University of California, Santa Barbara

Time Series Analysis:
Monthly Traffic Fatalities in Ontario (1960 - 1974)

PSTAT 174: Spring 2018

Authors: Blake Shaw
Pranavi Gandham
Lizeth Muñoz
Jae Yun Lee
Manny Corrales-Ibarra

Professor: Sudeep Bapat

June 6, 2018

Contents

1 Introduction 2

2 Data Exploratory Analysis 3

2.1 Time Series Plot 3

3 Data Transformation 4

3.1 Box-cox Transformation 4

3.2 Removing Seasonality and Trend 5

4 Model Building and Selection 6

4.1 Preliminary Identification 6

4.2 Model Selection through Criterion 6

4.3 Model Estimation 7

5 Diagnostics 7

5.1 Checks for normality 8

5.2 Checks for serial correlation 8

5.3 Normality Check 9

6 Forecasting 9

7 Conclusions 11

References

Appendices

Abstract

It is evident that the number of traffic fatalities vary from season to season due to changing weather, frequency of car usage, etc. So in this project we evaluate the trend of monthly traffic fatalities in Ontario, Canada by looking at past data, specifically between the years of 1960 and 1974 in order to forecast future occurrences. By implementing various diagnostics steps and through plotting the ACF and PACF models, using the Box-cox transformation method, and differencing the data to remove seasonality and trend we find the desired SARIMA model.

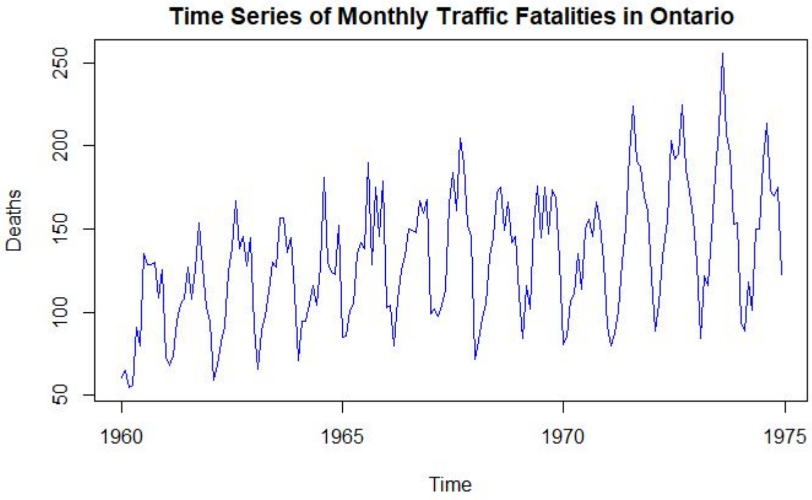
1 Introduction

In 2016, traffic fatalities in Ontario averaged about 36.5 a month coming out to 439 a year. Back in just August of 1973 the amount of traffic fatalities equaled 256. What has changed over the past 50 years? In our report, we used data that recorded the monthly traffic fatalities in Ontario from January 1960 to December 1974 from datamarket. This data includes 180 observations; one for each month of the year for 15 years. We want to use data that is out of date so we can see, through the best possible model we create in R statistical software, if we can predict the monthly fatalities of Ontario a few months after to see if our model is accurate and if not, what may have caused the discrepancy. Through exploratory analysis we conclude that the variance is nonstationary and that there is trend and seasonality. We continue on to use a square root transformation to stabilize the variance and to detrend and deseasonalize the time series data via differencing. After we decided on the square root transformation differenced at lag 12, we moved on to model selection. We observed the ACF and PACF plots of the series and identify three preliminary models and after using AICc criteria narrow down our model to ARIMA(1,0,1)x$(2,0,2)_{12}$. We then performed diagnostics such as checks for normality, independence and constant variance. Finally, we forecasted our model to predict 12 future observations, or one year's worth of monthly observations.

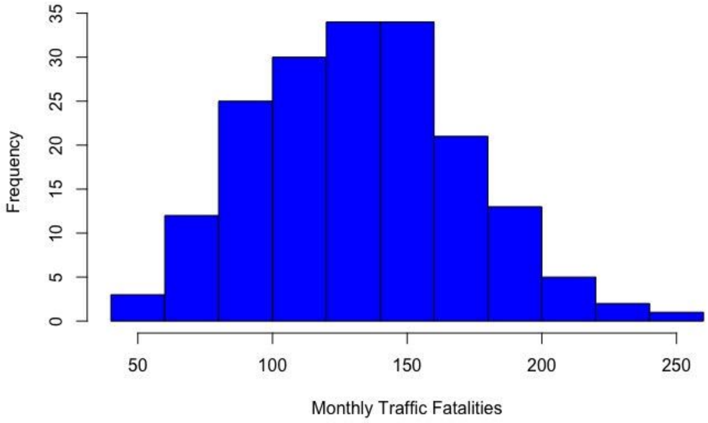
2 Data Exploratory Analysis

2.1 Time Series Plot

The data includes 180 observations; one for each month for fifteen years. In the time series plot of the data points we could tell that there was definitely seasonality and a possibly increasing trend based on the periodic peaks and length of said peaks. We noticed that the fatalities were at their highest in late summer to early fall (July-October) and then decreased to the lowest in spring (January-March). To see if the seasonality truly does exist we created a matrix of the monthly accidents. The matrix narrows down the month range and shows that August and September tend to have the highest fatalities while January and February tend to have the lowest fatalities. Thus confirming a seasonal component to the time series.



