MSCA 31006 Assignment4

Duo Zhou

11/1/2020

Instructions:

Total number of points is 60. The assignmnet final grade will be multiplied by 1/10 to calculate its weight on the final grade.

Mark the question number and your final answer clearly (use a textbox.)

Remember to show and explain your work (If you can't explain it, you don't understand it.)

Please submit your solution through Canvas.

For this assignment, you are given a hourly traffic flow datasets from Illinois Department of Transpiration (IDOT) for I80E 1EXIT for 6/16/2013 - 7/1/2013. See the second data column in each attached file. Note: each data point is an hourly count of the number of vehicles at a specific location on I80E.

```
library(reshape2)
library(ggplot2)
library(gdata)
library(stringr)
library(xts)
library(forecast)
library(timeSeries)
library(forecast)
library(tseries)
library(TSA)
library(readxl)
```

(2 points) Question 1:

Combine the data from the 16 files into a single dataset and plot it.

```
base_path <- "C:/Users/zd000/Desktop/MSCA/Time Series/Assignments/data/Traffic Flow Data/"
date_parse_helper <- function(d) {
  d <- str_replace_all(d, "I-57-|\\.xls", "")
  as.Date(d, format = "%Y-%B-%d")
}
read_i57_xls <- function(base_path, file_name) {
  df_ <- read_excel(paste(base_path,file_name,sep="/"),sheet=1)
  df_ <- df_[5:28,c(3,5)]
  names(df_)<-c('Time','I80E_1EXIT')
  df_$date <- date_parse_helper(file_name)
  for(i in 1:24){
```

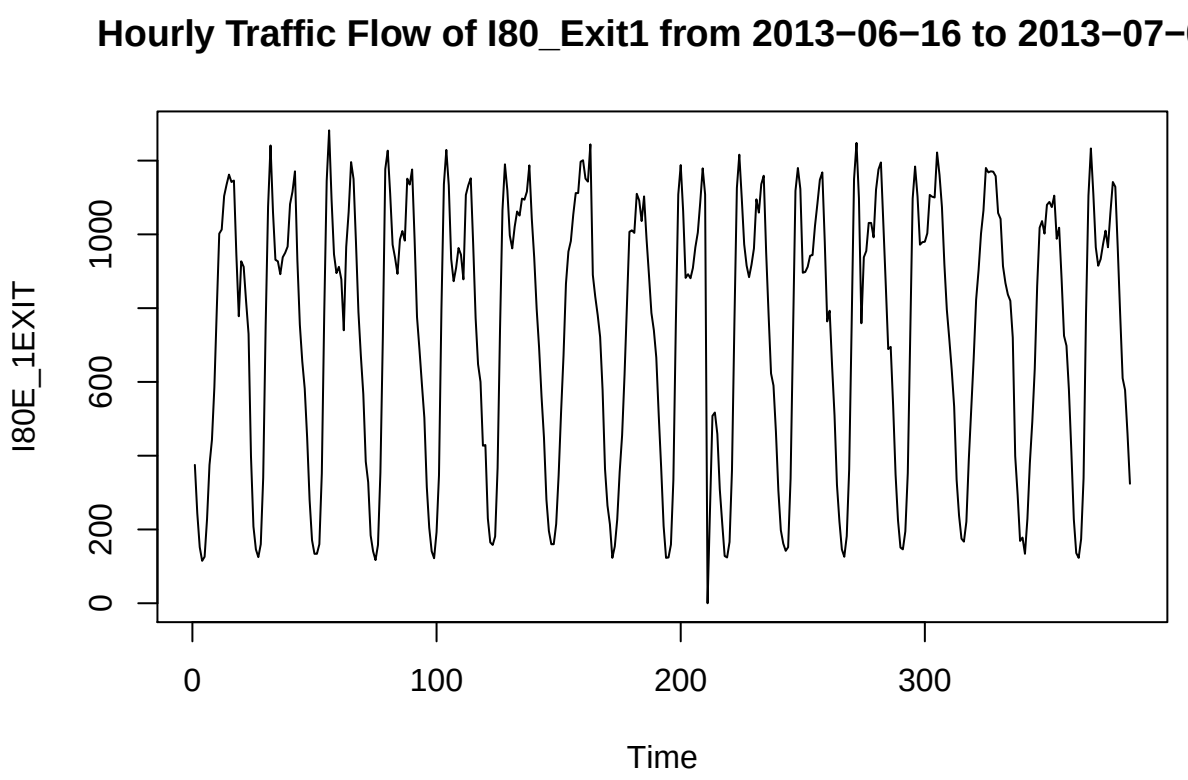
```
  df_$datetime[i] <-paste(df_$date[i],df_[i,'Time'])
  }
  return(df_[,c('datetime','I80E_1EXIT')])
}
df<-df <- plyr::ldply(list.files(base_path)[c(2:16,1)], function(f) read_i57_xls(base_path, f))

df$datetime<-strptime(df$datetime, "%Y-%m-%d %H:%M")

df[c(1:10,380:384),]
```

```
##                datetime I80E_1EXIT
## 1   2013-06-16 01:00:00        375
## 2   2013-06-16 02:00:00        244
## 3   2013-06-16 03:00:00        152
## 4   2013-06-16 04:00:00        115
## 5   2013-06-16 05:00:00        126
## 6   2013-06-16 06:00:00        228
## 7   2013-06-16 07:00:00        375
## 8   2013-06-16 08:00:00        443
## 9   2013-06-16 09:00:00        586
## 10  2013-06-16 10:00:00        800
## 380 2013-07-01 20:00:00        782
## 381 2013-07-01 21:00:00        610
## 382 2013-07-01 22:00:00        578
## 383 2013-07-01 23:00:00        459
## 384 2013-07-01 00:00:00        324
```

```
plot(ts(df$I80E_1EXIT),ylab='I80E_1EXIT', main='Hourly Traffic Flow of I80_Exit1 from 2013-06-16 to 201
```



