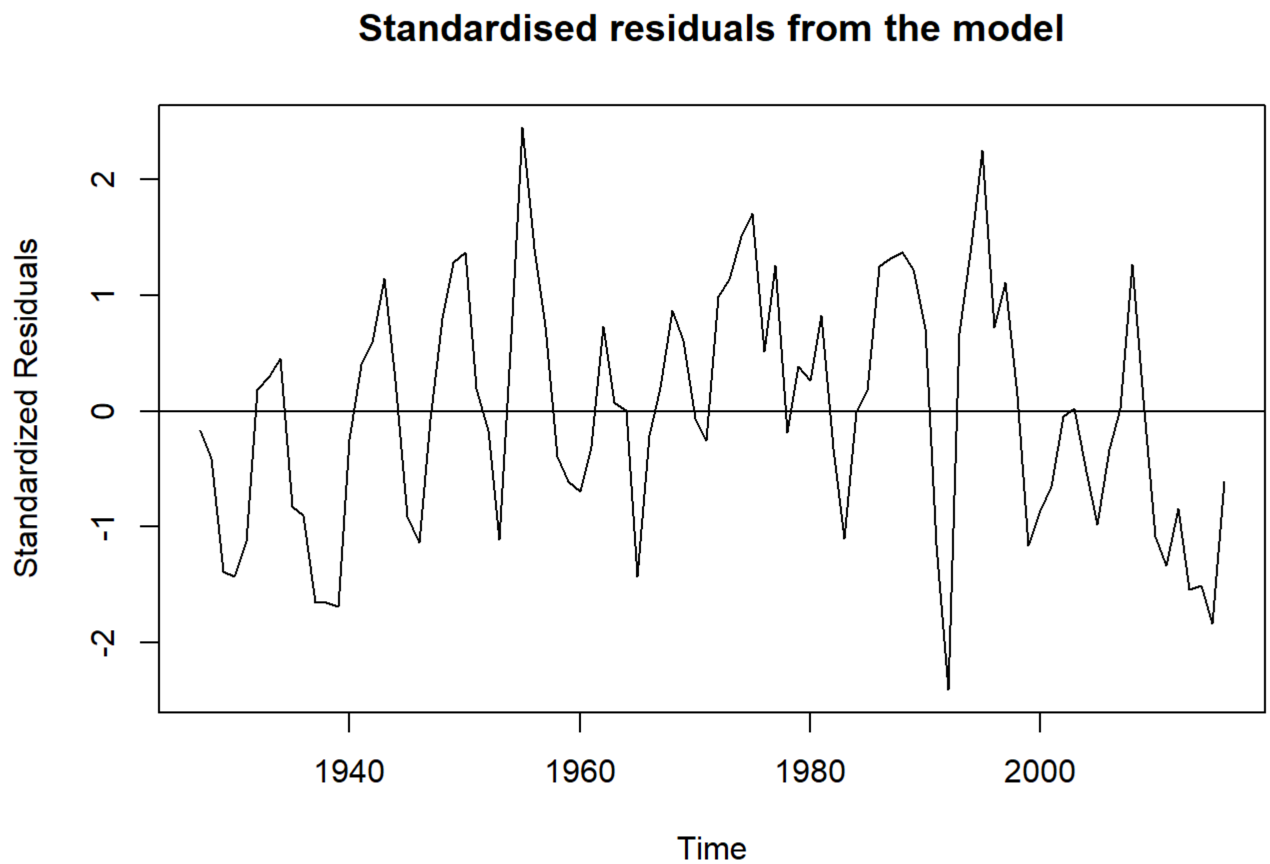
```
#Fitting the model
model_lm = lm(ozonethickness~time(ozonethickness))
summary(model_lm)
```

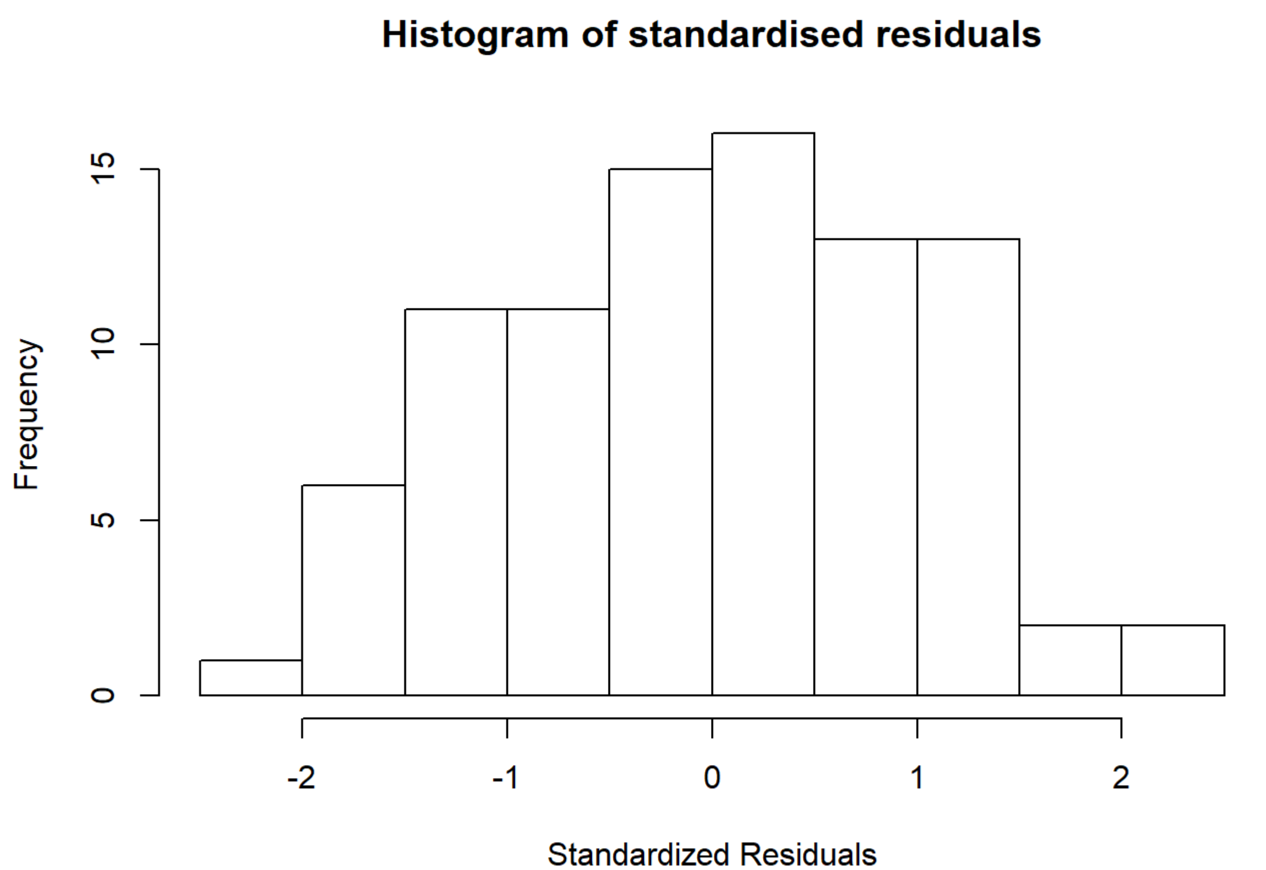
```
##
## Call:
## lm(formula = ozonethickness ~ time(ozonethickness))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7165 -1.6687  0.0275  1.4726  4.7940
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          213.720155  16.257158   13.15   <2e-16 ***
## time(ozonethickness)  -0.110029   0.008245  -13.34   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.032 on 88 degrees of freedom
## Multiple R-squared:  0.6693, Adjusted R-squared:  0.6655
## F-statistic: 178.1 on 1 and 88 DF,  p-value: < 2.2e-16
```

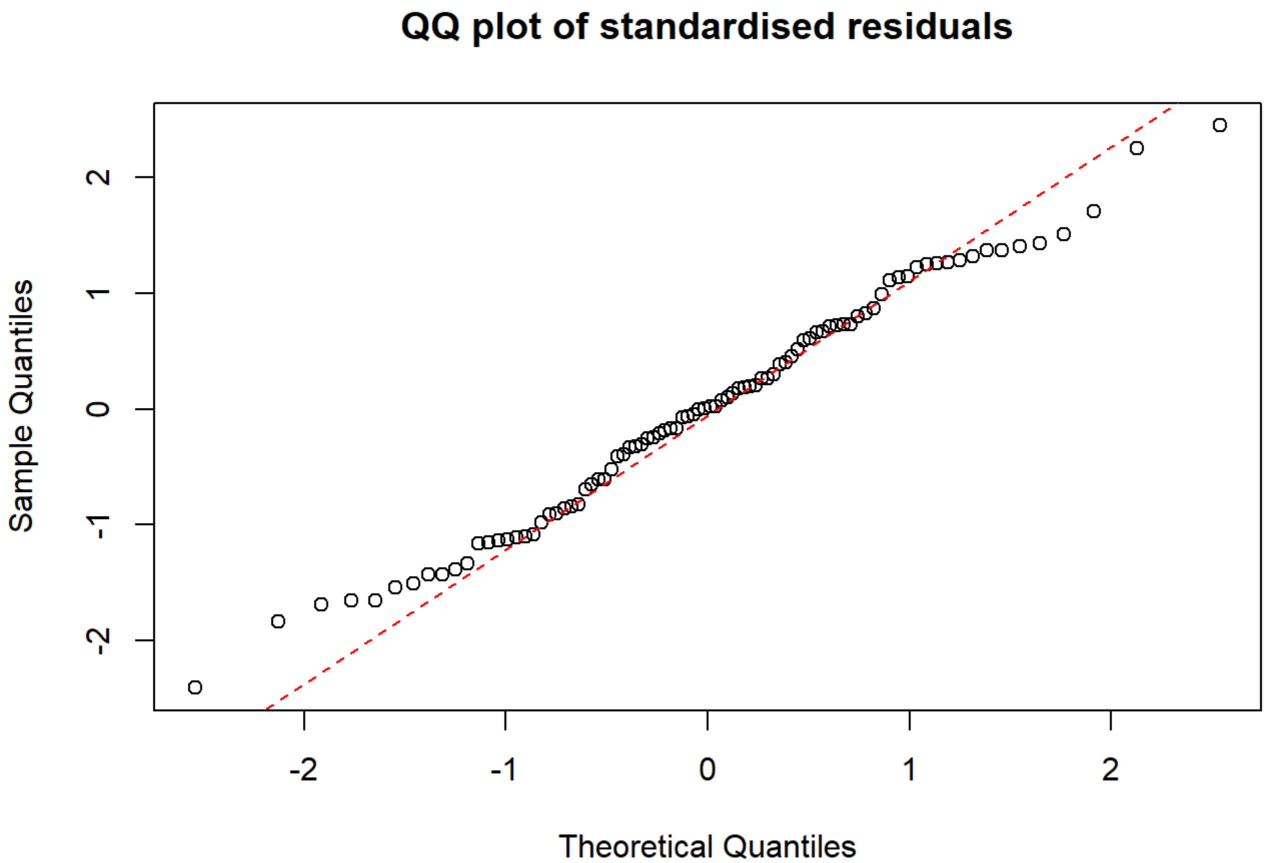
```
#Plotting the fitted model
plot(ozonethickness,
     type='o',
     ylab='Ozone Thickness',
     main = "Fitted linear model to Ozone Thickness Series")
abline(model_lm, col = 'red')
```

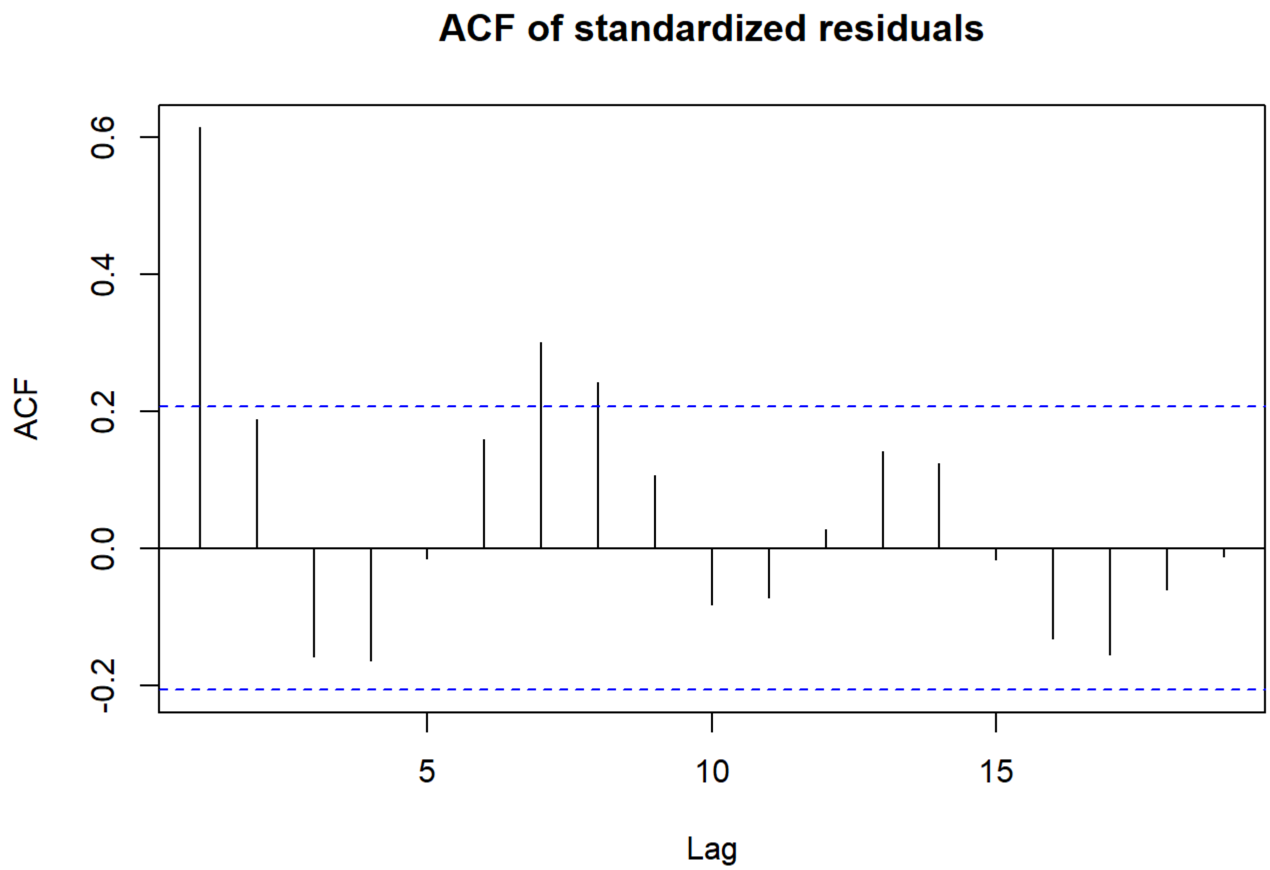


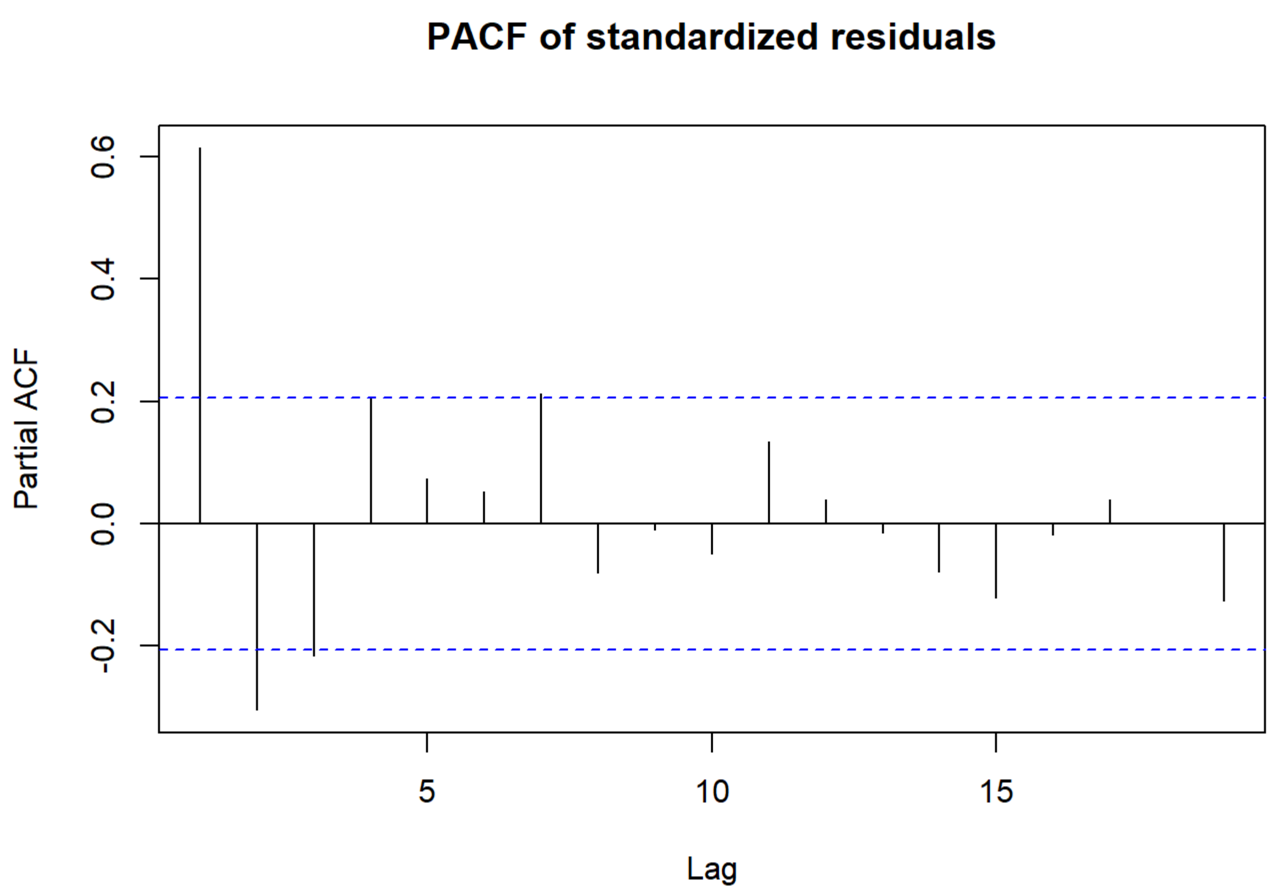

```
#Residual Analysis
residual_analysis(model_lm, ozonethickness)
```