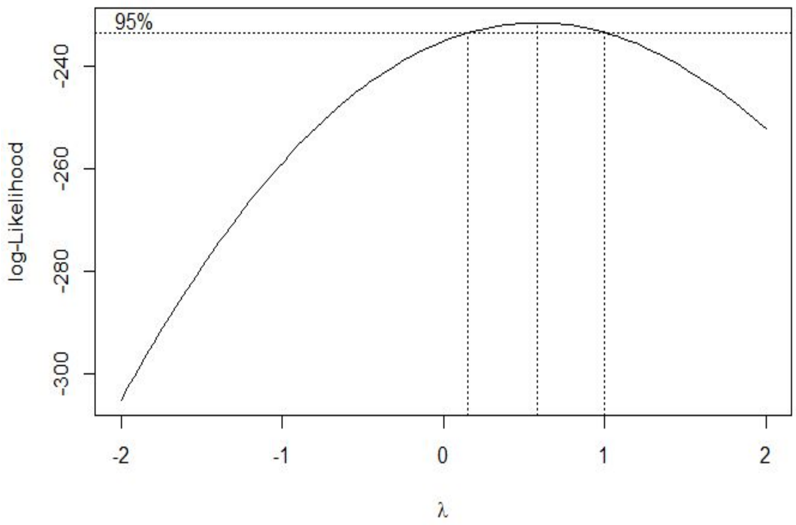
To see if there was a trend we calculated a sample mean for each year. Each year the means increases lead us to believe that there indeed is a trend in the time series. We also calculated the variance for each year and saw that as each year increased the variance increase making it nonstationary. We can also see that the histogram is slightly skewed to the



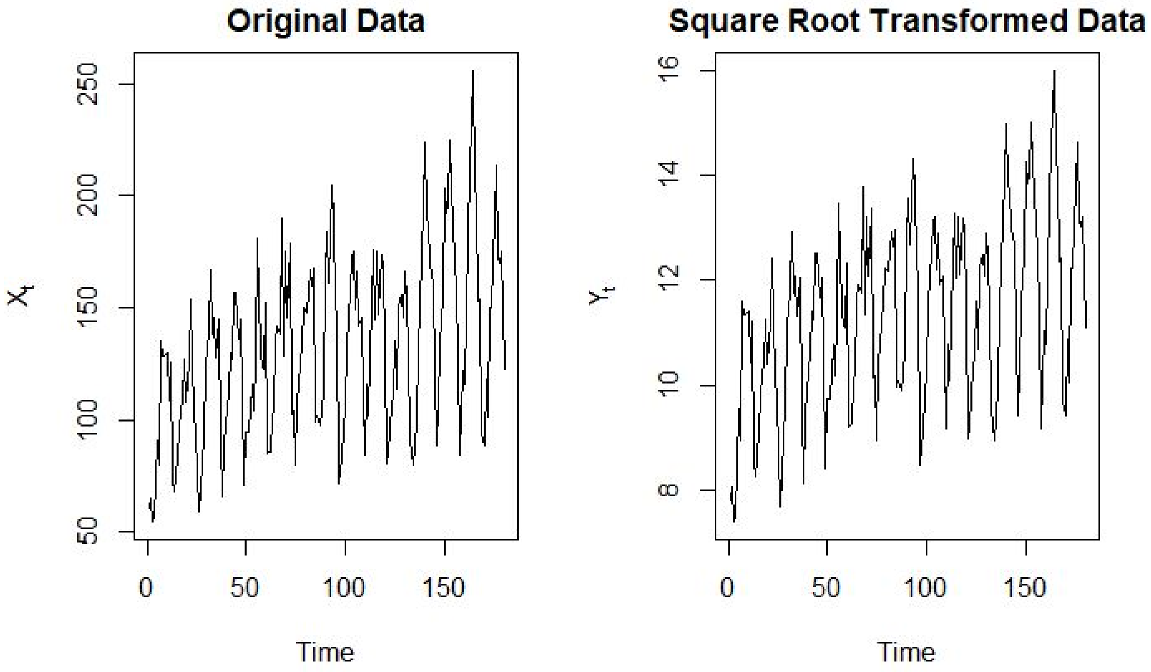
left indicating unequal variances. Therefore we needed to modify the data before we could begin building the model.