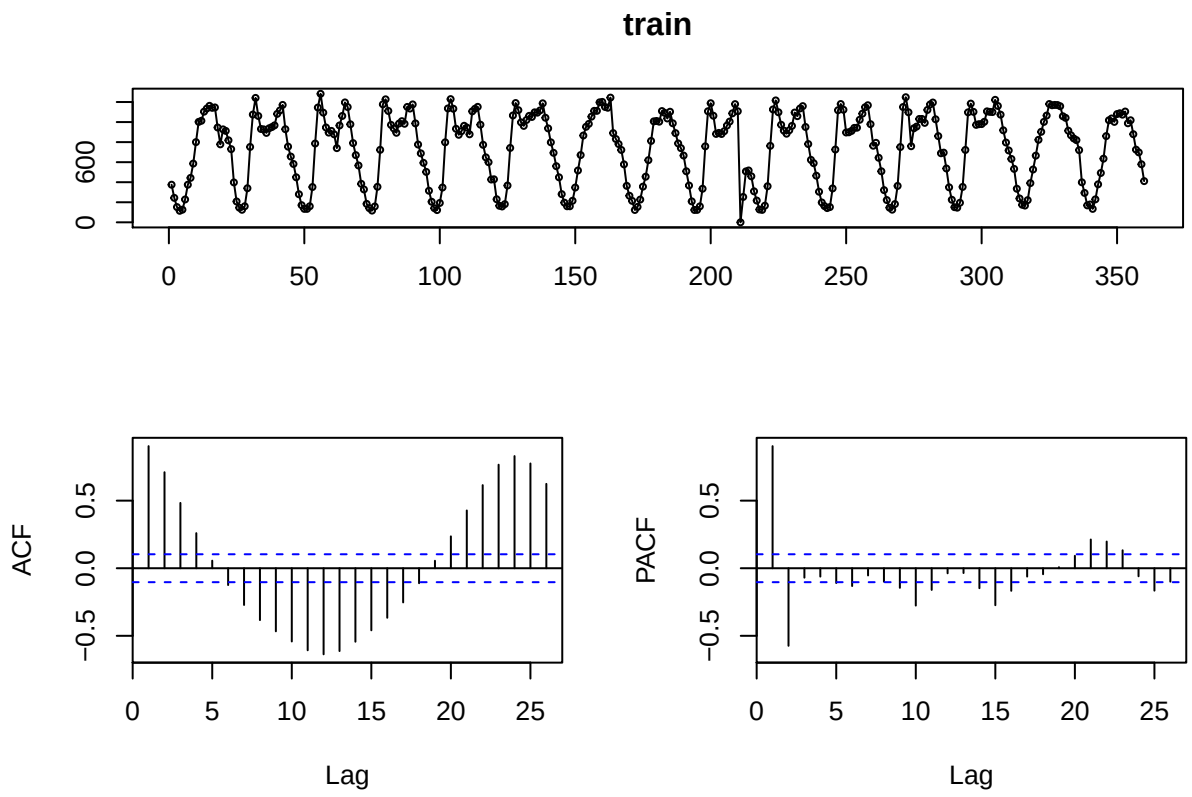
We can clearly see seasonality in the TS plot.