Standardised residuals from the model
Standardized Residuals
2
1
0
-1
-2
1940
1960
1980
2000
Time


Histogram of standardised residuals
Frequency
15
10
5
0
-2
-1
0
1
2
Standardized Residuals


QQ plot of standardised residuals
Sample Quantiles
Theoretical Quantiles
-2
-1
0
1
2


ACF of standardized residuals
ACF
0.6
0.4
0.2
0.0
-0.2
5
10
15
Lag


PACF of standardized residuals
Partial ACF
0.6
0.4
0.2
0.0
-0.2
5
10
15
Lag

```
## $breaks
##  [1] -2.5 -2.0 -1.5 -1.0 -0.5  0.0  0.5  1.0  1.5  2.0  2.5
##
## $counts
##  [1]  1  6 11 11 15 16 13 13  2  2
##
## $density
##  [1] 0.02222222 0.13333333 0.24444444 0.24444444 0.33333333 0.35555556
##  [7] 0.28888889 0.28888889 0.04444444 0.04444444
##
## $mids
##  [1] -2.25 -1.75 -1.25 -0.75 -0.25  0.25  0.75  1.25  1.75  2.25
##
## $xname
## [1] "res_model_used"
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
```

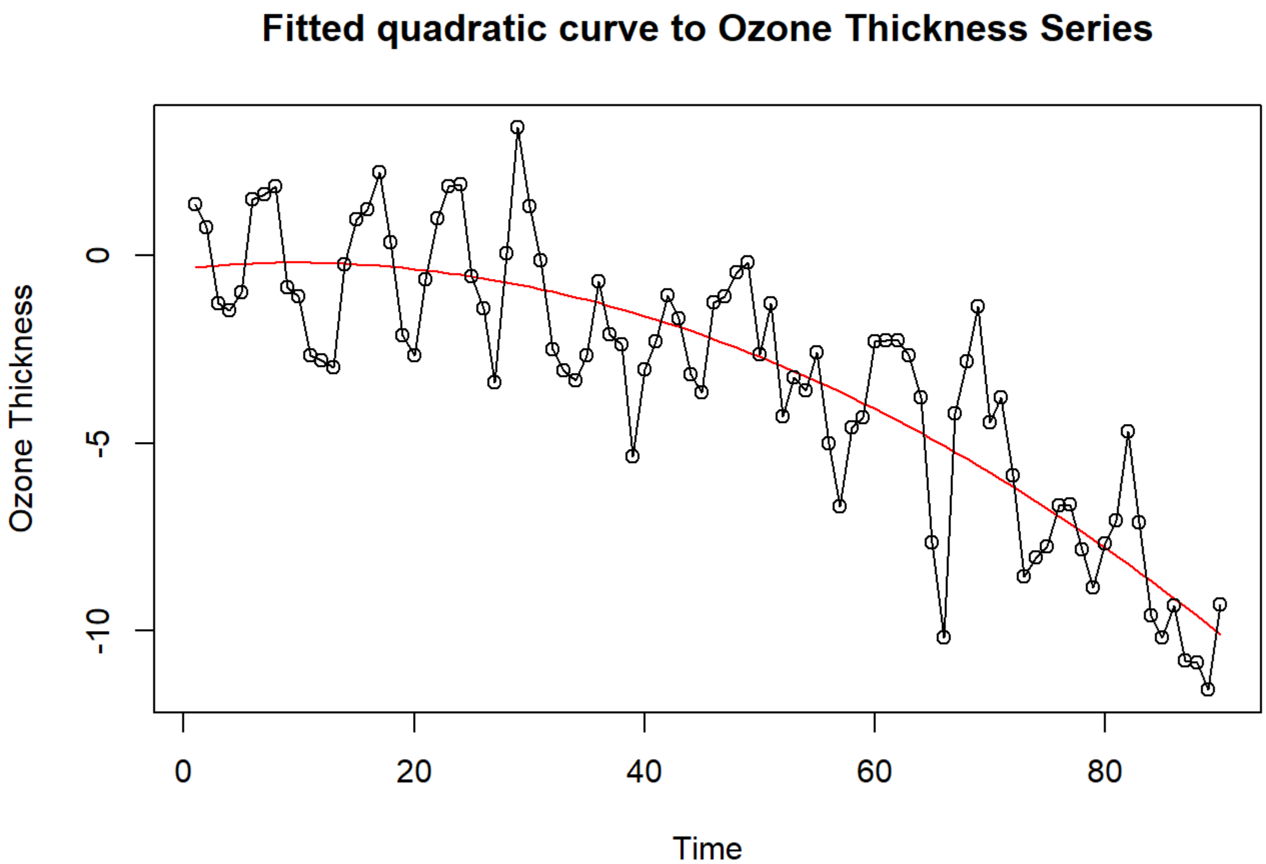
The p-value is less than 0.05 significance level. *R-squared value is 0.6693*. The plot of the standardized residuals is fitted evenly around 0. Adjusted R-squared is 0.6655. The histogram appears to be normally distributed. The QQ plot tails off at the higher and lower end values. The autocorrelation can be used to determine that the series is not a white noise.

Quadratic Model

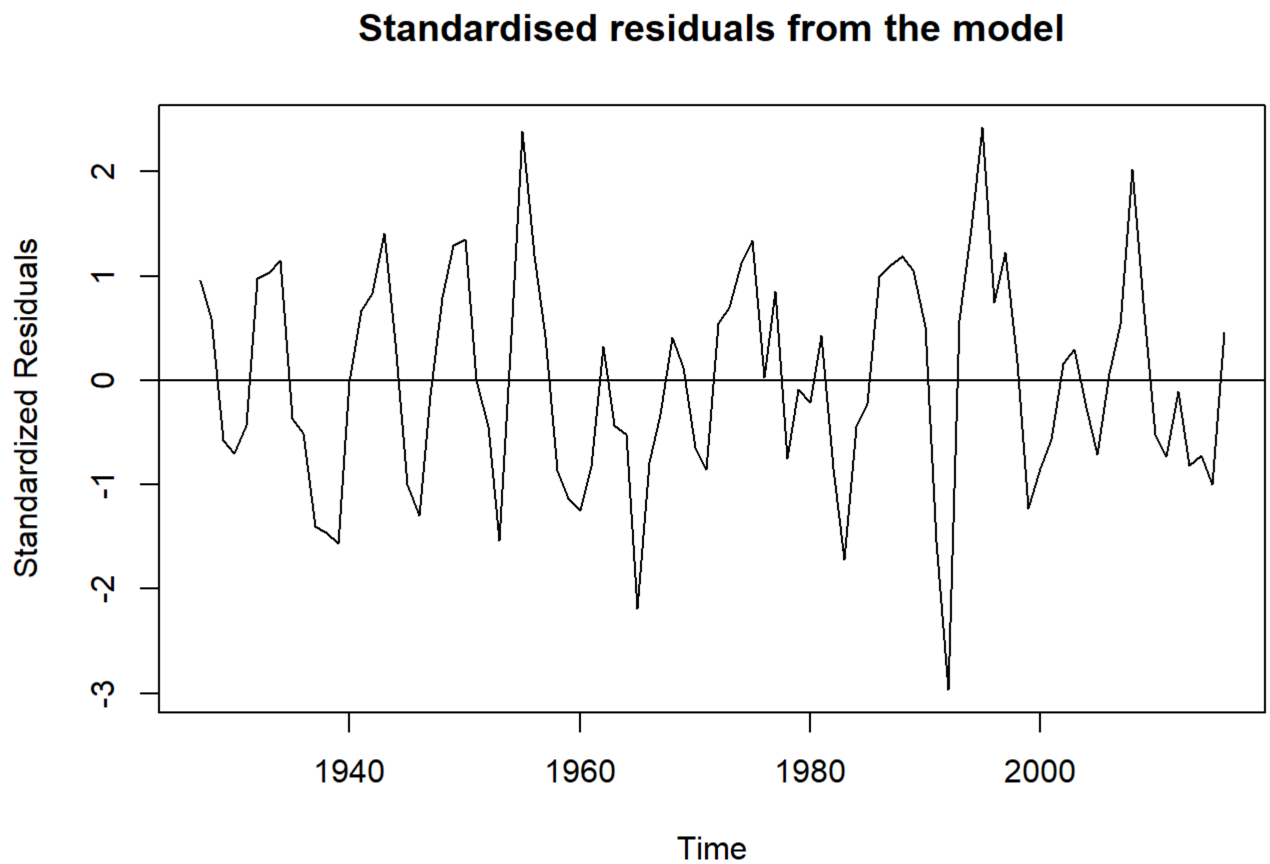
```
#Fitting the model
t = time(ozonethickness)
t2 = t^2
model_qd = lm(ozonethickness~t+t2)
summary(model_qd)
```

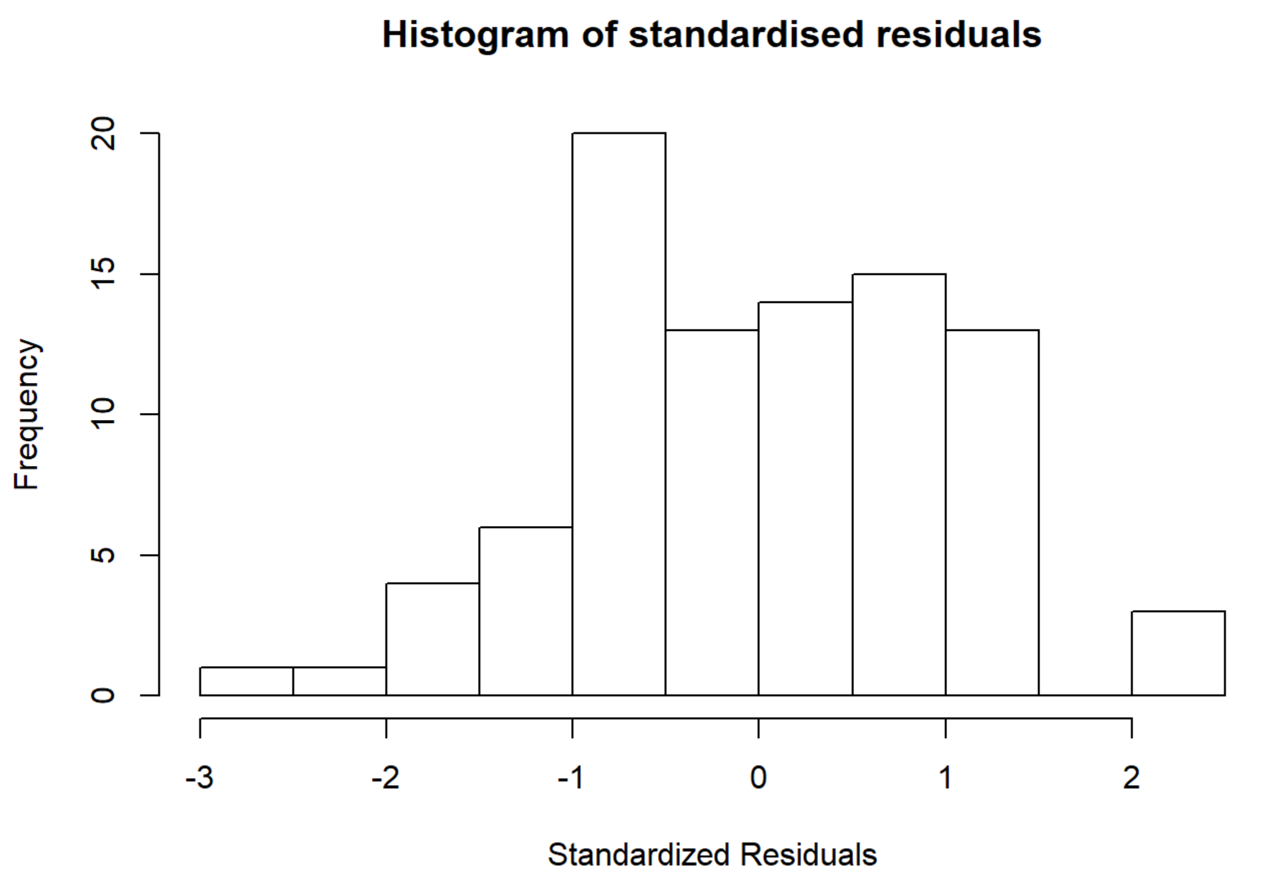
```
##
## Call:
## lm(formula = ozonethickness ~ t + t2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.1062 -1.2846 -0.0055  1.3379  4.2325
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.733e+03  1.232e+03  -4.654 1.16e-05 ***
## t            5.924e+00  1.250e+00   4.739 8.30e-06 ***
## t2          -1.530e-03  3.170e-04  -4.827 5.87e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.815 on 87 degrees of freedom
## Multiple R-squared:  0.7391, Adjusted R-squared:  0.7331
## F-statistic: 123.3 on 2 and 87 DF,  p-value: < 2.2e-16
```

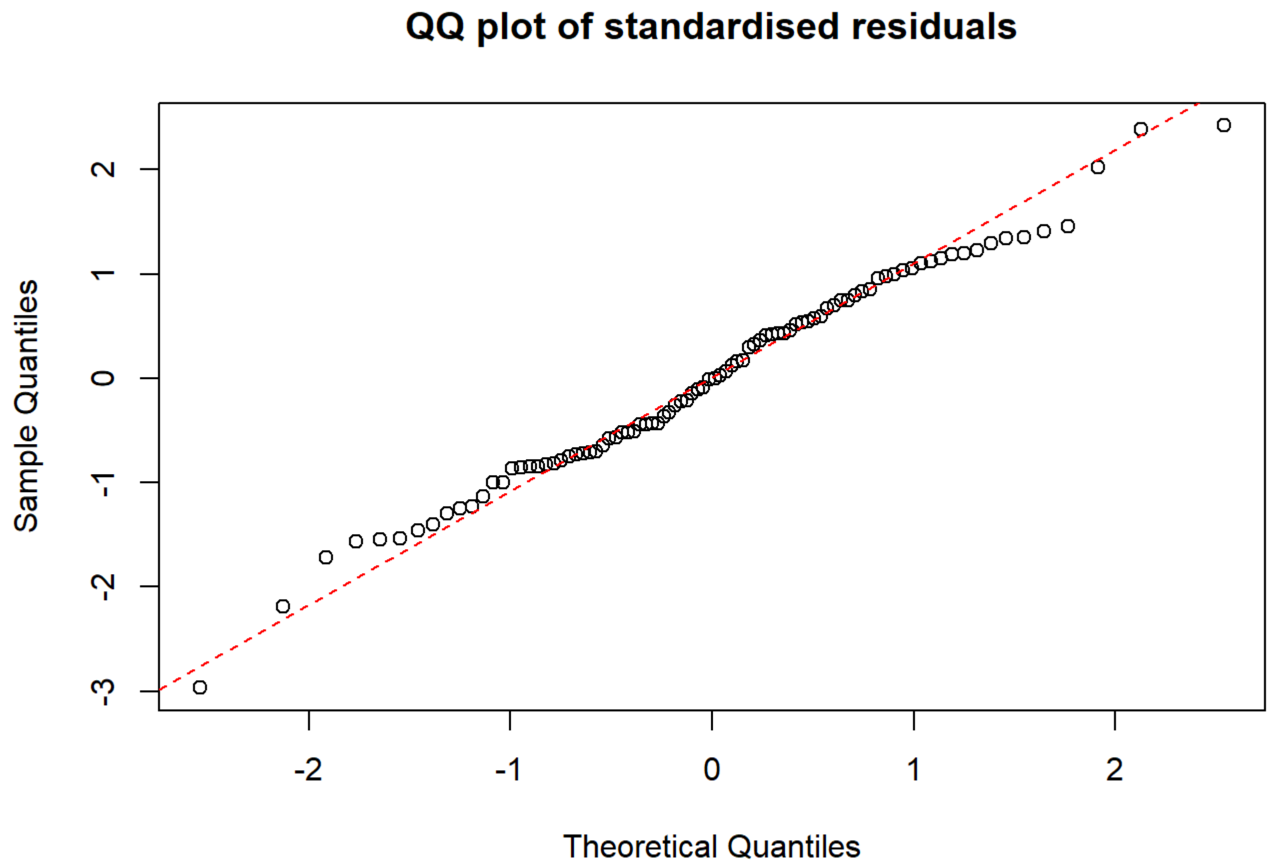
```
#Plotting the built model
plot(ts(fitted(model_qd)),
     ylim = c(min(c(fitted(model_qd),
     as.vector(ozonethickness))),
     max(c(fitted(model_qd),as.vector(ozonethickness)))),
     ylab='Ozone Thickness',
     main = "Fitted quadratic curve to Ozone Thickness Series",
     col = 'red')
lines(as.vector(ozonethickness), type="o")
```