3 Data Transformation

3.1 Box-cox Transformation



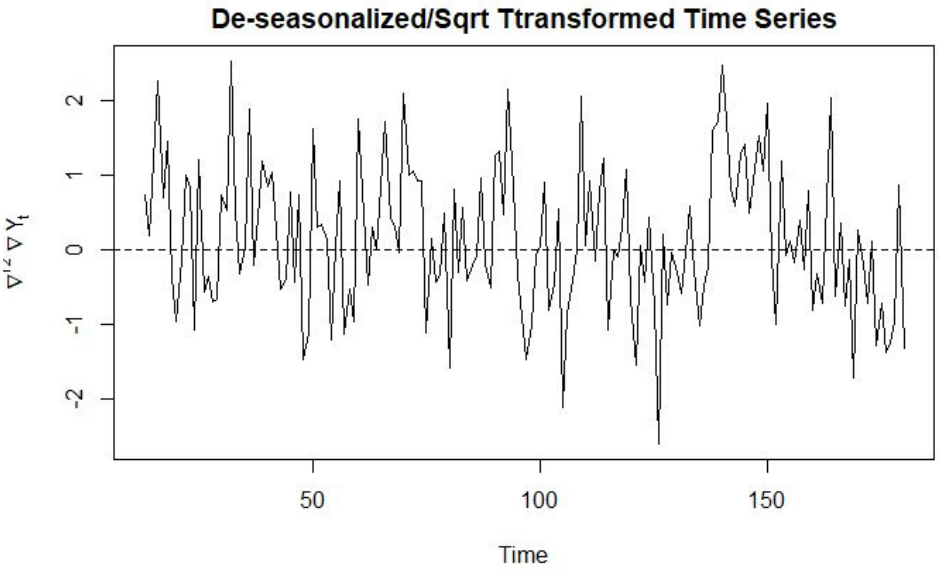
To stabilize the variance we used a box-cox transformation on the original data. Our lower and upper bounds for lambda were 0.181 and 0.989, respectively, and our estimate was 0.585. Zero is not in the confidence interval, so we could not apply a log transform on our time series. The estimate for lambda is very close to 0.5, so we chose to use square root as the transformation for time series. The variance we do get after the square root transformation is 2.94.

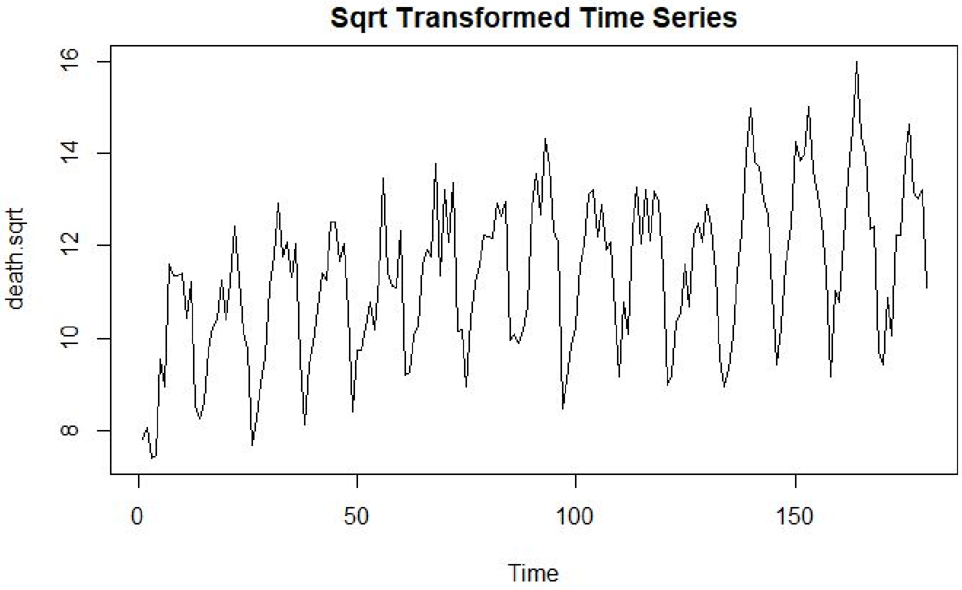


3.2 Removing Seasonality and Trend

In order to remove possible seasonality and trend, we must difference the data and find the lag d that will make the model stationary.

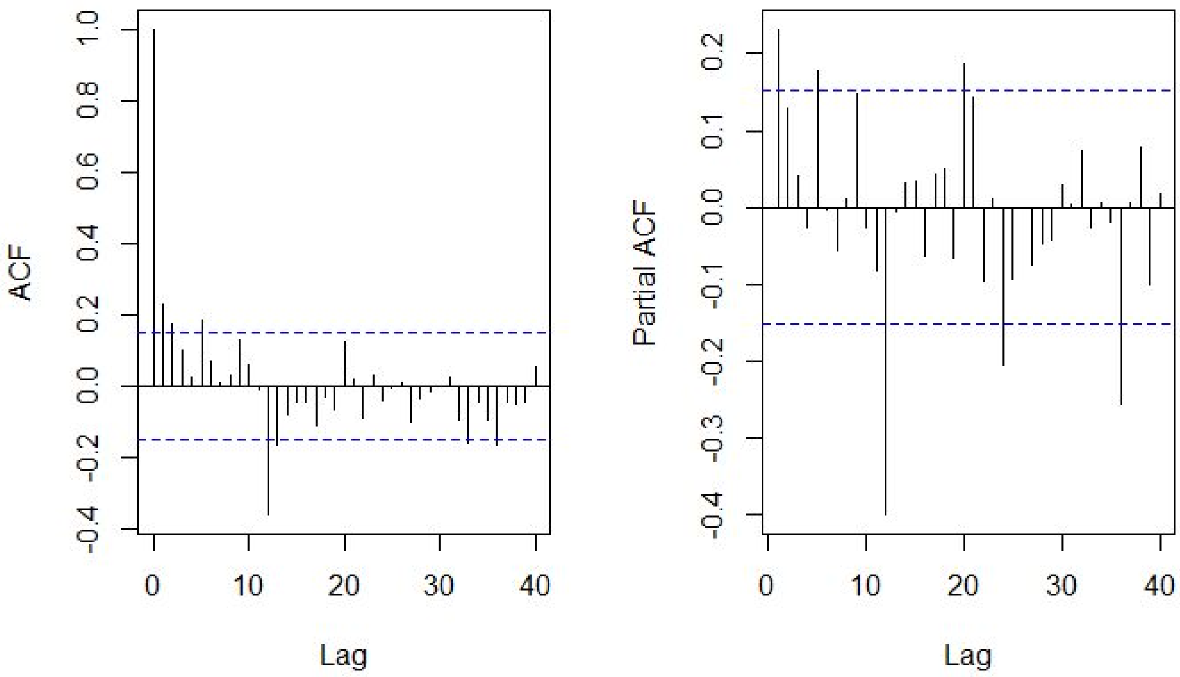
First we difference the data at lag 12 to remove seasonality. This reduced variance to 0.934. After attempting to apply another difference at lag one, this increased the variance to 1.43. This means that the difference taken to remove seasonality either removed the partial trend or there was not a trend at all. Also, after performing an Augmented Dickey–Fuller Test, we calculated a p-value of 0.01. Thus we can reject the null with a significance level of 0.05 that the time series is non-stationary. Therefore, the model is stationary.