(3 points) Question 2:

Split the dataset into a training dataset which includes 6/16/2013 - 6/30/2013 samples and a test dataset which includes 7/1/2013 samples. Plot the ACF and PACF, and apply the Augmented Dickey-Fuller Test to check if the training dataset is stationary

```
time_index <- seq(from = as.POSIXct("2013-06-16 01:00"),
                  to = as.POSIXct("2013-07-02 0:00"), by = "hour")
obs <- xts(df['I80E_1EXIT'], order.by = time_index)
# first 360 data points are from 6/16/2013 - 6/30/2013
train <- as.numeric(ts(obs[1:360]))
# last 24 data points are 7/1/2013 samples
test <- as.numeric(ts(obs[361:384]))
```

```
tsdisplay(train)
```




```
tseries::adf.test(train,k=24)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  train
## Dickey-Fuller = -3.6372, Lag order = 24, p-value = 0.02966
## alternative hypothesis: stationary
```

Up to lag order = 24, P=0.02966 < 0.05. We can reject H0 and accept the alternative hypothesis that training data is stationary.

(10 points) Question 3:

Build an ARIMA(p,d,q) model using the training dataset and R auto.arima() function. Change the values of p and q and determine the best model using AICc and BIC values. Do AICc and BIC select the same model as the best model? For each derived model, review the residual plots for the residuals ACF and normality.

```
fit1 <- auto.arima(train, seasonal = F,stepwise = T, trace=T, approximation = FALSE)
```

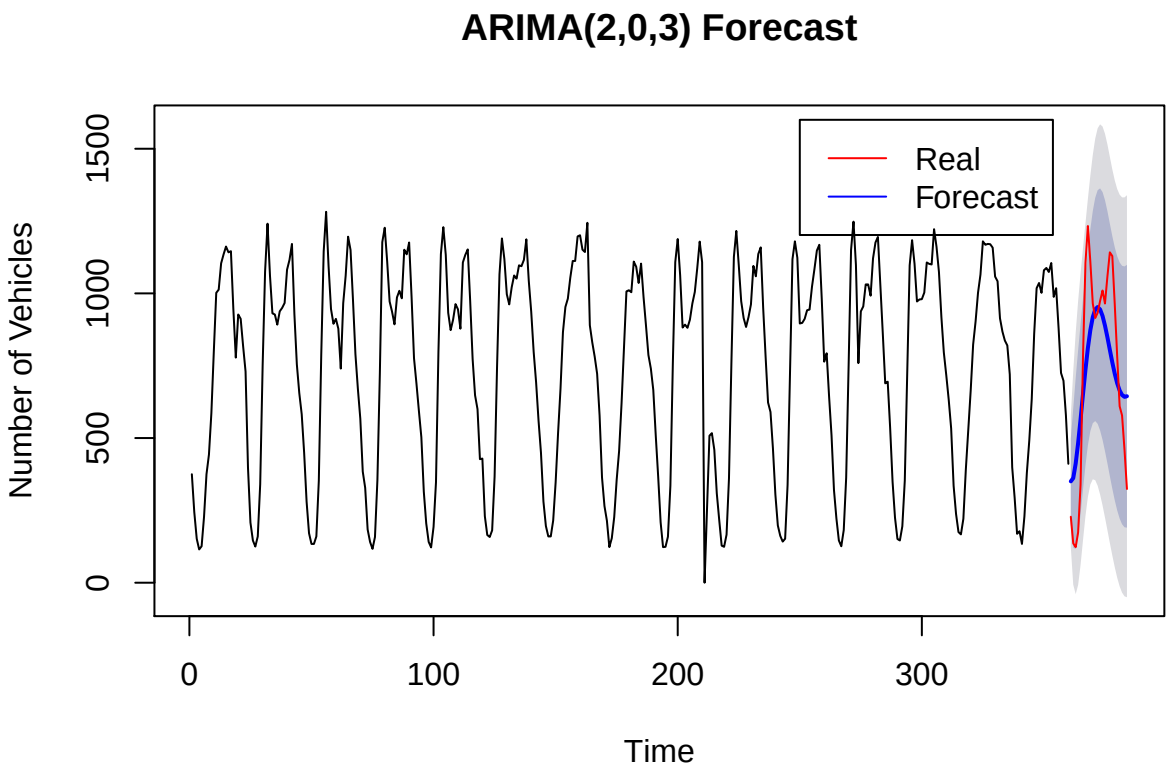
```
##
##  ARIMA(2,0,2) with non-zero mean : 4456.755
##  ARIMA(0,0,0) with non-zero mean : 5263.162
##  ARIMA(1,0,0) with non-zero mean : 4648.822
```

```
##  ARIMA(0,0,1) with non-zero mean : 4888.607
##  ARIMA(0,0,0) with zero mean     : 5858.997
##  ARIMA(1,0,2) with non-zero mean : 4517.937
##  ARIMA(2,0,1) with non-zero mean : 4498.756
##  ARIMA(3,0,2) with non-zero mean : 4457.253
##  ARIMA(2,0,3) with non-zero mean : 4455.881
##  ARIMA(1,0,3) with non-zero mean : 4510.955
##  ARIMA(3,0,3) with non-zero mean : 4457.678
##  ARIMA(2,0,4) with non-zero mean : 4456.163
##  ARIMA(1,0,4) with non-zero mean : 4512.573
##  ARIMA(3,0,4) with non-zero mean : Inf
##  ARIMA(2,0,3) with zero mean     : 4548.013
##
##  Best model: ARIMA(2,0,3) with non-zero mean
```