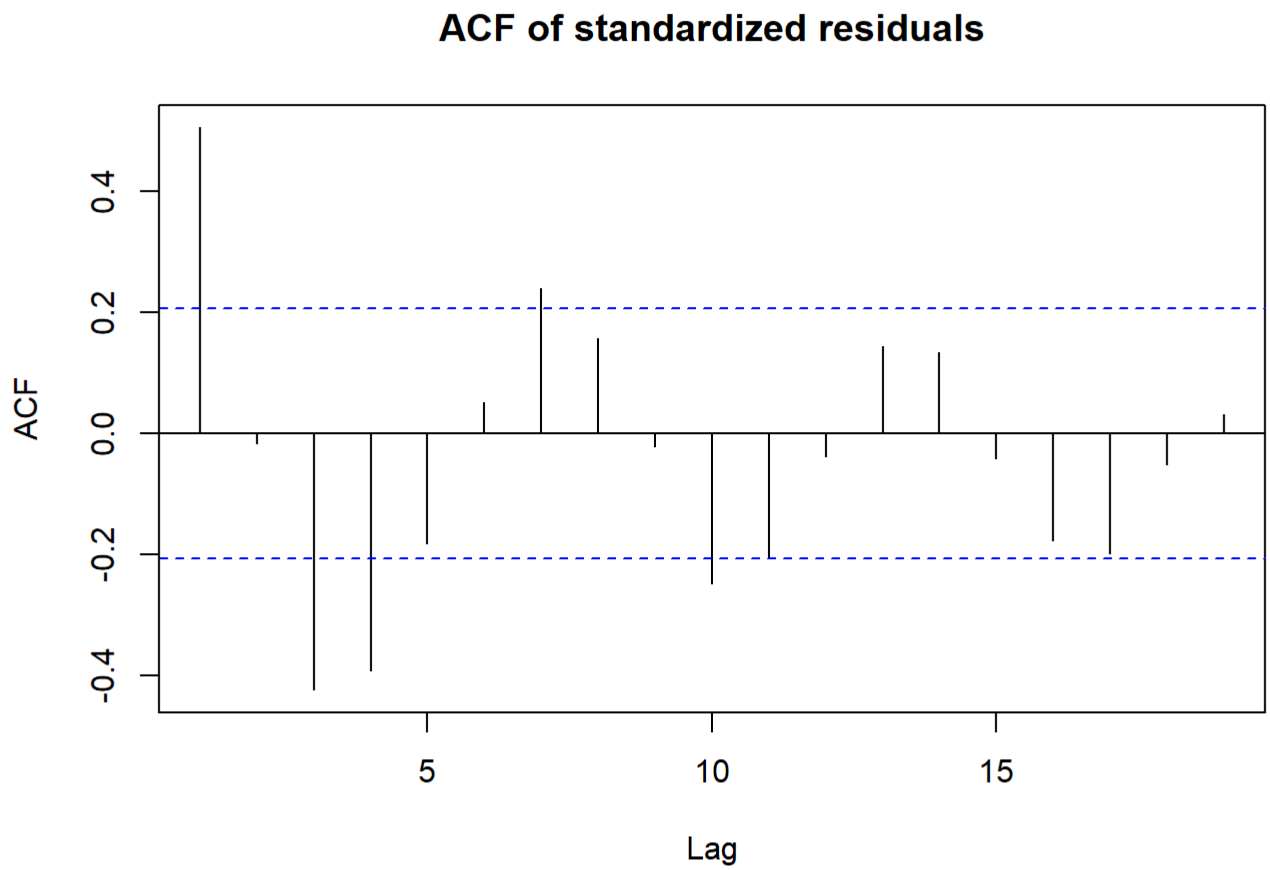


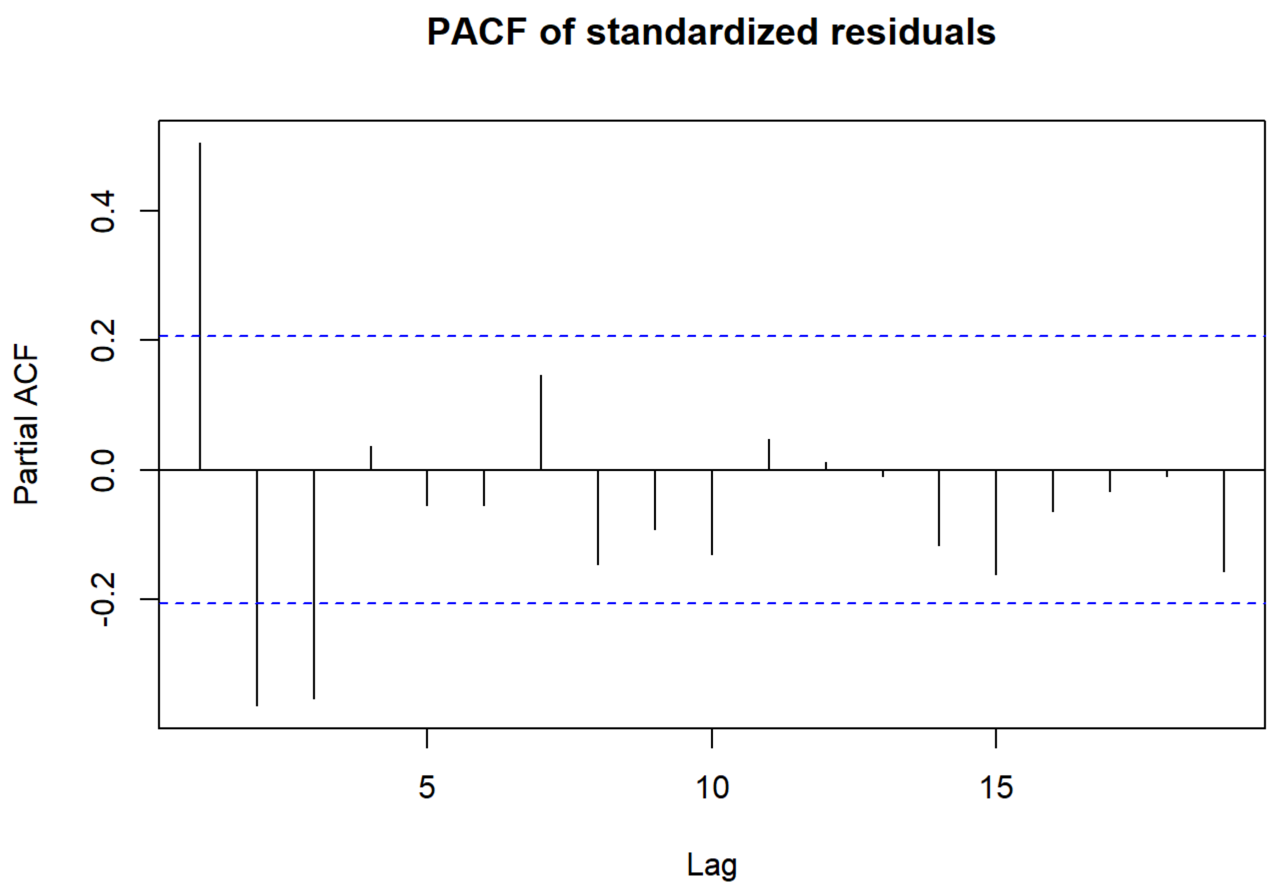

```
#Residual Analysis
residual_analysis(model_qd,ozonethickness)
```


Standardised residuals from the model
Standardized Residuals
-3
-2
-1
0
1
2
1940
1960
1980
2000
Time


Histogram of standardised residuals
Frequency
0
5
10
15
20
-3
-2
-1
0
1
2
Standardized Residuals


QQ plot of standardised residuals
Sample Quantiles
Theoretical Quantiles
-3
-2
-1
0
1
2
-2
-1
0
1
2


ACF of standardized residuals
ACF
0.4
0.2
0.0
-0.2
-0.4
5
10
15
Lag


PACF of standardized residuals
Partial ACF
0.4
0.2
0.0
-0.2
5
10
15
Lag

```
## $breaks
##  [1] -3.0 -2.5 -2.0 -1.5 -1.0 -0.5  0.0  0.5  1.0  1.5  2.0  2.5
##
## $counts
##  [1]  1  1  4  6 20 13 14 15 13  0  3
##
## $density
##  [1] 0.02222222 0.02222222 0.08888889 0.13333333 0.44444444 0.28888889
##  [7] 0.31111111 0.33333333 0.28888889 0.00000000 0.06666667
##
## $mids
##  [1] -2.75 -2.25 -1.75 -1.25 -0.75 -0.25  0.25  0.75  1.25  1.75  2.25
##
## $xname
## [1] "res_model_used"
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
```

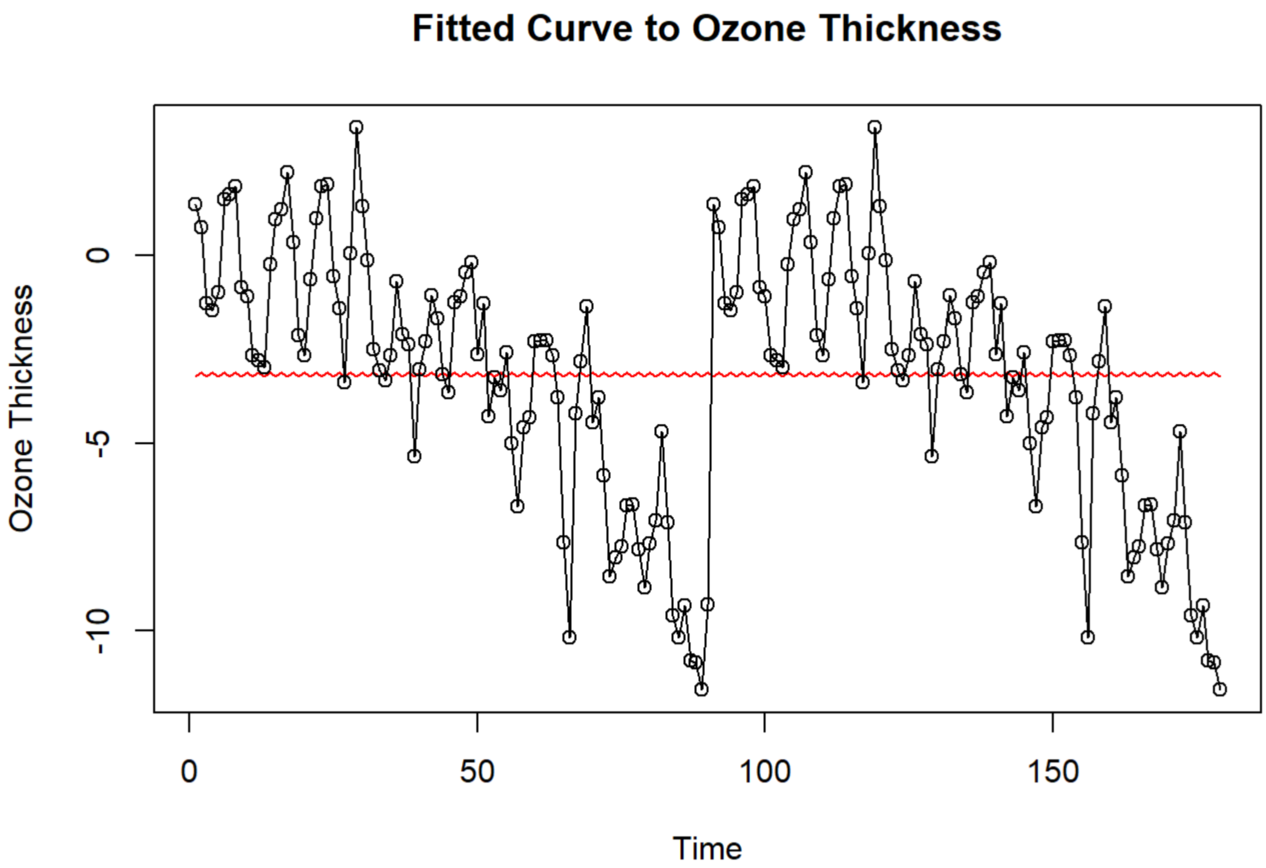
The p-value is less than 0.05 significance level. *R-squared value is 0.7391*. The plot of the standardized residuals is fitted evenly around 0. Adjusted R-squared is 0.6655. The histogram appears to be normally distributed with a slight skew at the left edges. The QQ plot tails off at the higher and lower end values. The autocorrelation can be used to determine that the series is not a white noise.

Seasonal and Cyclic Trends

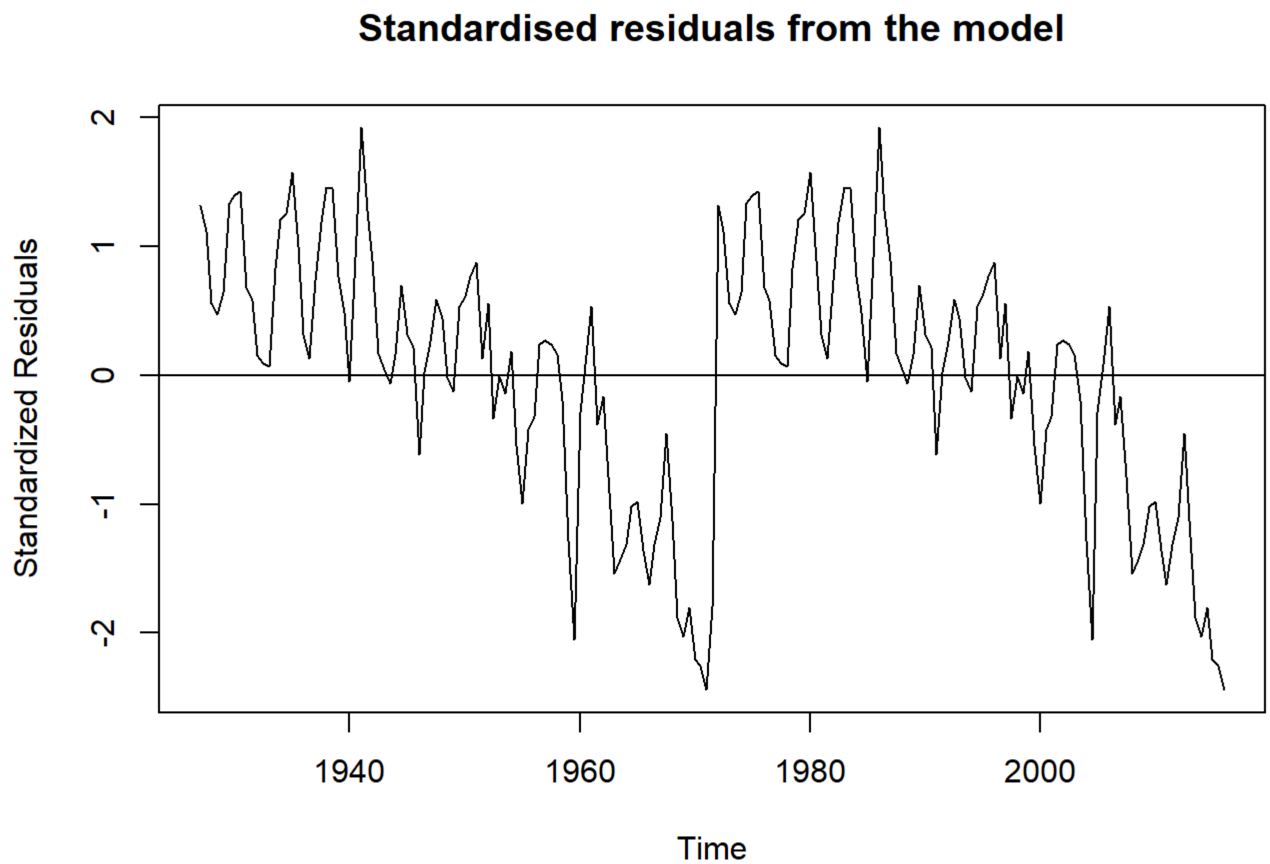
```
#Selecting the same data in a different dataframe because we want to set a frequency for the cyc
lic/seasonal model
ozonethickness2 <- ts(as.vector(ozonethickness), start=1927, end=2016, frequency = 2)
month = season(ozonethickness2)
model_cyc = lm(ozonethickness2~month - 1)
summary(model_cyc)
```

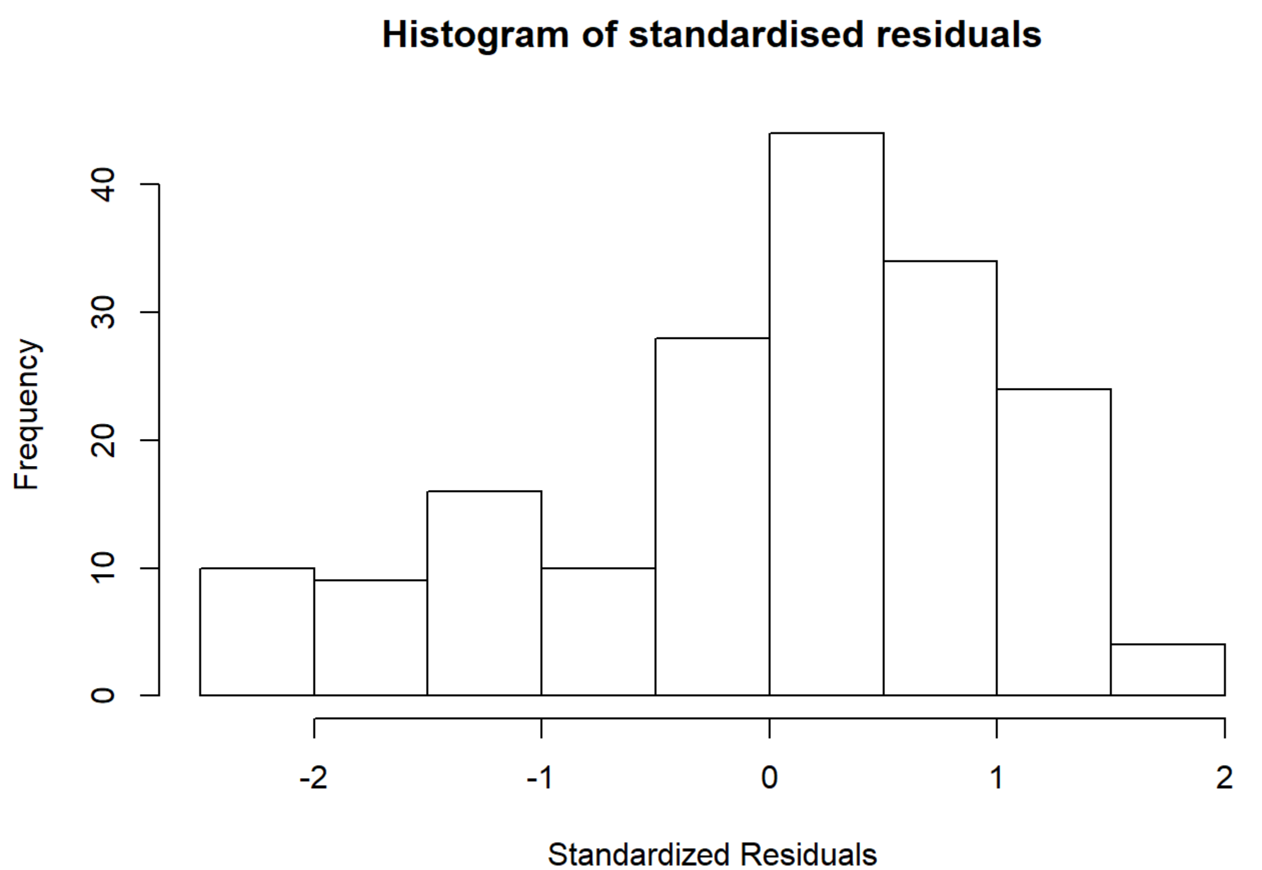
```
##
## Call:
## lm(formula = ozonethickness2 ~ month - 1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.3605 -1.7335  0.5545  2.3792  6.6261
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## monthSeason-1  -3.2189     0.3682  -8.742 1.72e-15 ***
## monthSeason-2  -3.1169     0.3703  -8.418 1.26e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.493 on 177 degrees of freedom
## Multiple R-squared:  0.4542, Adjusted R-squared:  0.448
## F-statistic: 73.64 on 2 and 177 DF,  p-value: < 2.2e-16
```