De-seasonalized/Sqrt Ttransformed Time Series


Sqrt Transformed Time Series

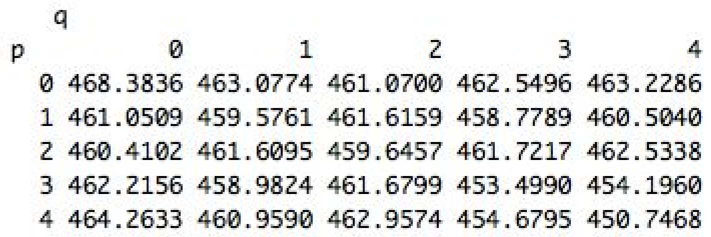
4 Model Building and Selection

4.1 Preliminary Identification



When we look at the initial ACF/PACF plots we can that there are significant spikes at 0 and 1, however we can also see that there are spikes that happen in the PACF plot around 2, 3 and even 4. Therefore we use 0 to 4 as parameters for both p and q.

4.2 Model Selection Through Criterion



```
    q
p          0        1        2        3        4
  0 468.3836 463.0774 461.0700 462.5496 463.2286
  1 461.0509 459.5761 461.6159 458.7789 460.5040
  2 460.4102 461.6095 459.6457 461.7217 462.5338
  3 462.2156 458.9824 461.6799 453.4990 454.1960
  4 464.2633 460.9590 462.9574 454.6795 450.7468
```

After considering our p and q values we evaluate the AICc values of the models. We used the square root lag 12 differenced box-cox transformation in our ARIMA model and ran a for nested loop for the values of p and q. In our matrix we see that the three lowest AICc values are at ARMA(1,1), ARMA(3,1) and ARMA(2,2).

We choose model ARMA(1,1) even though ARMA(3,1) gives us a smaller AICc the ACF plot does not show a significant spike at 3 and also because the principle of parsimony states that we should choose the model with the fewest parameters. When we run an auto.arima function in R, we calculated that the lowest AICc model is ARIMA(1,0,1)x$(2,0,2)_{12}$ which confirms our conclusions. Also, a yule-walker estimate estimated that there were 12 parameters for the autoregressive portion of the model, but that would overfit our model.

4.3 Model Estimation

In order to select a model, we had to estimate our coefficients. Our calculated coefficients are: $\theta_1 = -0.5774$, $\Phi_1 = -0.7625$, and $\mu = 0.1572$, where $\theta(B) = 1 - 0.5774B$, and $\Phi(B) = 1 - 0.7625B$. Since our roots are less than 1 and they lie outside of the unit circle, we determined our model to be stationary and invertible. We noticed that for both the seasonal and non-seasonal part of the model, the order is less than 1 so because the magnitude of the coefficients are all less than 1, our model is causal and invertible. The formula for our final model is:

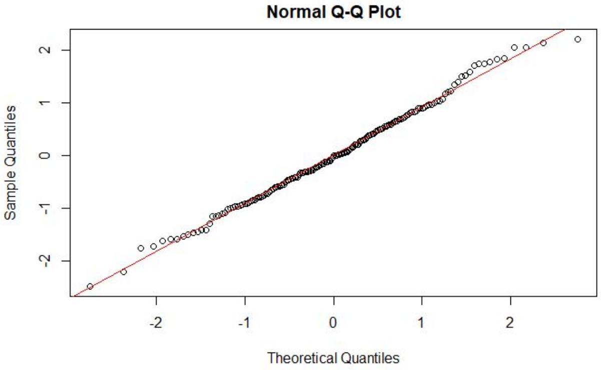
$$X_t = 0.7625X_{t-1} - 0.5774Z_{t-1} + Z_t + 0.1572 \text{ where } Z_t \sim WN(0,\sigma^2)$$