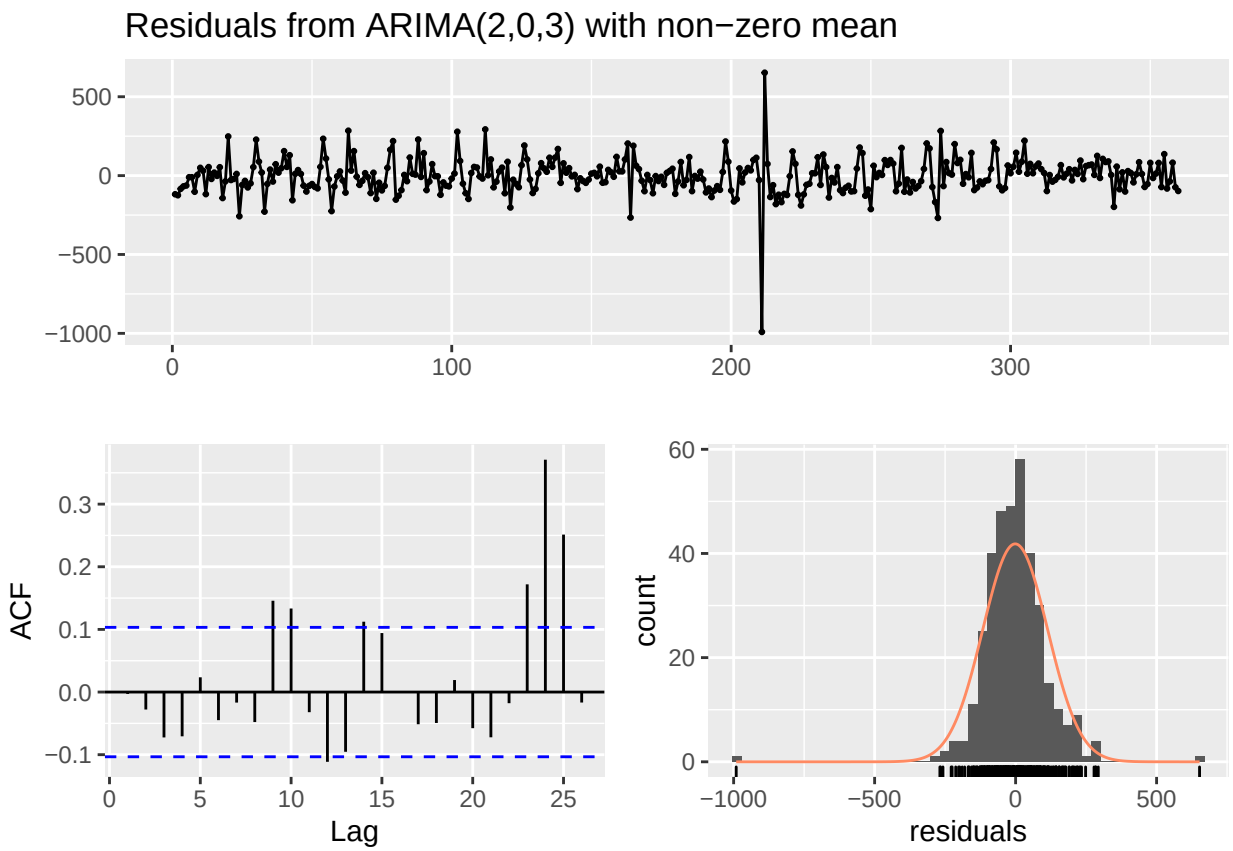
```
summary(fit1)
```

```
## Series: train
## ARIMA(2,0,3) with non-zero mean
##
## Coefficients:
##          ar1      ar2      ma1      ma2      ma3      mean
##       1.8088  -0.8853  -0.5348  -0.2671  -0.1157  746.3181
## s.e.  0.0288   0.0287   0.0600   0.0596   0.0654    6.8586
##
## sigma^2 estimated as 13443:  log likelihood=-2220.78
## AIC=4455.56   AICc=4455.88   BIC=4482.77
##
## Training set error measures:
##                      ME     RMSE    MAE  MPE MAPE      MASE         ACF1
## Training set -1.390098 114.9732 79.019 -Inf  Inf 0.7027304 -0.003018285
```

```
plot(forecast(fit1, 24), xlab="Time", ylab="Number of Vehicles",main="ARIMA(2,0,3) Forecast")
lines(x=c(361:384), y =test, col="red")
legend(250, 1600, legend=c("Real", "Forecast"),
       col=c("red", "blue"), lty=1)
```