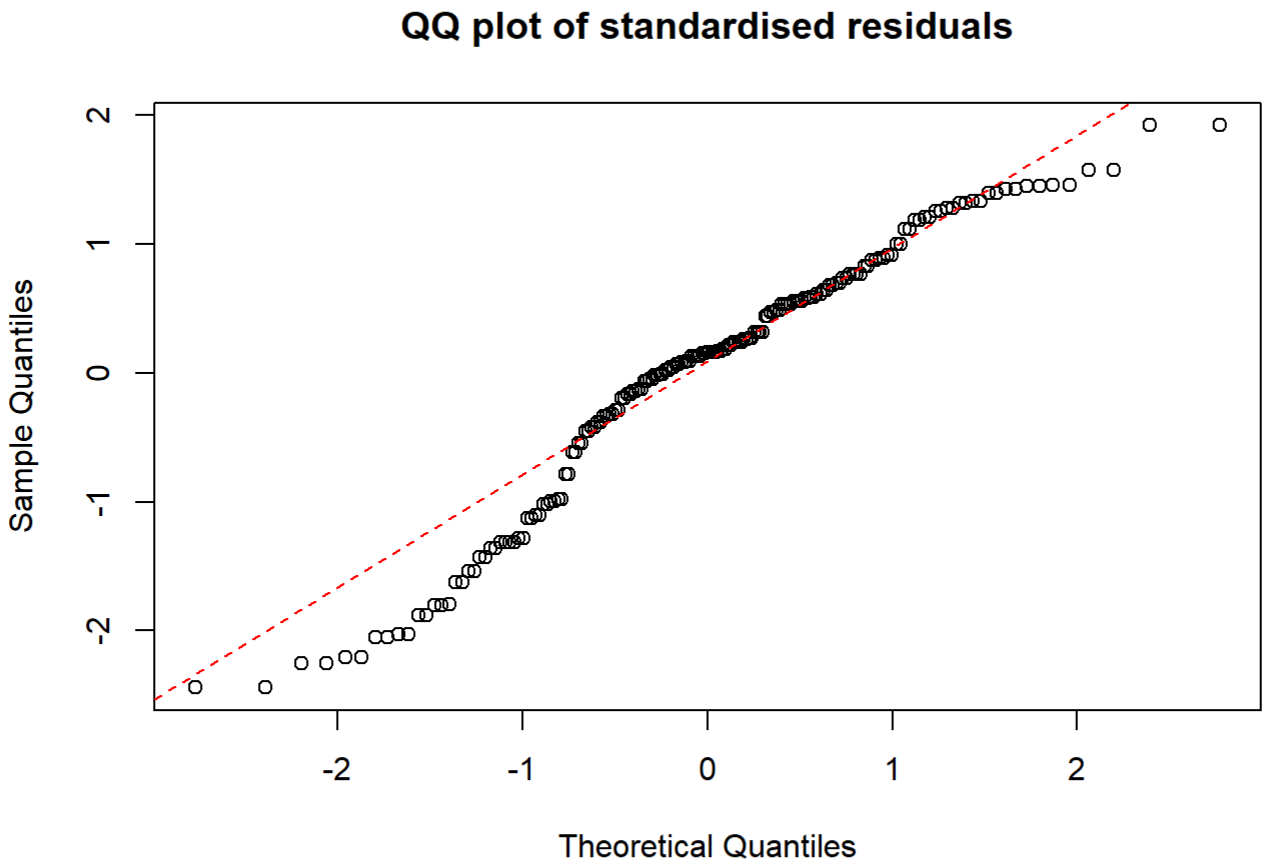
```
plot(ts(fitted(model_cyc)),
     ylim = c(min(c(fitted(model_cyc), as.vector(ozonethickness2))),
              max(c(fitted(model_cyc),as.vector(ozonethickness2)))),
     ylab='Ozone Thickness',
     main = "Fitted Curve to Ozone Thickness",
     col="red")
lines(as.vector(ozonethickness2),type="o")
```

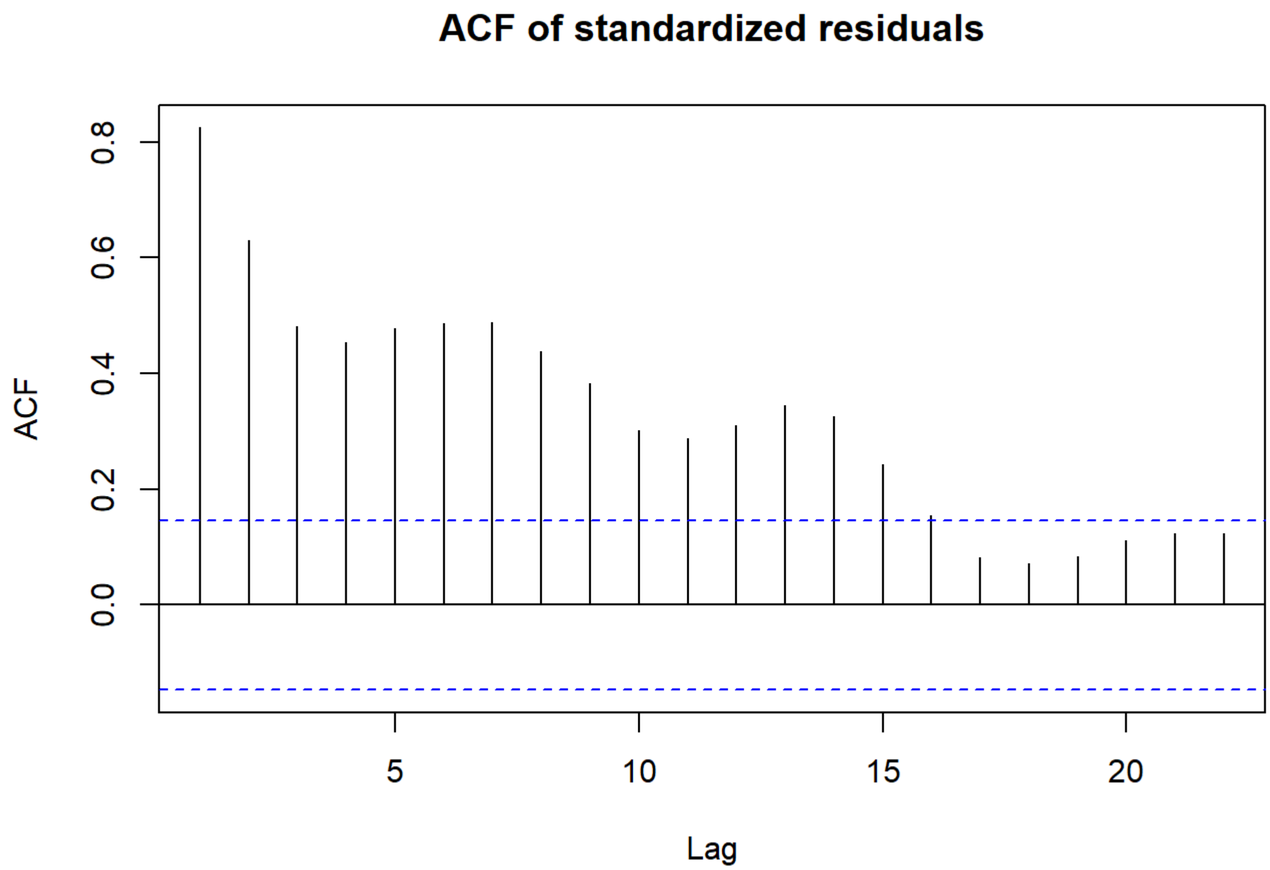


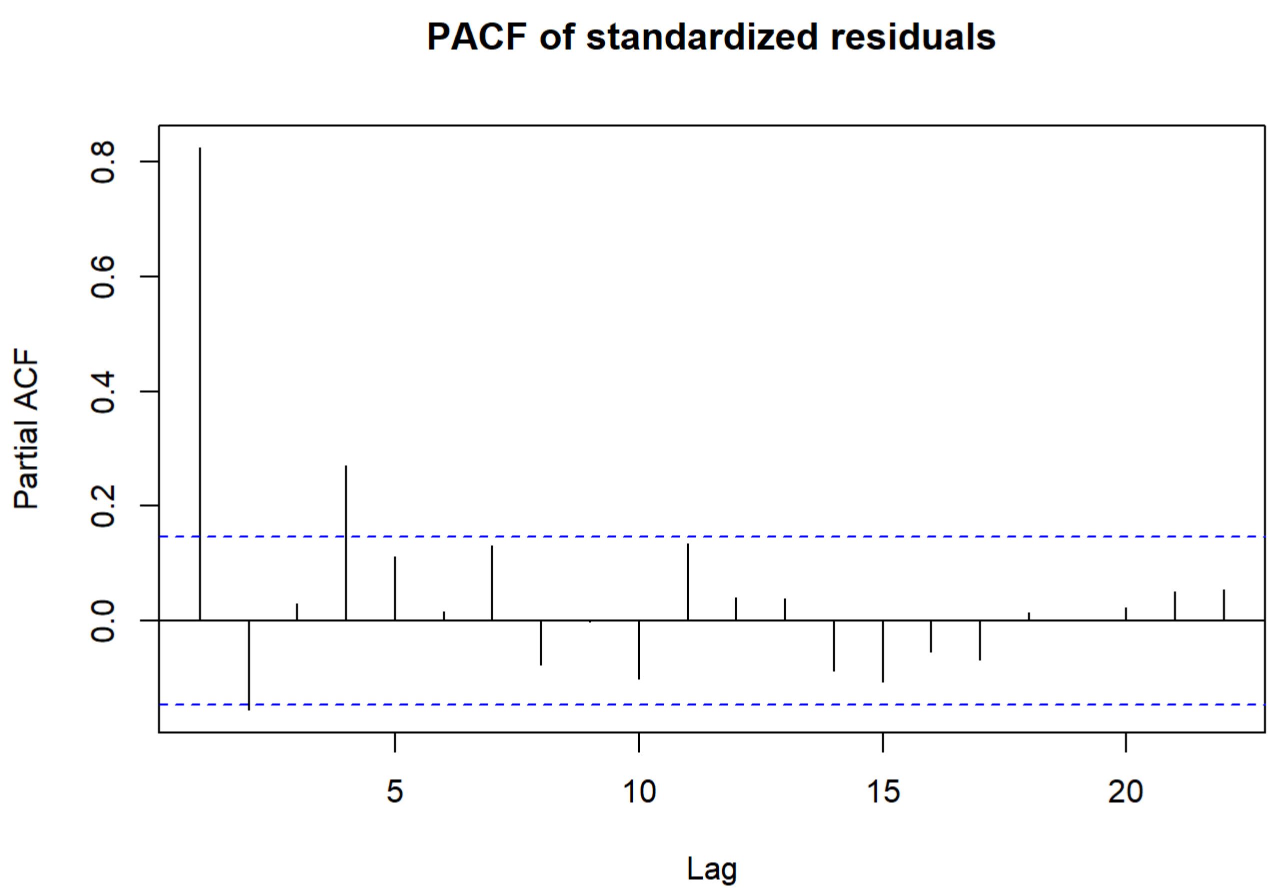

```
residual_analysis(model_cyc,ozonethickness2)
```


Standardised residuals from the model
Standardized Residuals
2
1
0
-1
-2
1940
1960
1980
2000
Time


Histogram of standardised residuals
Frequency
40
30
20
10
0
-2
-1
0
1
2
Standardized Residuals


QQ plot of standardised residuals
Sample Quantiles
Theoretical Quantiles
-2
-1
0
1
2


ACF of standardized residuals
ACF
0.0
0.2
0.4
0.6
0.8
5
10
15
20
Lag


PACF of standardized residuals
Partial ACF
0.0
0.2
0.4
0.6
0.8
5
10
15
20
Lag

```
## $breaks
##  [1] -2.5 -2.0 -1.5 -1.0 -0.5  0.0  0.5  1.0  1.5  2.0
##
## $counts
## [1] 10  9 16 10 28 44 34 24  4
##
## $density
## [1] 0.11173184 0.10055866 0.17877095 0.11173184 0.31284916 0.49162011 0.37988827
## [8] 0.26815642 0.04469274
##
## $mids
## [1] -2.25 -1.75 -1.25 -0.75 -0.25  0.25  0.75  1.25  1.75
##
## $xname
## [1] "res_model_used"
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
```

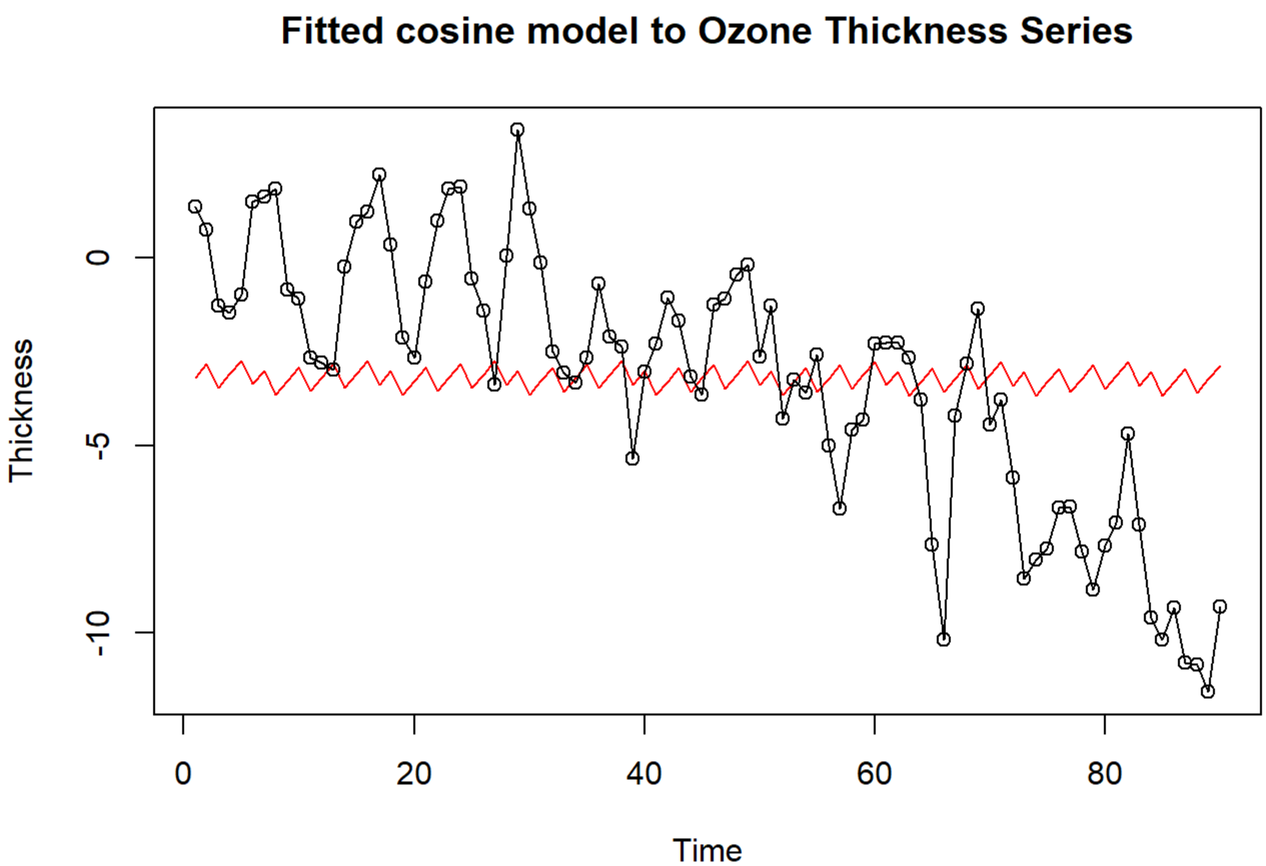
We have tryed to fit the time series into a cyclic and seasonal model by setting the frequency 2. The p-value is less than 0.05 significance level. *R-squared value is 0.4542*. The R-squared value is smaller than the quadratic and linear models. The series is not cyclic or seasonal and hence does not fit the cyclic or seasonal model.