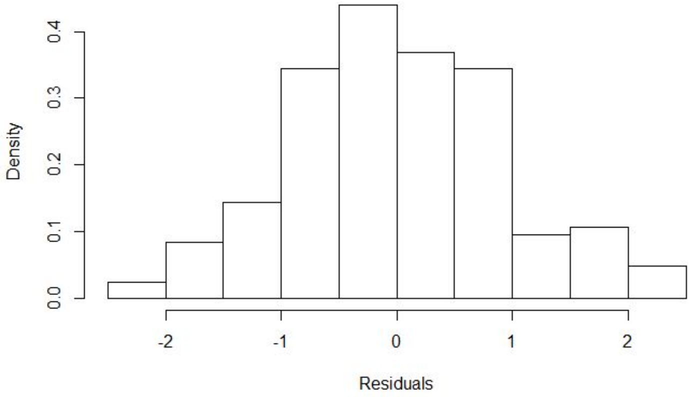
5 Diagnostics

After selecting a time series model and estimating parameters, we need to confirm that the assumptions that we put on on our model are valid. This is necessary because we cannot forecast accurate results if the assumptions we put on the model are not true. The three assumptions to examine are normality of residuals, independence of residuals, and heteroscedasticity.

5.1 Checks for Normality

We can see that our residuals are relatively normal when plotting sample and theoretical quantiles against one another. Most of our values fall in a straight line at 45°, which implies normality.





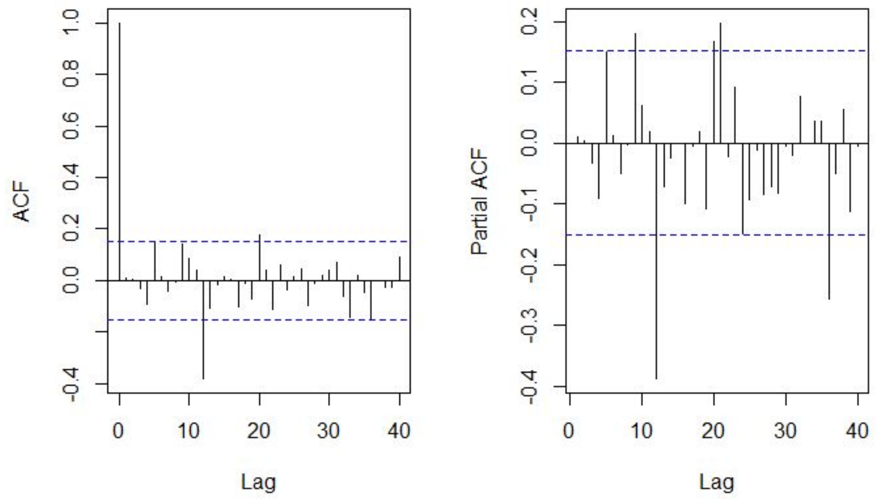
The histogram above shows that our residuals roughly follow a gaussian distribution. The mean of our residuals is close to zero, at -0.00204, while our variance is close to one at 0.85559. Also, a shapiro-wilk test gives us a p-value of 0.6922, which means we fail to reject null that the residuals do not follow a normal distribution at a significance level of 0.05.

5.2 Checks for Independence

We conducted a Ljung-Box and Box-Pierce to test test the hypothesis that there is no serial correlation at the significance level of 0.05. Both tests failed to reject the null, thus we cannot prove that there is serial correlation.