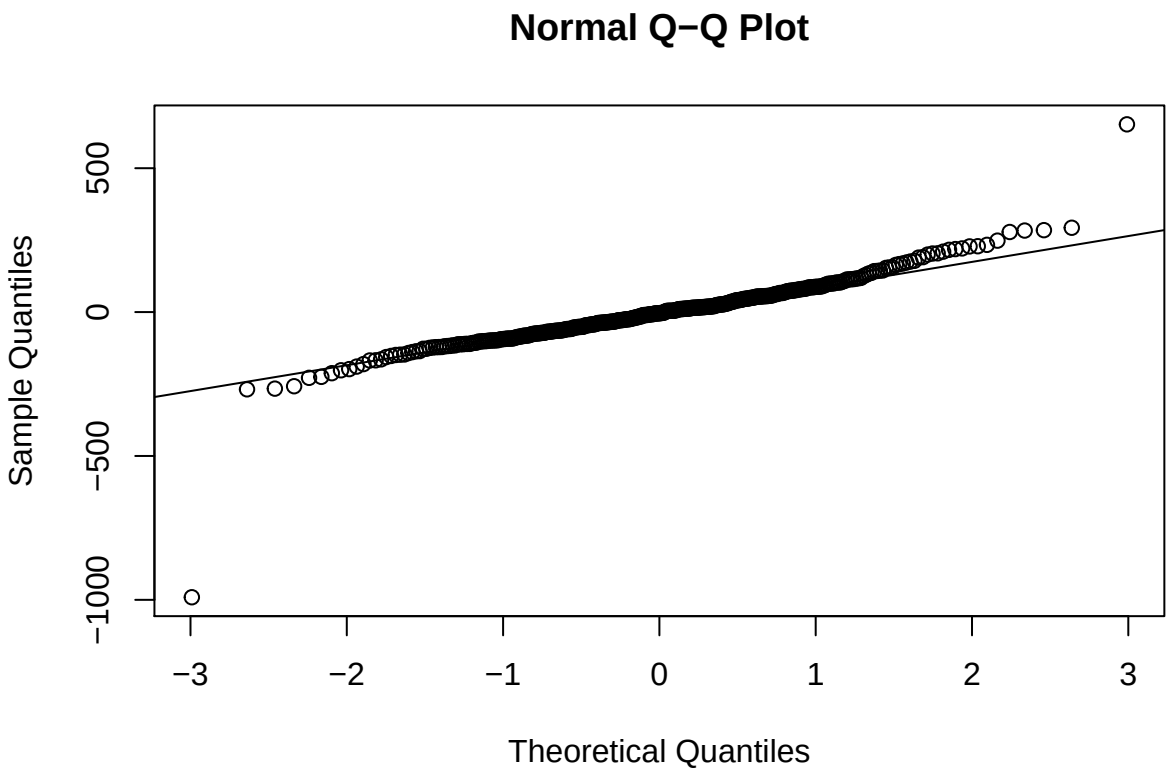



```
checkresiduals(fit1)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(2,0,3) with non-zero mean
## Q* = 20.439, df = 4, p-value = 0.0004089
##
## Model df: 6.   Total lags used: 10
```

```
qqnorm(fit1$residuals)
qqline(fit1$residuals)
```