Cosine Trends

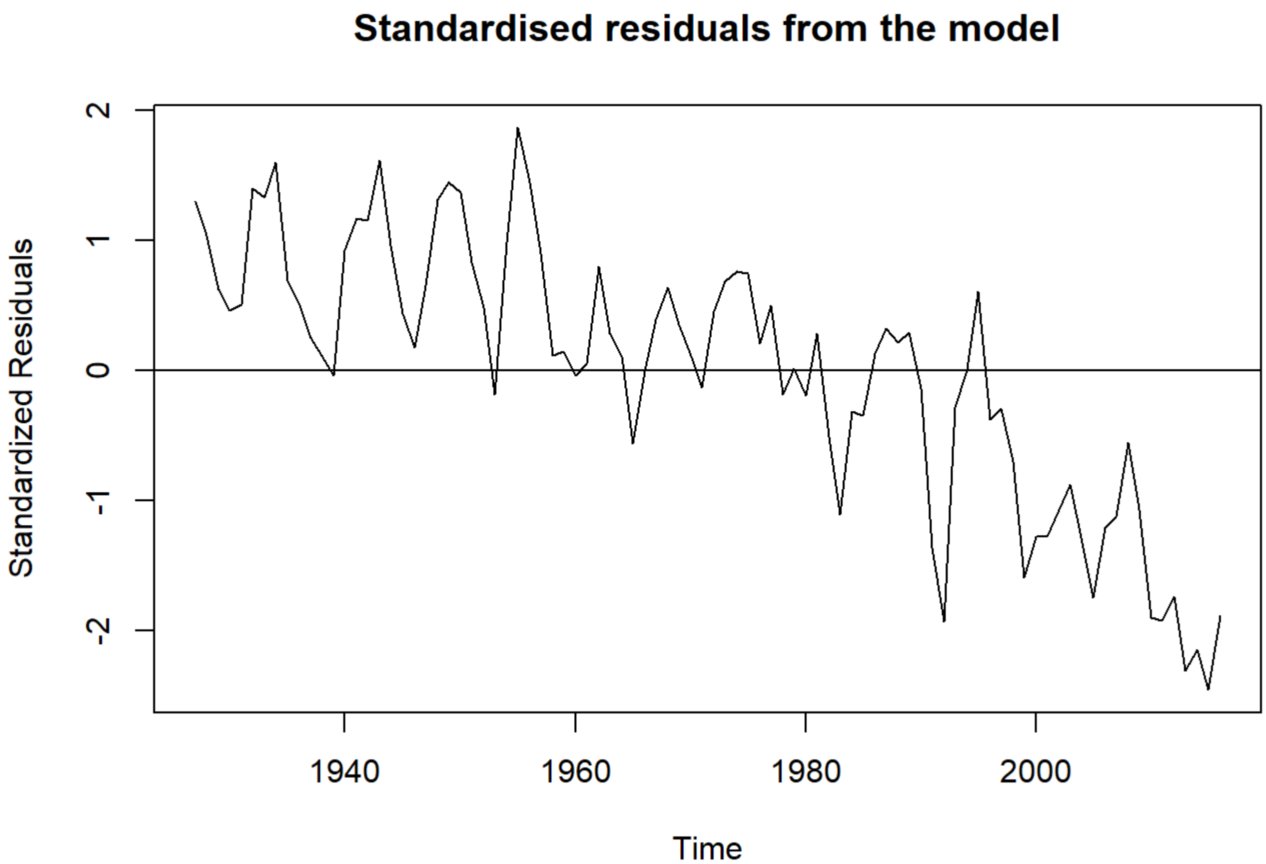
```
har=harmonic(ozonethickness,0.5)
model_cos=lm(ozonethickness~har)
summary(model_cos)
```

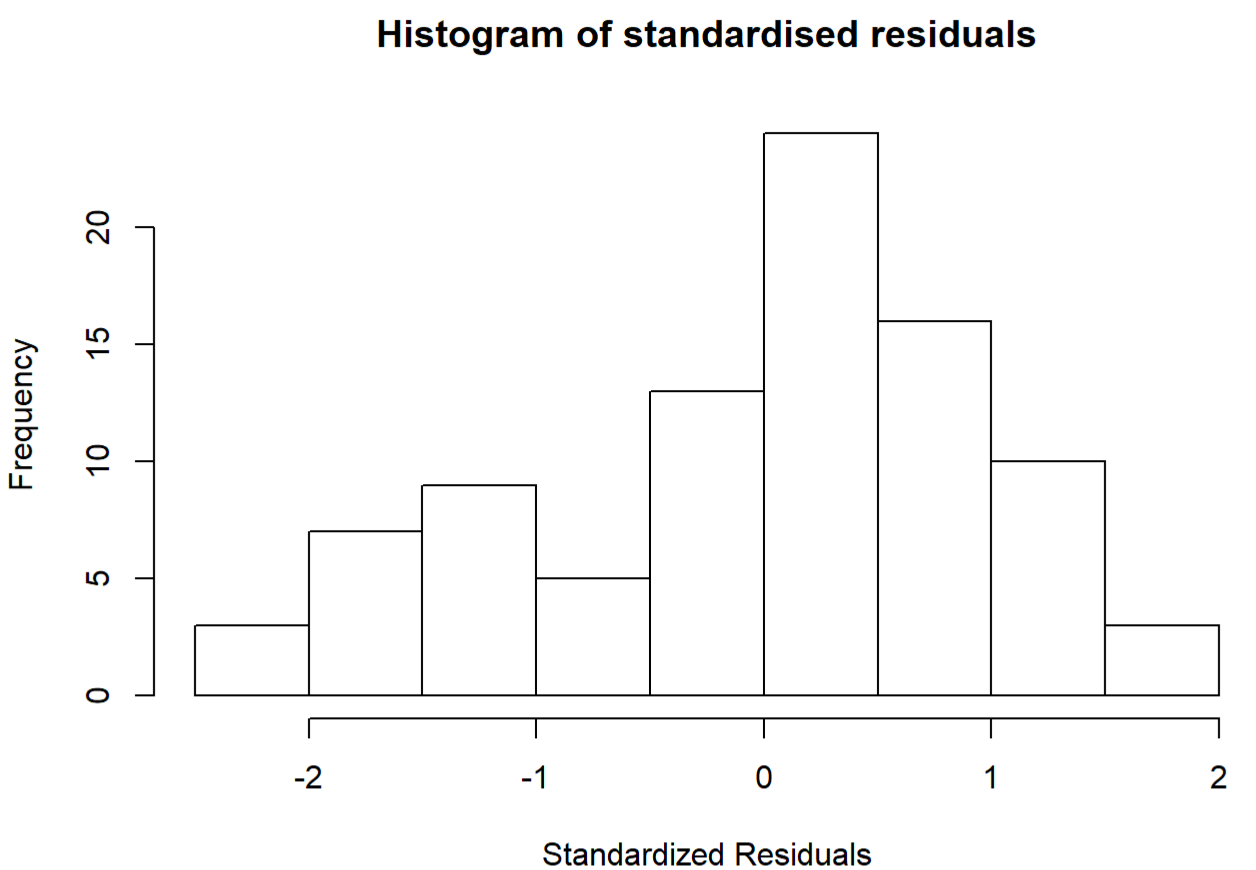
```
##
## Call:
## lm(formula = ozonethickness ~ har)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.3520 -1.8905  0.4837  2.3643  6.4248
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.970e+00  4.790e-01  -6.199 1.79e-08 ***
## har          5.462e+11  7.105e+11   0.769    0.444
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.522 on 88 degrees of freedom
## Multiple R-squared:  0.006672,   Adjusted R-squared:  -0.004616
## F-statistic: 0.5911 on 1 and 88 DF,  p-value: 0.4441
```

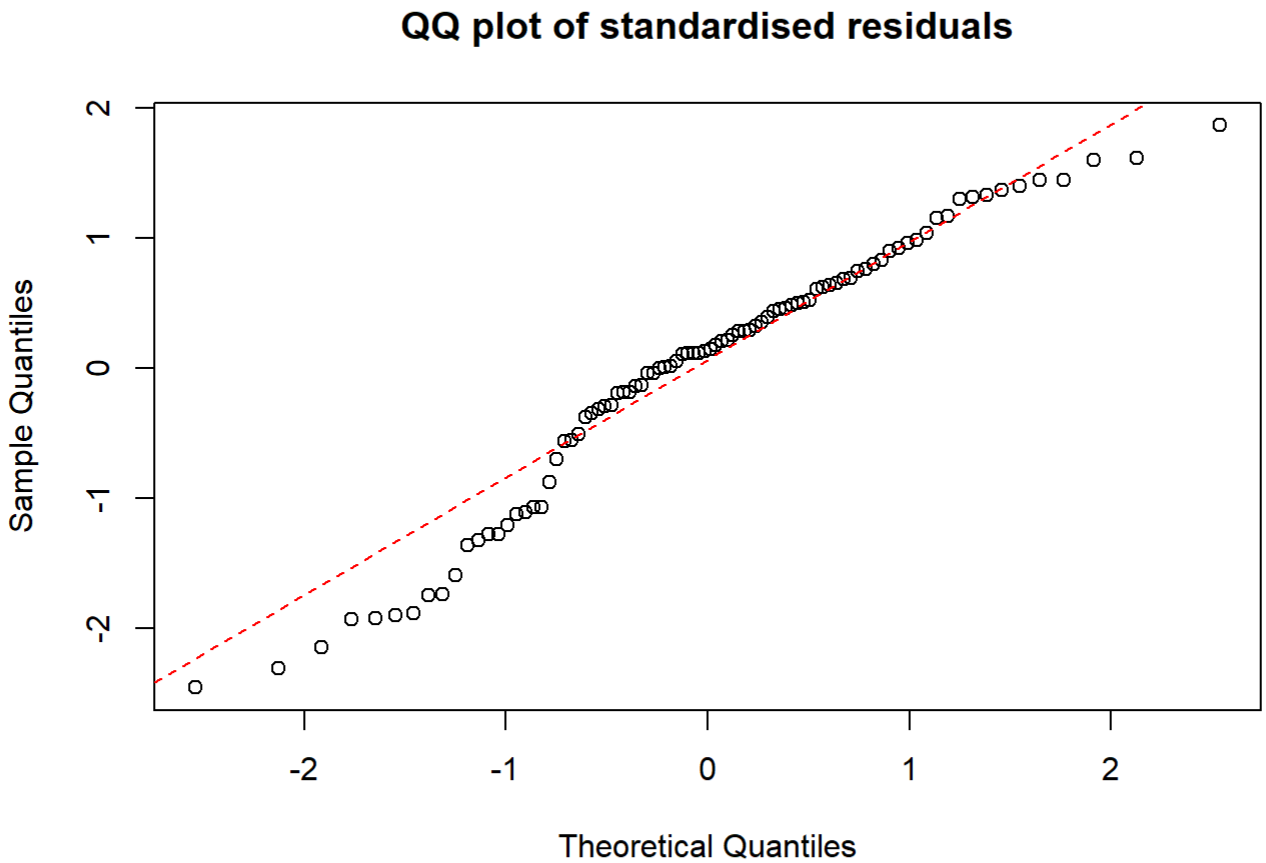
```
plot(ts(fitted(model_cos)),
     ylab='Thickness',
     type='l',
     ylim=range(c(fitted(model_cos),ozonethickness)),
     main="Fitted cosine model to Ozone Thickness Series",
     col = "red")
lines(as.vector(ozonethickness),type="o")
```

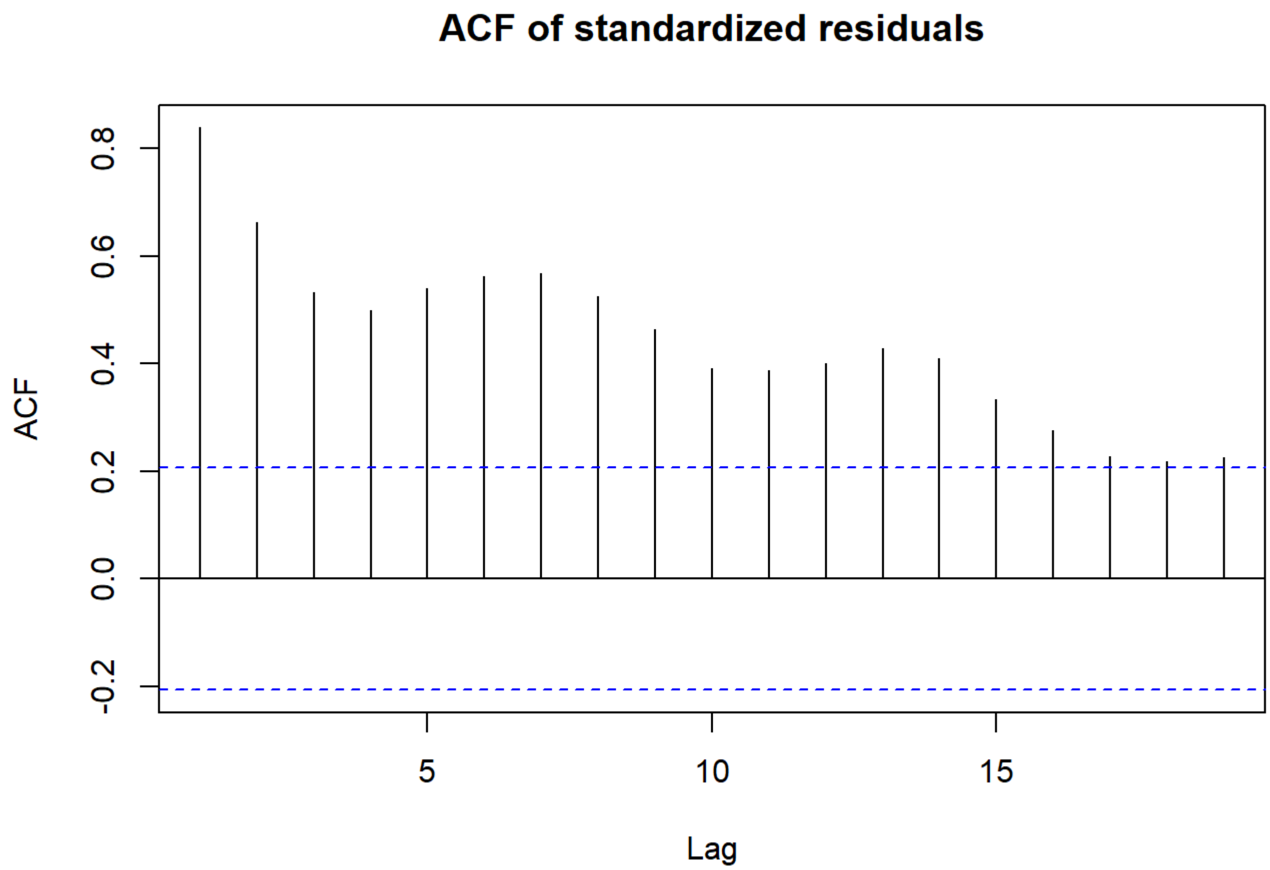


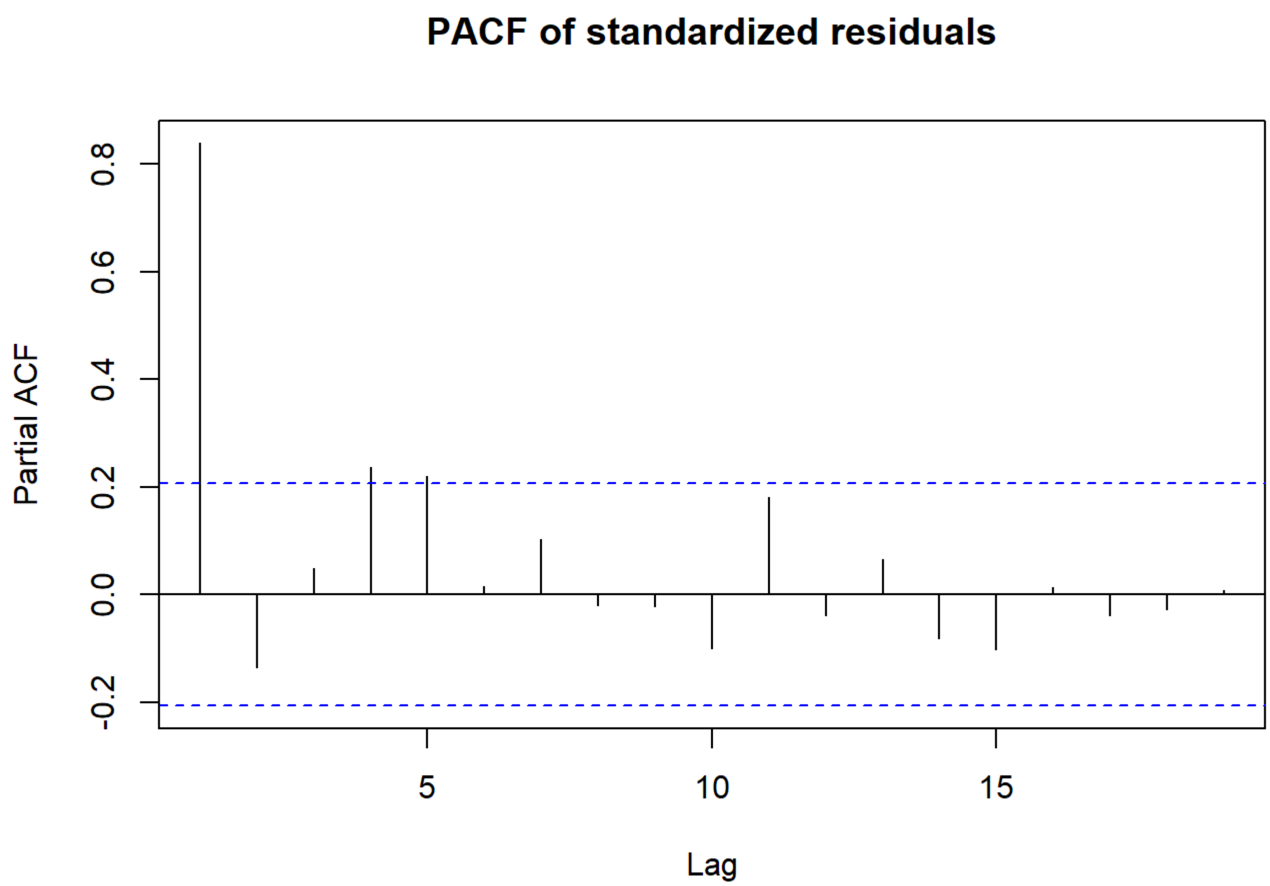

```
residual_analysis(model_cos,ozonethickness)
```


Standardised residuals from the model
Standardized Residuals
2
1
0
-1
-2
1940
1960
1980
2000
Time


Histogram of standardised residuals
Frequency
0
5
10
15
20
-2
-1
0
1
2
Standardized Residuals


QQ plot of standardised residuals
Sample Quantiles
Theoretical Quantiles
-2
-1
0
1
2


ACF of standardized residuals
ACF
0.8
0.6
0.4
0.2
0.0
-0.2
5
10
15
Lag


PACF of standardized residuals
Partial ACF
0.8
0.6
0.4
0.2
0.0
-0.2
5
10
15
Lag

```
## $breaks
##  [1] -2.5 -2.0 -1.5 -1.0 -0.5  0.0  0.5  1.0  1.5  2.0
##
## $counts
## [1]  3  7  9  5 13 24 16 10  3
##
## $density
## [1] 0.06666667 0.15555556 0.20000000 0.11111111 0.28888889 0.53333333 0.35555556
## [8] 0.22222222 0.06666667
##
## $mids
## [1] -2.25 -1.75 -1.25 -0.75 -0.25  0.25  0.75  1.25  1.75
##
## $xname
## [1] "res_model_used"
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
```

The p-value is more than 0.05 significance level. Hence, we reject the null hypothesis. We can also see that the model does not fit the series.

Forecasting

The best fitting model obtained is the Quadratic Model as the R and R-squared values of the Quadratic Model are greater than all the other fitted models.

We will now use the quadratic model and forecast the ozone layer thickness for the next 5 years i.e. 2017 to 2021.