Test	Test Statistic	P-value
Ljung-Box	11.083	0.2701
Box- Pierre	10.499	0.3116

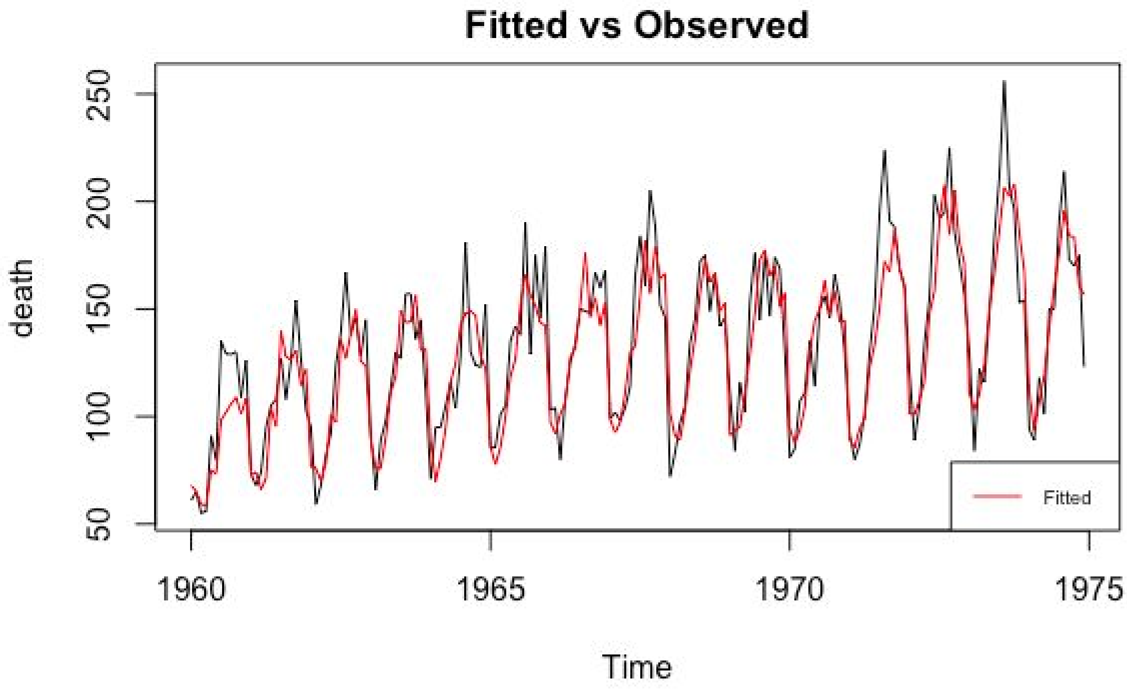
5.3 Constant Variance Checking



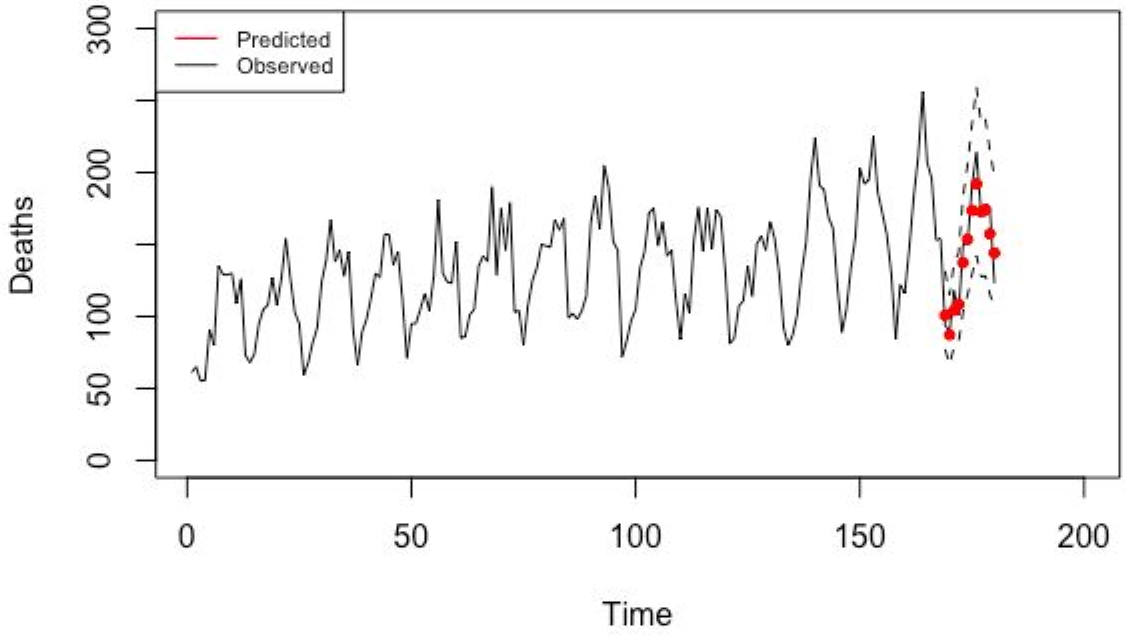
By looking at the ACF and PACF plots of the residuals, we can check to see if the variance of the model is consistent over time. If variance is erratic, it would be difficult to predict future values using our model. Visually, it appears that the model does not have a heteroskedasticity problem due to most of the lags being within the 95% confidence interval.

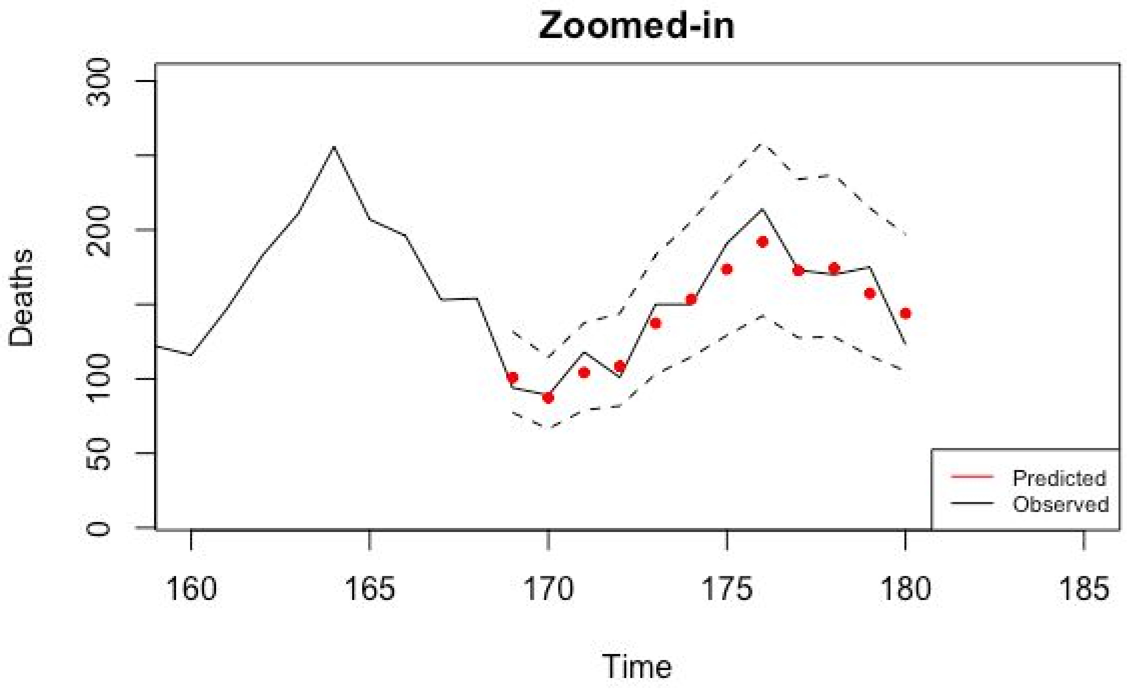
6 Forecasting

The forecast was used to predict 12 additional monthly observations for the year of 1975. The predicted values and their 95% confidence regions are shown on the plot above. The pattern of seasonality and trend are comparable to the original values. It can be seen that the confidence intervals for further predictions from the original data are increasing. This means that we have less and less certainty as we stray from our original data. Our predicted values are slightly smaller than the observed values. Furthermore, from the plot, our predicted values are within a 95% confidence band.