The auto.arima() function returns a model of ARIMA(2,0,3) with AICc = 4455.88 and BIC = 4482.77. In The forecast plot, red line is the actual number of vehicles and blue line is the forecast line, and as we could see that blue line does not match well with the red line. Also in the residual plot, there is a huge spike around the middle. ACF of the residuals suggests that there are still auto-correlations that was not captured in the model. The QQ plot shows that there are a few outliers in the residual that makes the residual distribution deviates from normality around the tails.

I am going to change p and q up to 5 and see which combination of p and q gives the best AICc and BICc.

```
AICc_min <- 5000
AICc_min_p <- 0
AICc_min_q <- 0
BIC_min <- 5000
BIC_min_p <- 0
BIC_min_q <- 0
for (p in 1:5){
  for (q in 1:5){
        fit11 <- Arima(train, order = c(p,0,q))
        AICc <- fit11$aicc
        BIC <-fit11$bic
        if(AICc < AICc_min){
        AICc_min <- AICc
        AICc_min_p <- p
        AICc_min_q <- q}
        if(BIC < BIC_min){
            BIC_min <- BIC
            BIC_min_p <- p
```

```
            BIC_min_q <- q}
  }
  }

cbind(AICc_min=AICc_min, AICc_min_p=AICc_min_p,AICc_min_q=AICc_min_q)
```

```
##      AICc_min AICc_min_p AICc_min_q
## [1,] 4409.439          4          3
```

```
cbind(BIC_min=BIC_min,BIC_min_p=BIC_min_p,BIC_min_q= BIC_min_q)
```

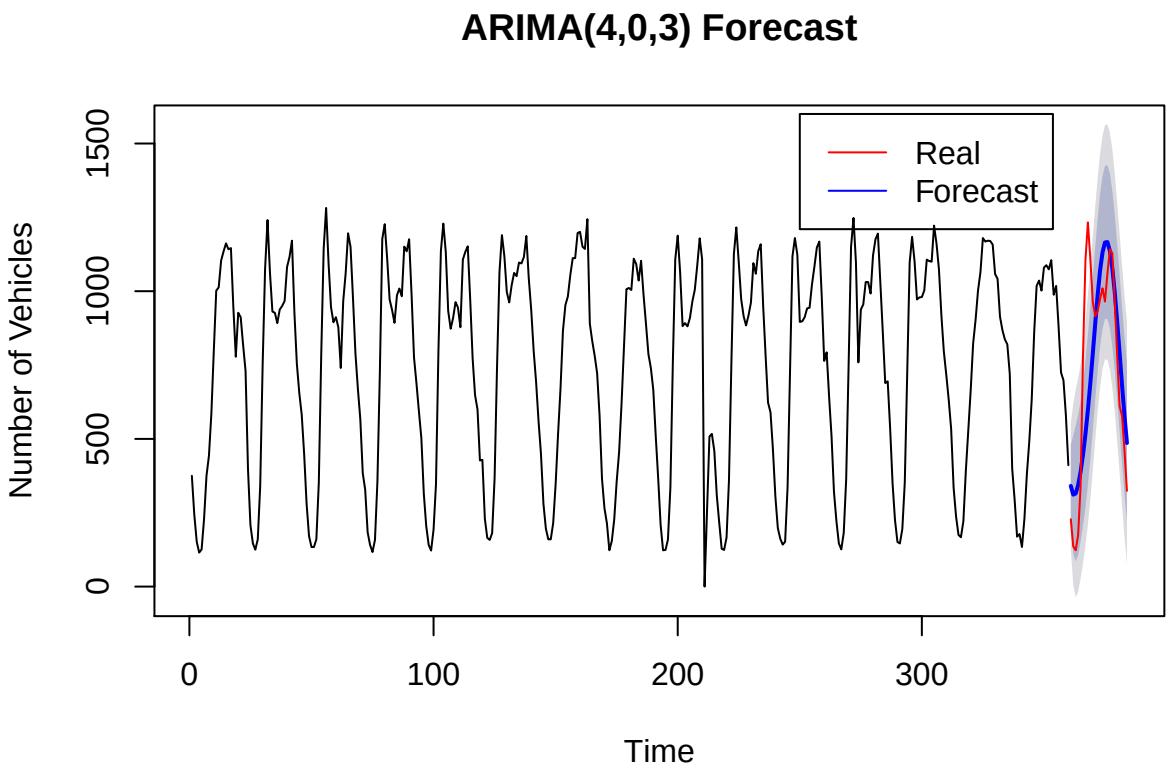
```
##      BIC_min BIC_min_p BIC_min_q
## [1,]  4443.9         4         3
```