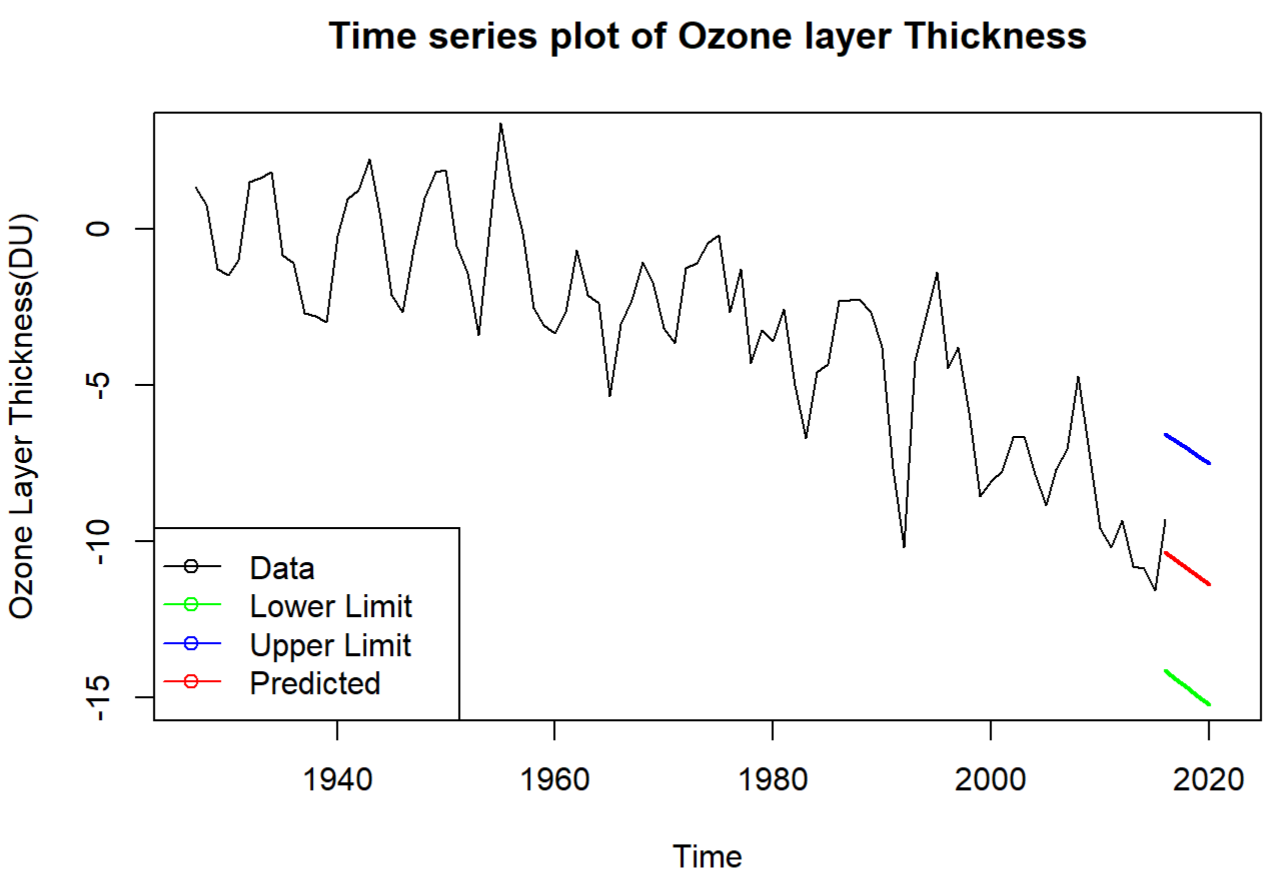
```
#Creating a time vector for the next five years after 2016
t = c(2017,2018,2019,2020,2021)
t2 = t^2

print(t2)
```

```
## [1] 4068289 4072324 4076361 4080400 4084441
```

```
#Creating a new data frame with to_forcast and to_forcast2
df_to_forcast = data.frame(t,t2)
forecasts = predict(model_qd, df_to_forcast, interval = "prediction")
print(forecasts)
```

```
##         fit       lwr       upr
## 1 -10.34387 -14.13556 -6.552180
## 2 -10.59469 -14.40282 -6.786548
## 3 -10.84856 -14.67434 -7.022786
## 4 -11.10550 -14.95015 -7.260851
## 5 -11.36550 -15.23030 -7.500701
```

```
#Plotting the time seties including the predicted 5 years
plot(ozonethickness,
     ylab = "Ozone Layer Thickness(DU)",
     ylim = c(-15,3),
     xlim = c(1927,2021),
     main = "Time series plot of Ozone layer Thickness")

lines(ts(as.vector(forecasts[,1]), start = 2016), col="red", type="l",lwd=2)
lines(ts(as.vector(forecasts[,2]), start = 2016), col="green", type="l",lwd=2)
lines(ts(as.vector(forecasts[,3]), start = 2016), col="blue", type="l",lwd=2)

legend("bottomleft",
       lty=1,
       pch=1,
       col=c("black","green","blue","red"),
       text.width = 18,
       c("Data","Lower Limit","Upper Limit", "Predicted"))
```

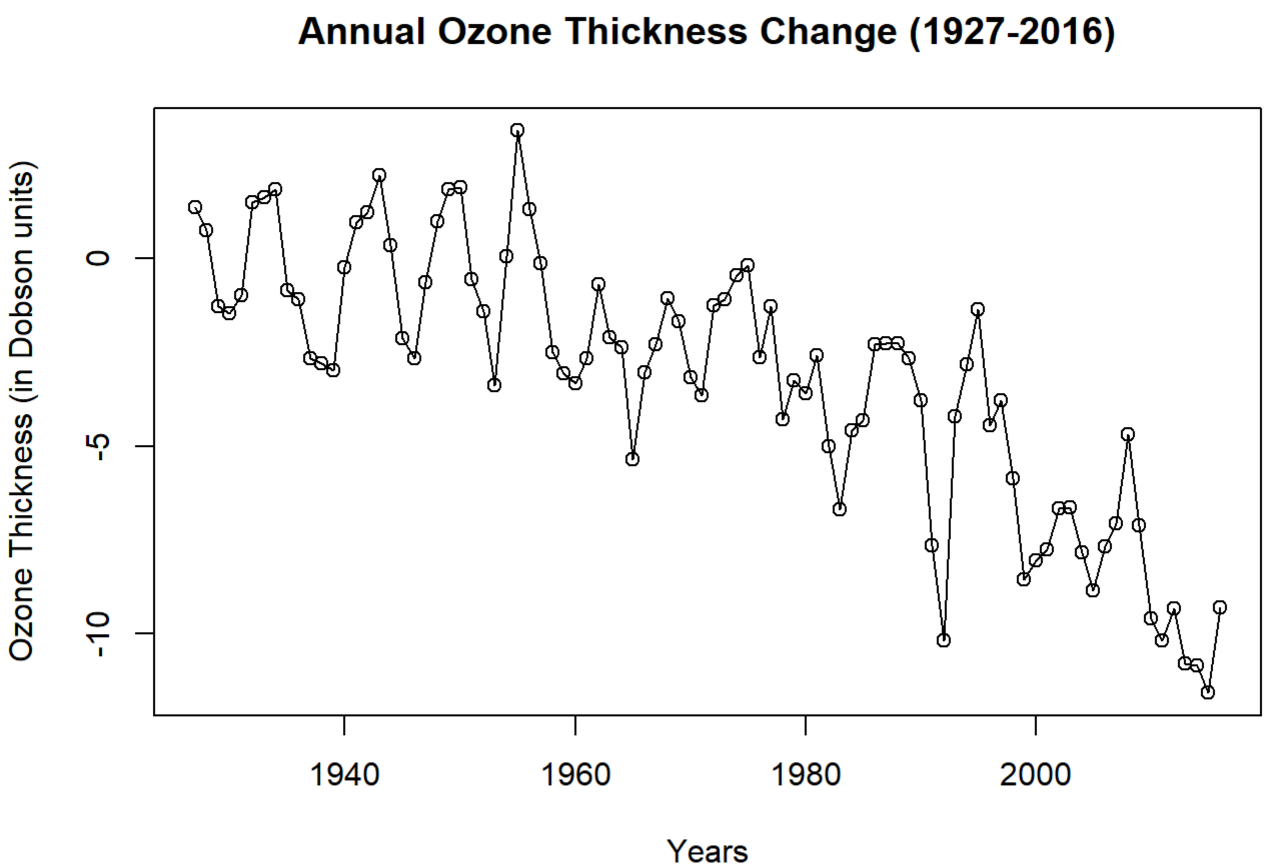


We can see that the ozone layer is going to get thinner in the next 5 years as predicted by the data. The predictions can vary between the upper and lower limit i.e. the 5% limit.

Proposing a set of possible ARIMA(p,d,q) models (Solution of Task 2)

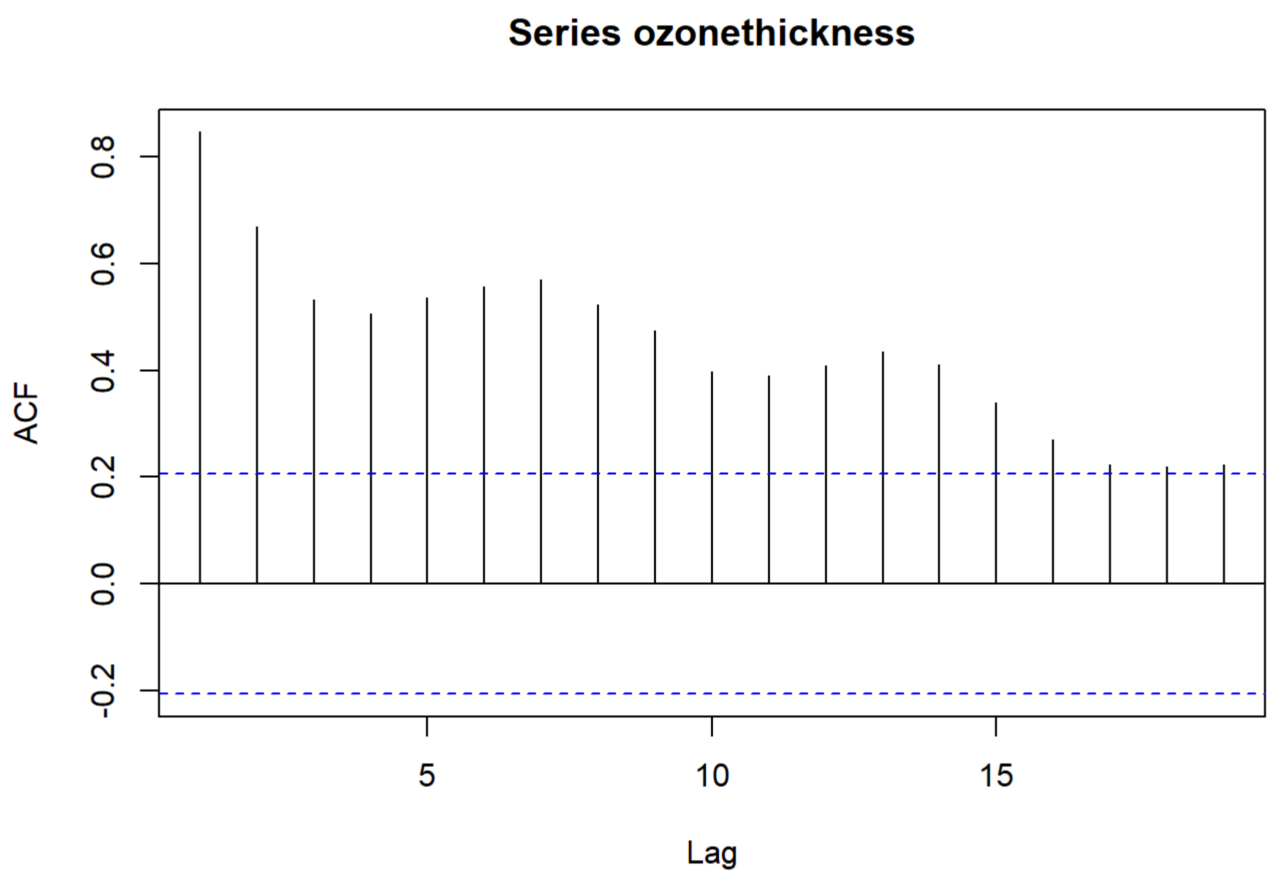
Plot of the original time series data

```
plot(ozonethickness,
     ylab='Ozone Thickness (in Dobson units)',
     xlab='Years',
     type='o',
     main = 'Annual Ozone Thickness Change (1927-2016)')
```

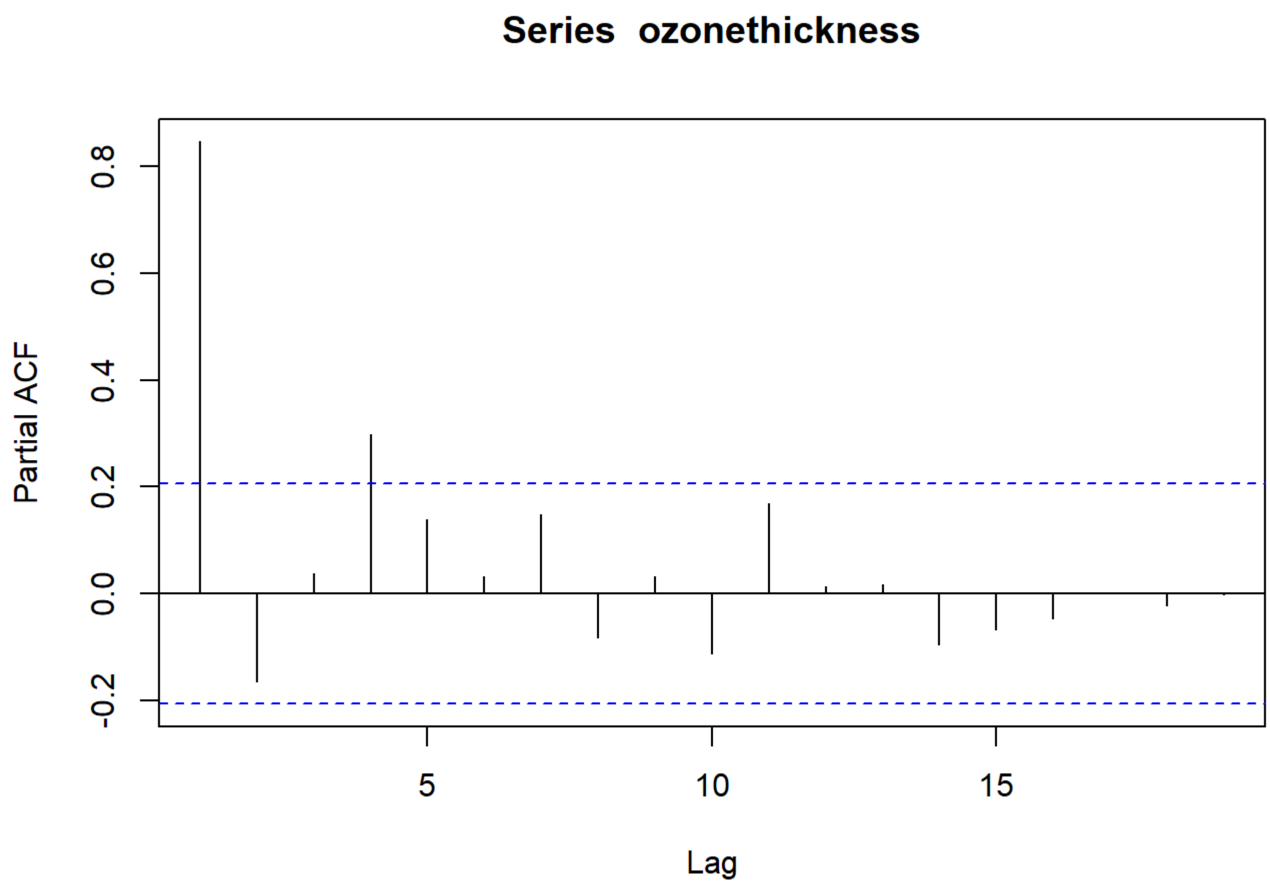


Plotting the Autocorrelation Function (ACF) and the Partial Autocorrelation Function PACF versus lag

```
acf(ozonethickness)
```




```
pacf(ozonethickness)
```



The ACF shows a gradual decay in the lags. The PACF has the first high lag value but all the values after that damp down. This is an evidance of non-stationary behaviour and the existance of trend. We will test the data further for the existance of non-stationary behaviour and try to make the time series data stationary.

Dickey-Fuller Unit Root Test

```
adf.test(ozonethickness)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ozonethickness
## Dickey-Fuller = -3.2376, Lag order = 4, p-value = 0.0867
## alternative hypothesis: stationary
```

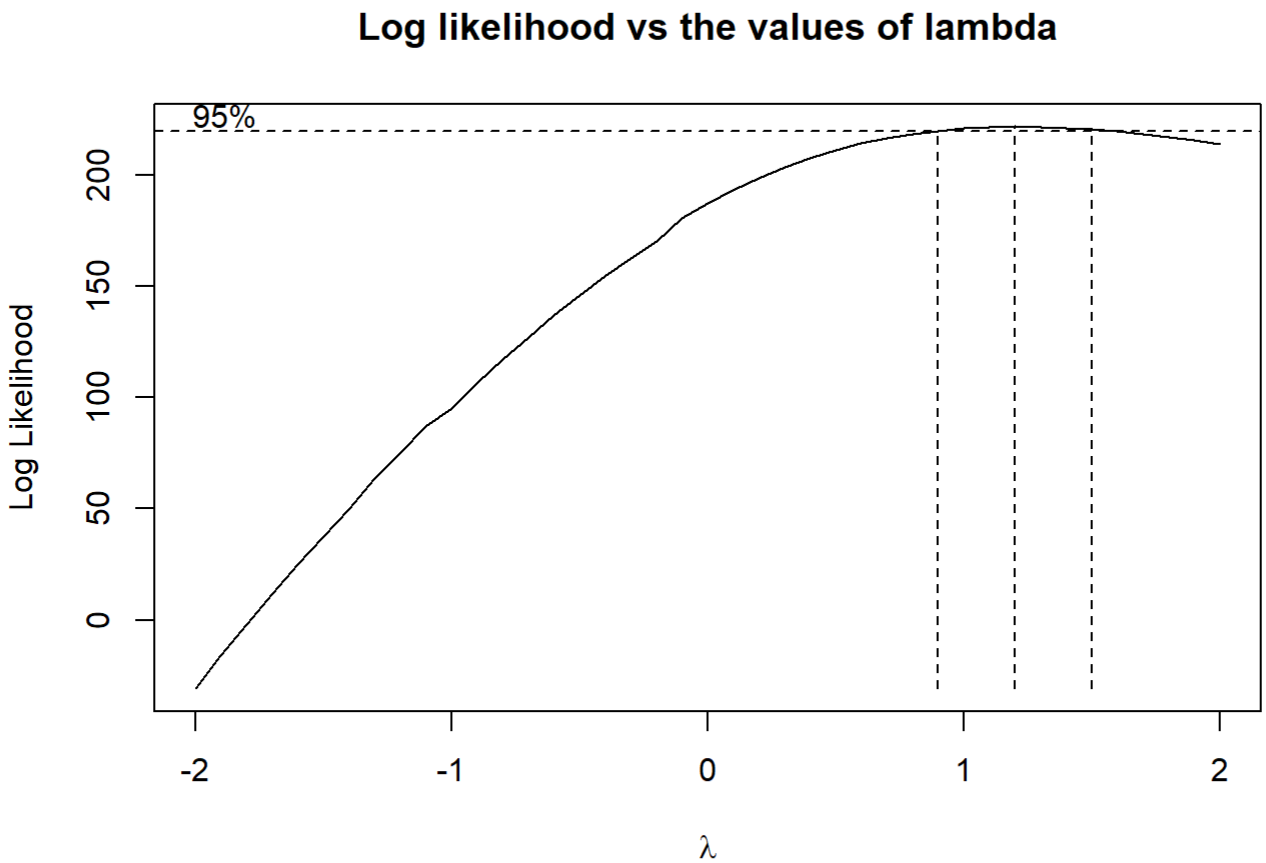
We can observe that the Dicky Fuller test has resulted in a p-value 0.0867 which is greater than 0.05. Hence we fail to reject the null hypothesis that the series is non-stationary.