Forecast/Prediction Plot





Dotted Line represents the 95% Confidence interval (value 1.96) and the red dots represent the 12 future observations.

7 Conclusion

Our goal for this project was to build a time series model that best explained monthly fatalities in Ontario, Canada from 1960 to 1974 and best predicted monthly fatalities of the next year of 1975 with our model. In summary, we used a square root transformation to stabilize variance, took the difference at lag 12 and at lag 1 to remove seasonality and trend, respectively. We also identified models using ACF/PACF plots, compared models using AICc to select the a model, conducted diagnostic checks to verify the validity of our model, and then forecasted it to make predictions over one year. This lead to our final model:

$$\text{ARIMA}(1,0,1)\text{x}(2,0,2)_{12}$$

with coefficients $\theta_1 = -0.5774$, $\Phi_1 = -0.7625$, and $\mu = 0.1572$.

We then predict the future fatalities in the next year of 1975 using 12 total values based on a 95% confidence interval. Our values are similar and fall into this confidence band. We see that it stays fairly close to what we have observed which supports the accuracy of our model.

References

[1] *DataMarket: Monthly Traffic Fatalities in Ontario 1960 - 1974* https://datamarket.com/data/set/22ty/monthly-traffic-fatalities-in-ontario-1960-1974#!ds=22ty&display=line

RStudio Statistical Software

Appendix

Libraries used

```
library(qpcR)

library(readxl)

library(forecast)

library(MASS)

library(readr)

library('ggplot2')

library(astsa)

library(tseries)
```

Importing code

```
ontario_excel <- read_excel("ontario excel.xlsx")

death <- ontario_excel

death <- ts(death)

# creating initial time series

plot(death, ylab = "Deaths", main = "Time Series of Monthly Traffic
Fatalities in Ontario", col = "blue")

 # Exploratory Analysis

# Creating matrix of monthly accidents with years as rows and columns
as months


year = matrix(NA, nrow = 15, ncol = 12)

rownames(year) <- paste(1960:1974)

colnames(year) <- paste(c('Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun',
'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'))
```

```
for (i in 0:14) {

  year[i+1,] <- death[((i*12)+1):(12*(i+1))]

}


year

# August and September tend to be the higher death months while
January and February tend to be the lower death months.  This shows
that there is a seasonal component to the time series.
```

Calculating sample mean for each year

```
# Calculating sample mean for each year

mean.values = c(rep(0, 15))

for (i in 1:15) {

  mean.values[i] <- mean(year[i,])

}

mean.values

# There seems to be an increasing mean as each year increases.  This
leads us to believe that there is a trend in the time series.
```

Calculating sample variance for each year

```
var.values = c(rep(0, 15))

for (i in 1:15) {

  var.values[i] <- var(year[i,])

}

var.values

# There seems to be an increasing variance as each year increases.
```

Box-Cox Transformation

```
bcTransform <- boxcox(death~ as.numeric(1:length(death)))

lambda.estimate <- bcTransform$x[which(bcTransform$y ==
max(bcTransform$y))]



lambda.estimate.values <- bcTransform$x[bcTransform$y >
max(bcTransform$y) - 1/2 * qchisq(.95,1)] # Values above 95% CI line

lambda.lower.bound <- min(lambda.estimate.values)

lambda.upper.bound <- max(lambda.estimate.values)


lambda.estimate

## [1] 0.5858586

lambda.lower.bound

## [1] 0.1818182

lambda.upper.bound

## [1] 0.989899
```

Square Root Transformation

```
death.sqrt <- sqrt(death)

var(death.sqrt) #variance of transformed time series

##         deaths

## deaths 2.943117
```

ACF and PACF of Square Root Transformed Data

```
op <- par(mfrow = c(1,2))

ts.plot(death, main = "Original Data", ylab = expression(X[t]))
ts.plot(death.sqrt, main = "Square Root Transformed Data", ylab =
expression(Y[t]))


par(op)
```

Diffencing

```
death.lag12 <- diff(death.sqrt, lag = 12)


death.lagged <- diff(death.lag12, lag = 1)

var(death.lagged)

##         deaths

## deaths 1.428867

var(death.lag12)  # differenced at lag 12

##         deaths

## deaths 0.9343891

var(death.lagged) # then differenced at lag 1

##         deaths

## deaths 1.428867

# therefore not necessary to difference at lag 1 to remove trend
because it increases variance
```

Dickey-Fuller Test

```
adf.test(death.lag12)

##

##  Augmented Dickey-Fuller Test

##

## data:  death.lag12
## Dickey-Fuller = -3.9686, Lag order = 5, p-value = 0.01241

## alternative hypothesis: stationary
```