Based on AICc and BIC, the same model was selected as the best model: ARIMA(4,0,3) with AICc=4409.439 and BIC=4443.9.

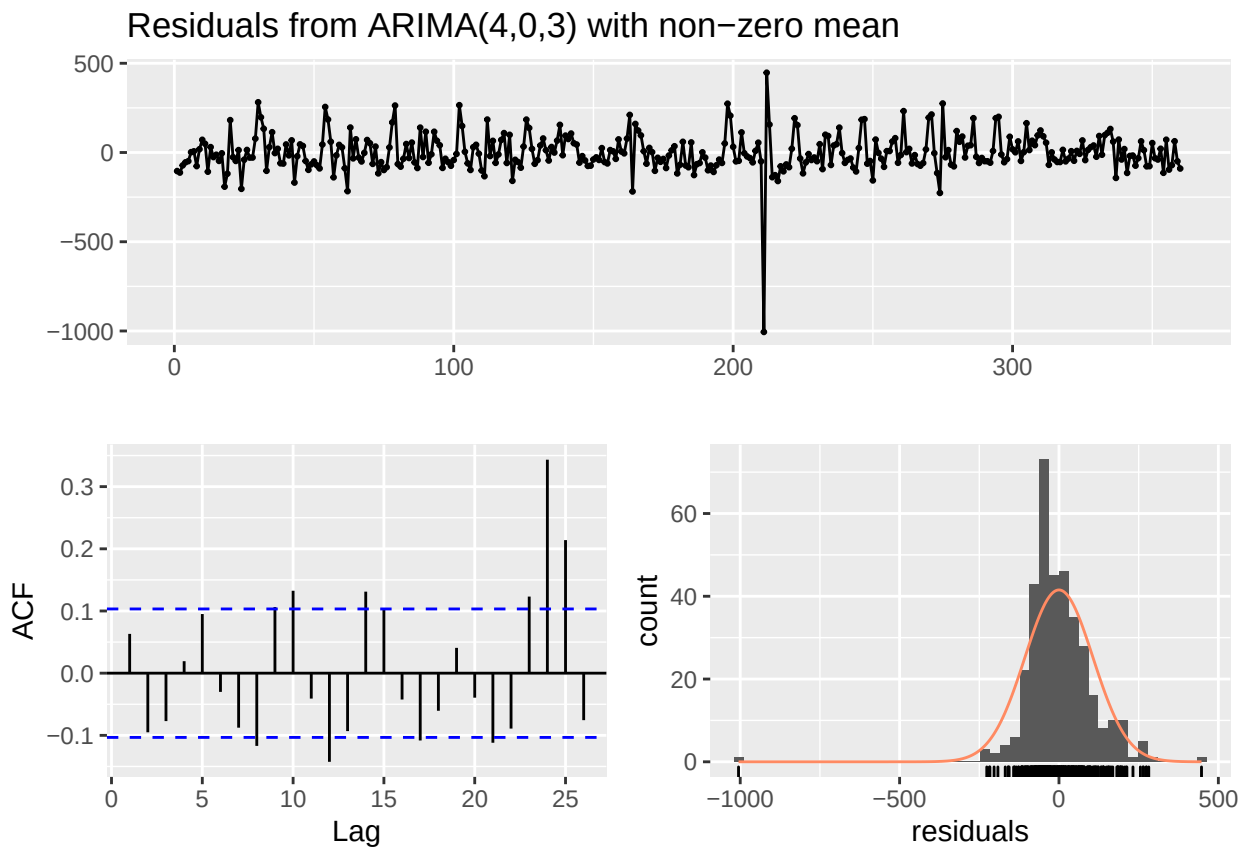
```
fit.best <- Arima(train, order=c(4,0,3))
fit.best
```

```
## Series: train
## ARIMA(4,0,3) with non-zero mean
##
## Coefficients:
##          ar1      ar2     ar3      ar4      ma1     ma2      ma3      mean
##       3.4090  -4.6365  2.9893  -0.7825  -2.3609  1.8743  -0.4778  743.2782
## s.e.  0.1766   0.4835  0.4519   0.1428   0.3014  0.6100   0.3282    9.8632
##
## sigma^2 estimated as 11649:  log likelihood=-2195.46
## AIC=4408.92   AICc=4409.44   BIC=4443.9
```

```
plot(forecast(fit.best, 24), xlab="Time", ylab="Number of Vehicles",main="ARIMA(4,0,3) Forecast")
lines(x=c(361:384), y =test, col="red")
legend(250, 1600, legend=c("Real", "Forecast"),
       col=c("red", "blue"), lty=1)
```