We have to move forward to Box-Cox and Natural Logarithmic transformations to remove the variance from the data.

Box-Cox Transformation

We can transform the data into Box-Cox form. However, all the values need to be positive. We have transformed the data by using the following method.

```
#Add minimum value in the time series data and then add 1
ozonethickness_pos <- ozonethickness + abs(min(ozonethickness))+1
```



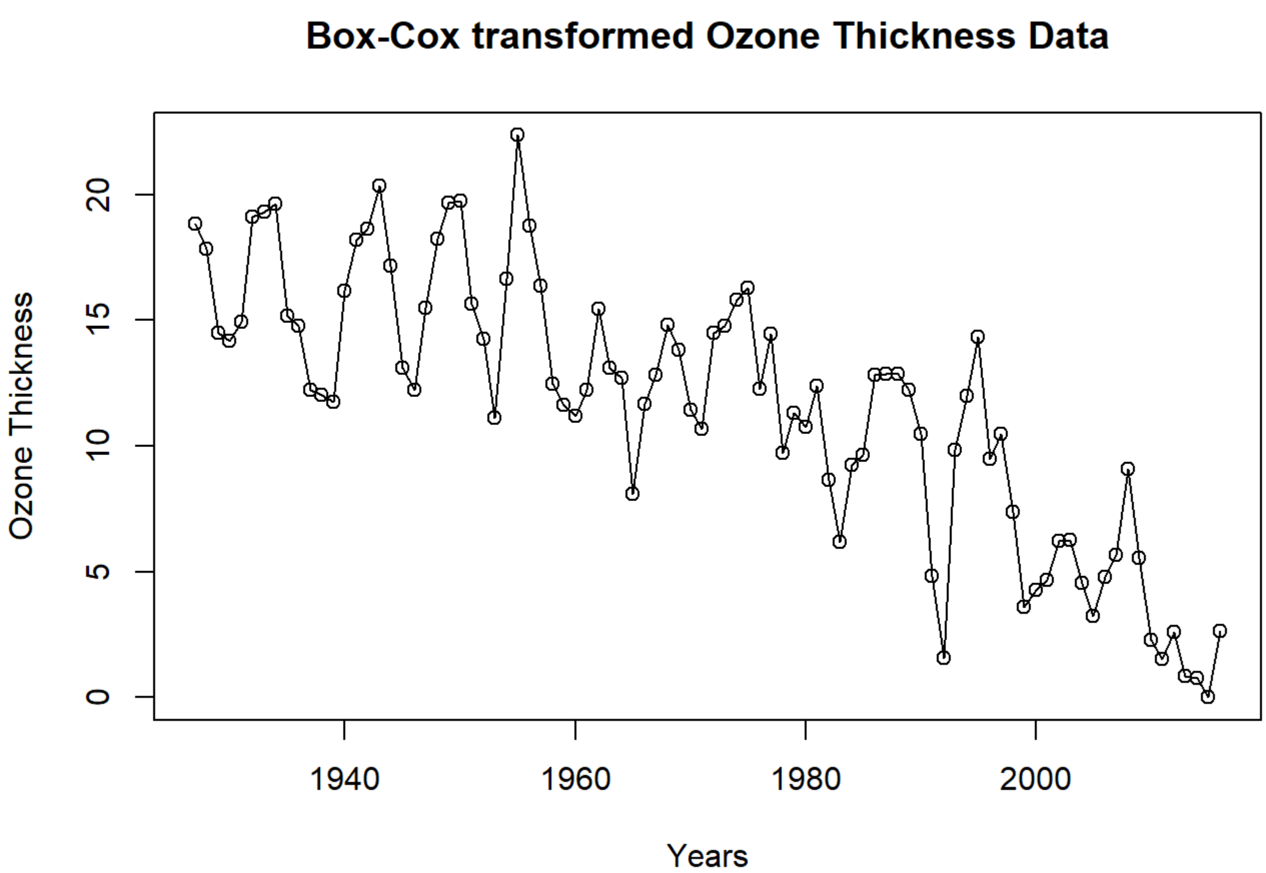
We will now check the confidence interval and determine the value of lambda.

```
#Checking the Confidence Interval
ozonethickness_bc_tf$ci
```

```
## [1] 0.9 1.5
```

The confidence interval is 0.9 to 1.5. We can select the intermittent value of lambda = 1.2. We will now transform the time series data using the Box-Cox Transformation with lambda = 1.2

```
#Transforming the Data
#Setting lambda value to 1.2
lambda = 1.2

# Create Box-Cox transformed data
ozonethickness_bc = (ozonethickness_pos^lambda-1)/lambda

#Plotting the Box-Cox Plot
plot(ozonethickness_bc,
     type='o',
     ylab='Ozone Thickness',
     xlab='Years',
     main='Box-Cox transformed Ozone Thickness Data')
```

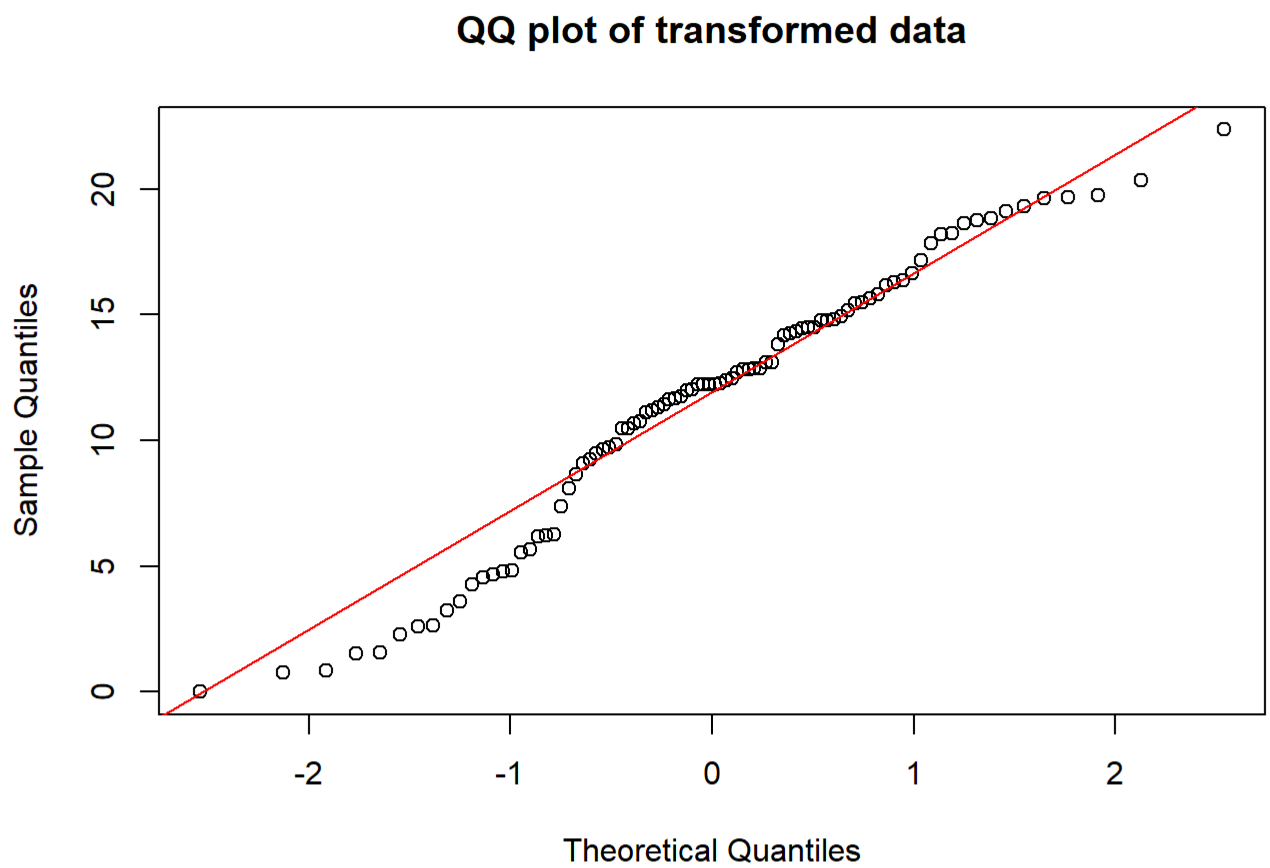


Test for Normality

We will test the Box-Cox Transformed data for normality by plotting the QQ Plot and using the Shapiro Test. We have created a function to check normality which plots the qq plot and performs the Shapiro-Wilkins test.

```
normality_test <- function(transform_used) {
  qq = qqnorm(transform_used, main = 'QQ plot of transformed data')
  qqline(transform_used, col = 2)
  sha = shapiro.test(transform_used)
  return(sha)
}
```

```
normality_test(ozonethickness_bc)
```

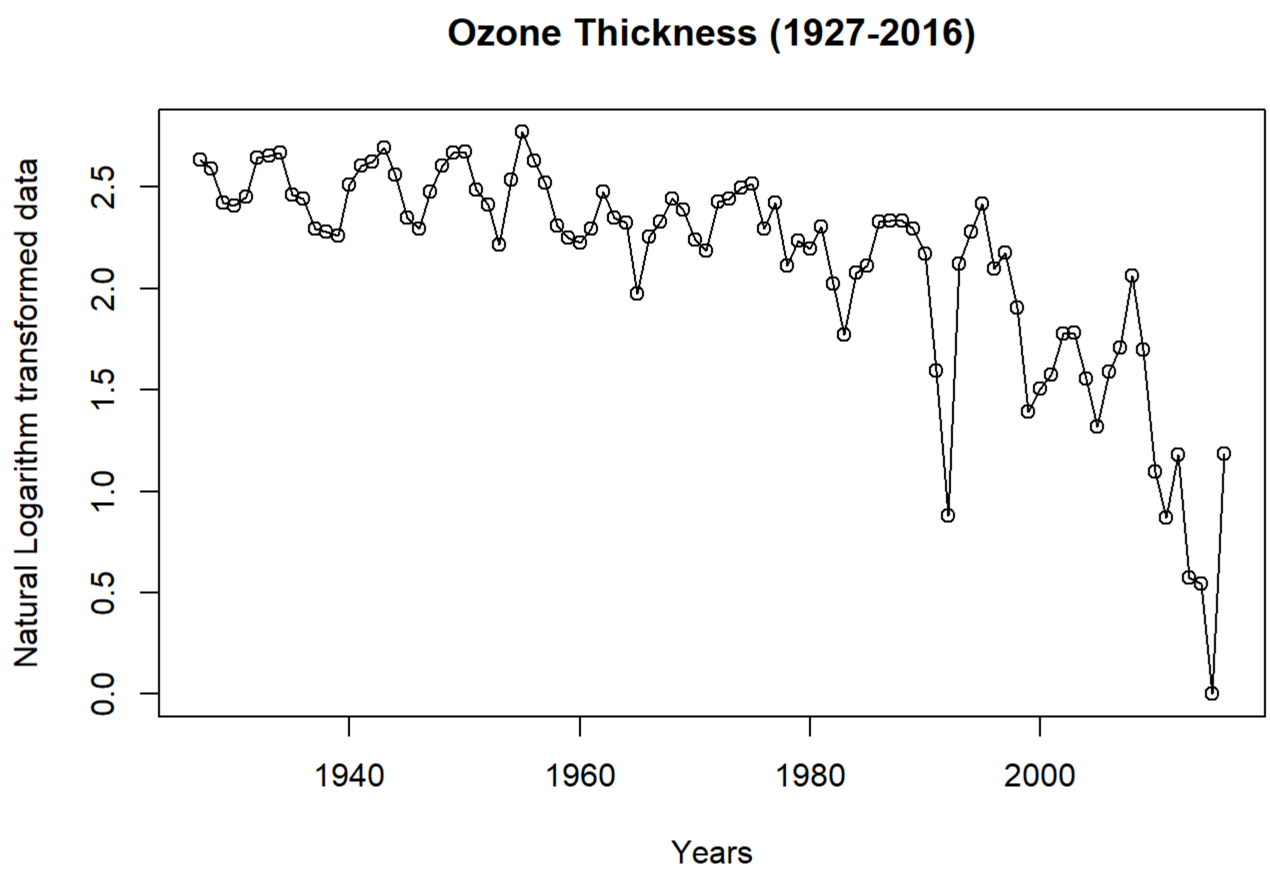



```
##
##  Shapiro-Wilk normality test
##
## data:  transform_used
## W = 0.96644, p-value = 0.01995
```

Conclusion: The qq plot is not normally distributed. The p-value for the Shapiro-Wilkins testis 0.01995, which is less than 0.05. Hence, we reject the null hypothesis of normal distribution. Thus the Box-Cox Transformation was unsucessful to attain normality. We will try the Natural Logarithmic Transformation.

Natural Logarithmic Transformation

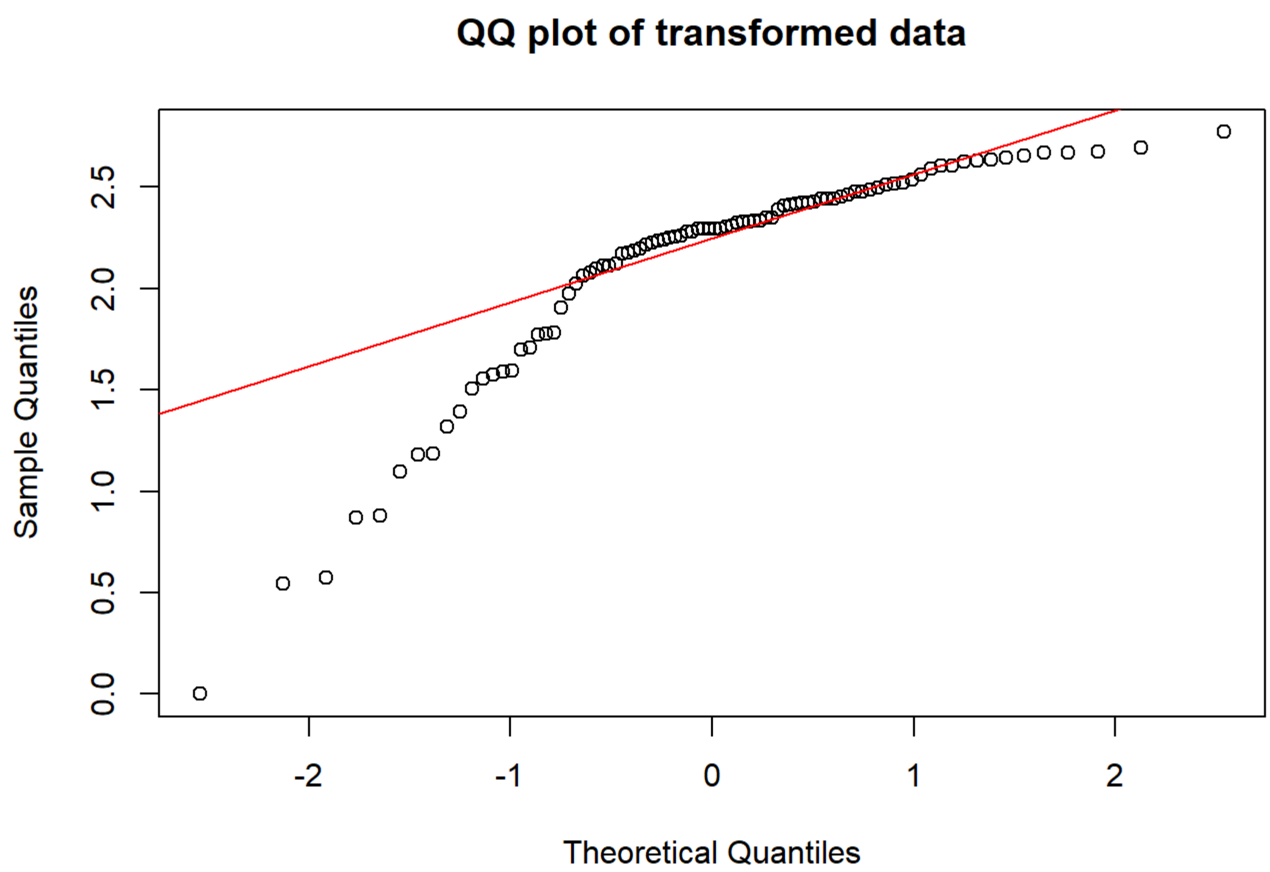
```
#Creating Logarithm transformed data
ozonethickness_log <- log(ozonethickness_pos)

#Plotting the Logarithm transformed data
plot(ozonethickness_log,
     ylab='Natural Logarithm transformed data',
     xlab="Years",
     type='o',
     main = "Ozone Thickness (1927-2016)")
```



Test for Normality

We have used the function created above to check for normality

```
normality_test(ozonethickness_log)
```