Time Series of Differenced and transformed series and Times Sereis with just tranformation

```
ts.plot(death.lag12, main = "De-seasonalized/Sqrt Ttransformed Time Series", ylab = expression(nabla^{12}~nabla~Y[t]))

abline(h=0, lty = 2)


ts.plot(death.sqrt, main = "Sqrt Transformed Time Series")
```

ACF/PACF of transformed and differenced time series

```
# re-calculate the sample variance and examine the ACF and PACF

op = par(mfrow = c(1,2))

acf(death.lag12, lag.max = 40, main = "")

pacf(death.lag12, lag.max = 40, main = "")

title("", line = -1, outer=TRUE)


par(op)
```

AICc chart of possible models

```
aiccs <- matrix(0,nrow = 5,ncol = 5)

dimnames(aiccs) <- list(p = 0:4, q = 0:4)

for (i in 0:4){

6

for (j in 0:4){
```

```
aiccs[i+1,j+1] <- AICc(arima(death.lag12,order = c(i,0,j),method =
"ML"))
}
}
aiccs
##    q
## p          0       1    2        3       4
##   0 468.3836 463.0774 461.0700 462.5496 463.2286
##   1 461.0509 459.5761 461.6159 458.7789 460.5040
##   2 460.4102 461.6095 459.6457 461.7217 462.5338
##   3 462.2156 458.9824 461.6799 453.4990 454.1960
##   4 464.2633 460.9590 462.9574 454.6795 450.7468
```

Parameters of desired model type arima(1,0,1)

```
fit <- Arima(death.lag12, order=c(1,0,1))
fit
## Series: death.lag12
## ARIMA(1,0,1) with non-zero mean
##
## Coefficients:
##          ar1       ma1    mean
##       0.7625  -0.5774  0.1572
## s.e.  0.1619   0.2032  0.1259
##
## sigma^2 estimated as 0.8751:  log likelihood=-225.71
## AIC=459.43   AICc=459.68   BIC=471.93
```

Yule-Walker

```
(fit <- ar(death.lag12, method = "yule-walker"))

##

## Call:

## ar(x = death.lag12, method = "yule-walker")

##

## Coefficients:

##    1      2        3       4      5        6       7      8

##  0.1711   0.1425   0.0796  -0.0991   0.1661   0.0252  -0.0041
-0.0523

##    9     10       11      12

##  0.1703   0.0499  -0.0009  -0.3985

##

## Order selected 12  sigma^2 estimated as  0.7365
```

Diagnostics

```
i <- arima(death.lag12, order = c(1,0,1), method = "ML", xreg =
1:length(death.lag12))


residuals <- resid(i)
```

Checking for normality

```
plot(residuals)


hist(residuals, probability = T, main = "", xlab = "Residuals")


shapiro.test(residuals)
```

```
##

##  Shapiro-Wilk normality test

##

## data:  residuals

## W = 0.99374, p-value = 0.6922

mean(residuals)

## [1] -0.002042972

var(residuals)

## [1] 0.8555856

qqnorm(residuals)

qqline(residuals, col = "red")
```

Checking for serial correlation

```
#

Box.test(residuals, lag = 11, type = c("Ljung-Box"), fitdf = 2)

##

##  Box-Ljung test

##

## data:  residuals

## X-squared = 11.083, df = 9, p-value = 0.2701

Box.test(residuals, lag = 11, type = c("Box-Pierce"), fitdf = 2)

##

##  Box-Pierce test

##

## data:  residuals
## X-squared = 10.499, df = 9, p-value = 0.3116
```

Checking for heteroskedacity

```
op <- par(mfrow = c(1,2))

acf(residuals, main = "", lag.max = 40)

pacf(residuals, main = "", lag.max = 40)


par(op)

#we have arma(1,0,1) model
```

Forecast Prediction

```
op = par(mfrow = c(1,1))
data_pred = predict(fit, n.ahead=12)
ts.plot(ontario, xlim = c(1,200), ylim=c(0,300), ylab="Deaths")
points(169:180, exp(data_pred$pred), col = "red", pch=20)
lines(169:180, exp(data_pred$pred+1.96*data_pred$se), lty=2)
lines(169:180, exp(data_pred$pred-1.96*data_pred$se), lty=2)

#legend
legend("topleft", legend = c("Predicted", "Observed"), col=c("red","black"),
lty=1, cex = 0.7)
```

zoom in

```
# zoom in at the predicted points
ts.plot(ontario, xlim=c(160, 185), ylim=c(10, 300), main = "Zoomed-in", ylab
= "Deaths")
points(169:180, exp(data_pred$pred), col = "red", pch=20)
lines(169:180, exp(data_pred$pred+1.96*data_pred$se), lty=2)
lines(169:180, exp(data_pred$pred-1.96*data_pred$se), lty=2)

#legend
legend("bottomright", legend = c("Predicted", "Observed"),
col=c("red","black"), lty=1, cex = 0.7)
```

fitted vs observed

```r
#observed and fitted plot
ts.plot(death, lwd=1, lty=1, main = "Fitted vs Observed")
lines(exp(fit$fitted), lwd=1, lty=1, col="red")

#legend
legend("bottomright", legend = "Fitted", col="red", lty=1, cex = 0.6)
```