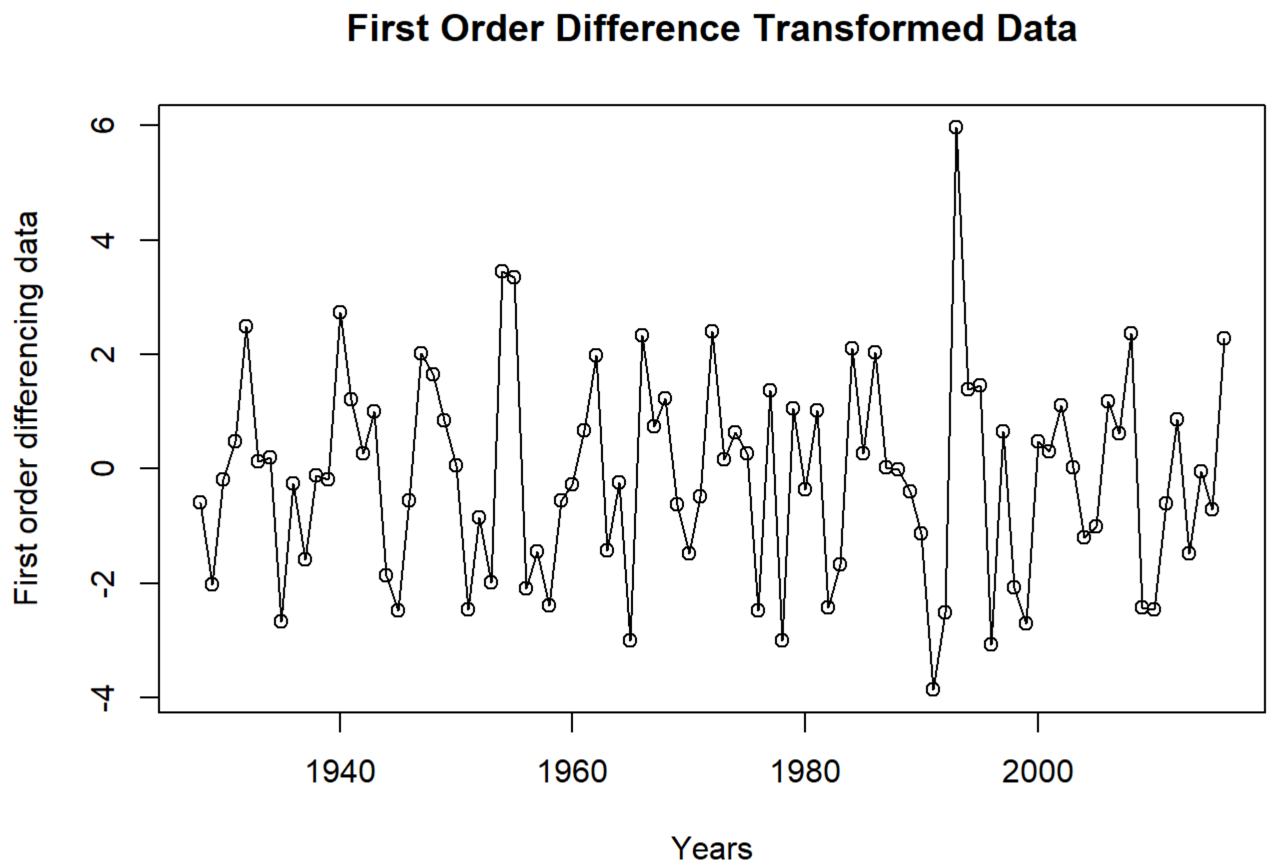

```
##
##  Shapiro-Wilk normality test
##
## data:  transform_used
## W = 0.81905, p-value = 3.949e-09
```

Conclusion: The qq plot is not normally distributed. The p-value for the Shapiro-Wilkins testis 3.949e-09, which is less than 0.05. Hence, we reject the null hypothesis of normal distribution. Thus, the Natural Logarithmic Transformation was unsucessful to attain normality.

We will now apply the first difference transformation to the time series data.

First Order Differencing of the Time Series Data

```
#First Differencing of the Time Series Data
ozonethickness_diff = diff(ozonethickness)

#Plotting the data
plot(ozonethickness_diff,
     type='o',
     ylab='First order differencing data',
     xlab='Years',
     main='First Order Difference Transformed Data')
```



Dicky Fuller Test

```
adf.test(ozonethickness_diff)
```

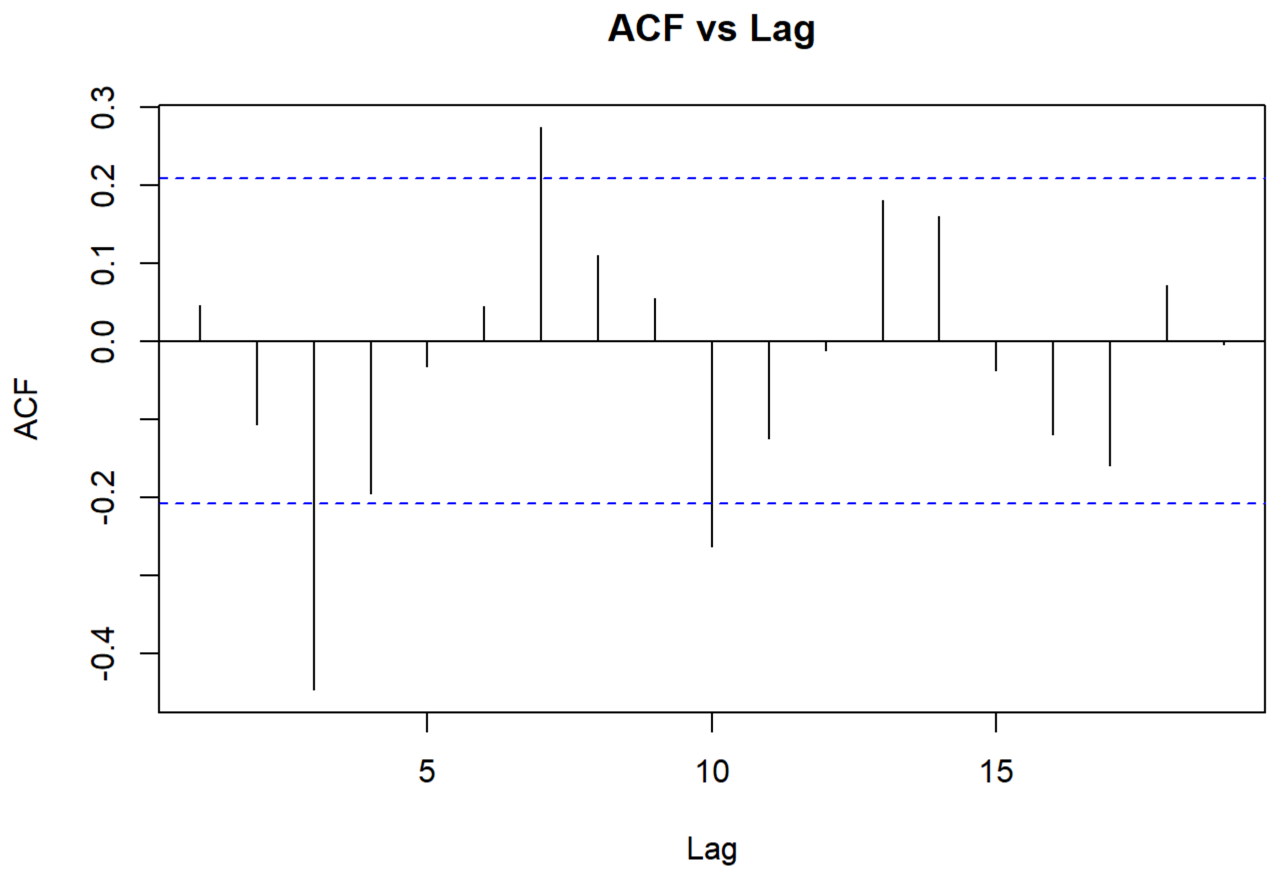
```
## Warning in adf.test(ozonethickness_diff): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ozonethickness_diff
## Dickey-Fuller = -7.1568, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

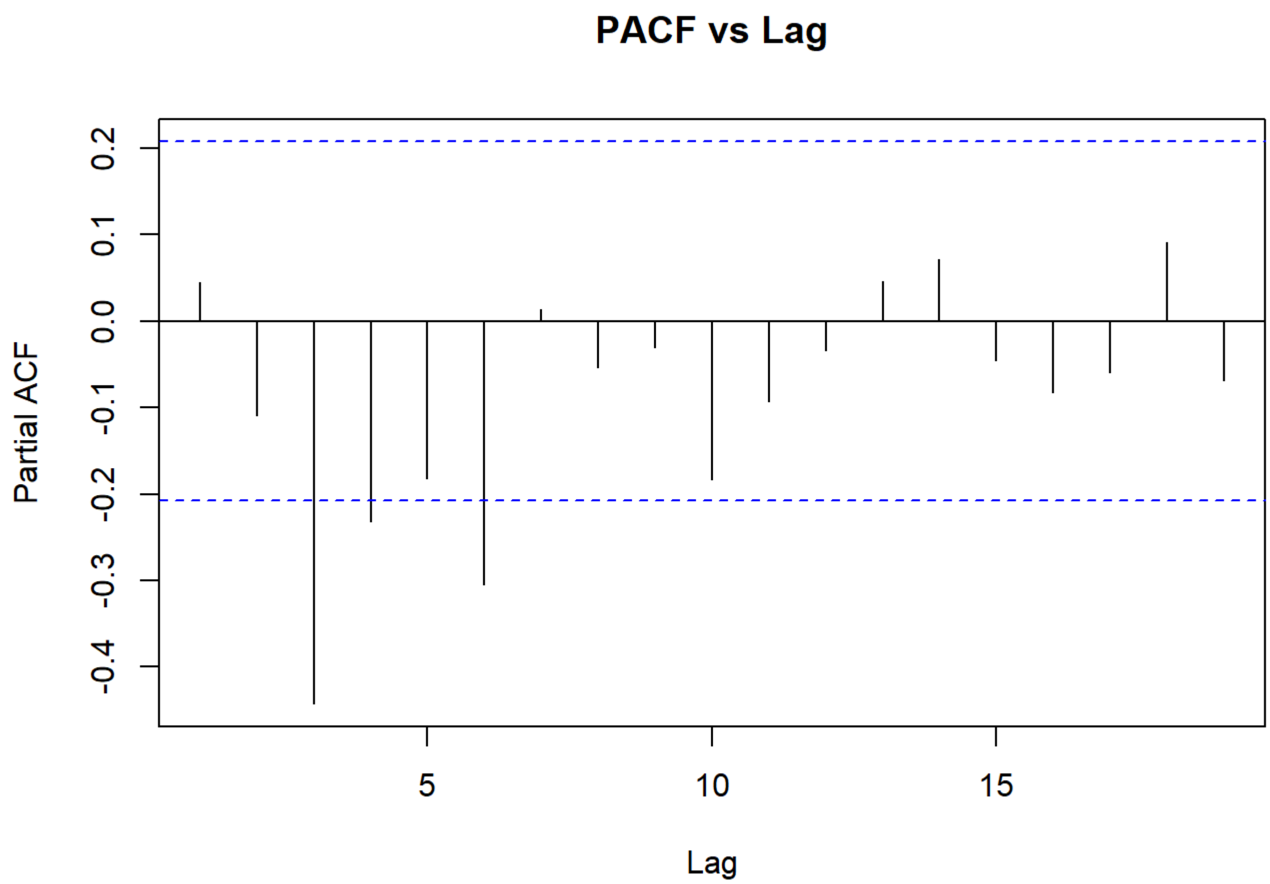
The Dicky Fuller test gives a p-value smaller than 0.05, hence we can reject the null hypothesis that the series is non-stationary.

ACF and PACF of First Difference Transform

```
acf(ozonethickness_diff, main='ACF vs Lag')
```




```
pacf(ozonethickness_diff, main='PACF vs Lag')
```



We can select the value p = 3 from the PACF and the value of q = 3 from ACF. The value of d will be 1 as we have considered the first order difference. Therefore, the possible model is ARIMA(3,1,3).

It can also be said that the models ARMIA(3,1,2), ARMIA(3,1,1), ARMIA(3,1,0), ARMIA(2,1,3), ARMIA(2,1,2), ARMIA(2,1,1), ARMIA(2,1,0), ARMIA(1,1,3), ARMIA(1,1,2), ARMIA(1,1,1), ARMIA(1,1,0), ARMIA(0,1,3), ARMIA(0,1,2), ARMIA(0,1,1) are possible because they are smaller than ARMIA(3,1,3). we can select some of these models only after analyzing them further.

Extended Autocorellation Function

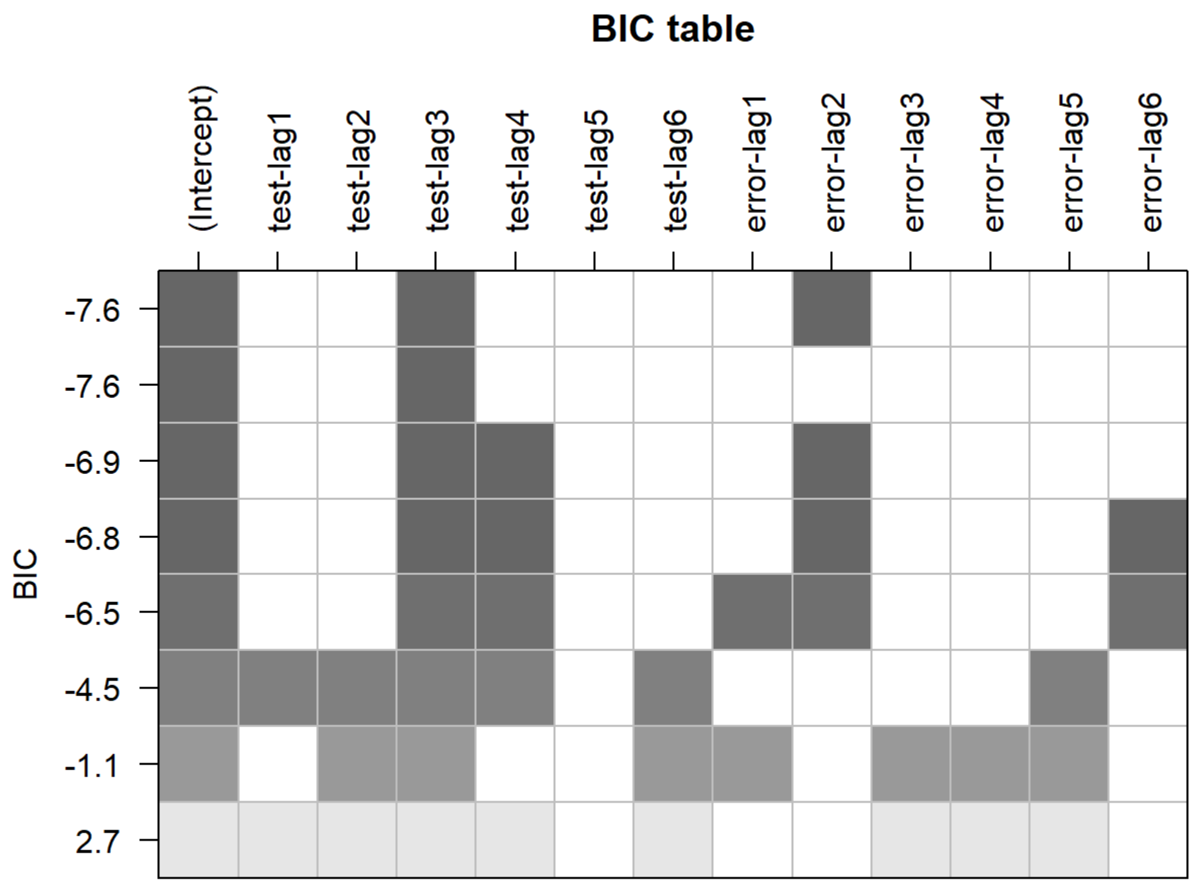
```
#Creating the Extended Autocorrelation Function (EACF) model
eacf(ozonethickness_diff)
```

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 o o x o o o x o o x o  o  o  o
## 1 x o x o o o o o o x o  o  o  o
## 2 x o x o o o x o o x o  o  o  o
## 3 x o x o o x o o o o o  o  o  o
## 4 x o o x o x o o o o o  o  o  o
## 5 x x x x o x o o o o o  o  o  o
## 6 o o o x x o o o o o o  o  o  o
## 7 o o o x o o o o o o o  o  o  o
```

Here, the value of p = 0,1 and the value of q = 0,1,3,4. The possible models are ARIMA(0,1,1), ARIMA(0,1,3), ARIMA(0,1,4), ARIMA(1,1,0), ARIMA(1,1,3) and ARIMA(1,1,4)

Bayesian Information Criterion

```
#Plotting the BIC table
plot(armasubsets(y=ozonethickness_diff, nar=6, nma=6, y.name='test', ar.method='ols'))
title(main = 'BIC table', line= 6)
```



The values that can be attained by using the BIC table are p = 3 and q = 2 Hence the possible models are ARIMA(3,1,2) and ARMIA(3,1,0).

Conclusion

- Task 1:- The best fitted model is the Quadratic Model. The quadratic model was used to forecast the values for the next 5 years. The next five years showed a continuing downward trend in the thickness of the ozone layer.
- Task 2:- The final set of models that can be selected are ARIMA(3,1,3),ARIMA(0,1,1), ARIMA(0,1,3), ARIMA(0,1,4), ARIMA(1,1,0), ARIMA(1,1,3), ARIMA(1,1,4), ARIMA(3,1,2) and ARMIA(3,1,0).

References

- Time Series Analysis Notes by Dr. Hayder Demirhan

- Bhalla, D., n.d. Regression : Transform Negative Values. [online] ListenData. Available at: https://www.listendata.com/2015/09/regression-transform-negative-values.html (https://www.listendata.com/2015/09/regression-transform-negative-values.html) [Accessed 16 April 2021].
- Boostedml. 2020. Stationarity and Non-stationary Time Series with Applications in R - Boostedml. [online] Available at: https://boostedml.com/2020/05/stationarity-and-non-stationary-time-series-with-applications-in-r.html#Background (https://boostedml.com/2020/05/stationarity-and-non-stationary-time-series-with-applications-in-r.html#Background) [Accessed 16 April 2021].