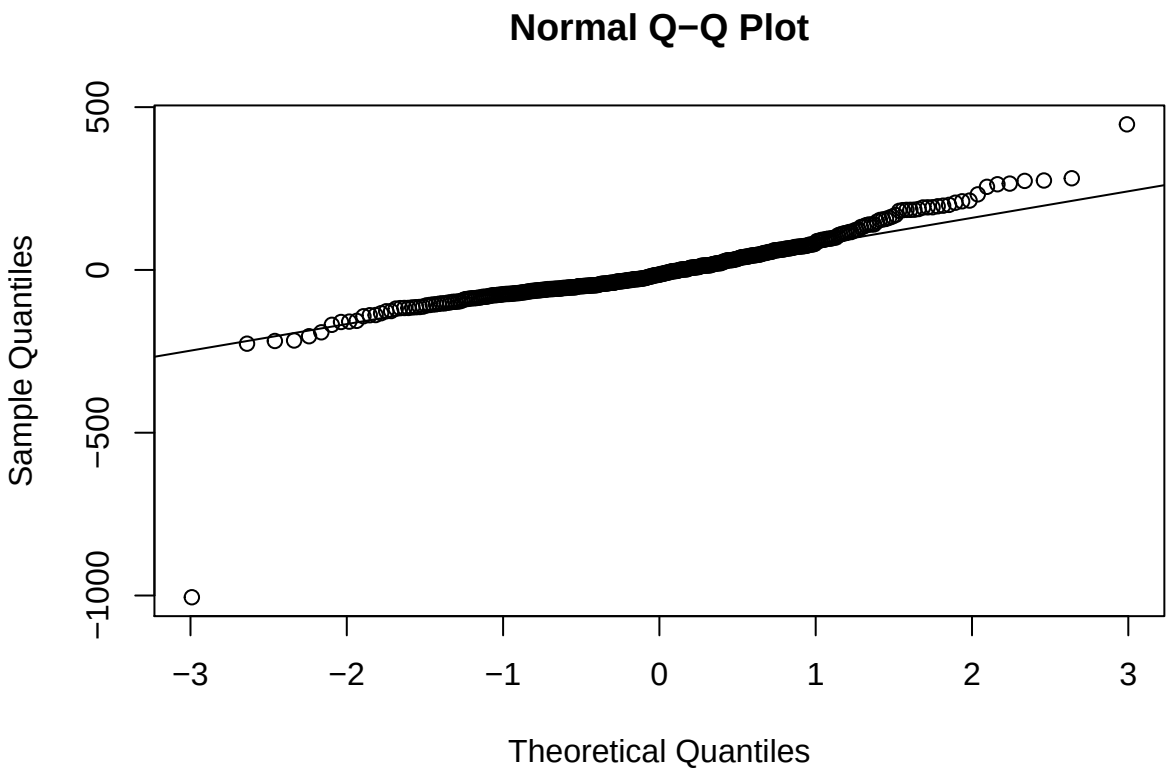



```
checkresiduals(fit.best)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(4,0,3) with non-zero mean
## Q* = 29.959, df = 3, p-value = 1.408e-06
##
## Model df: 8.   Total lags used: 11
```

```
qqnorm(fit.best$residuals)
qqline(fit.best$residuals)
```

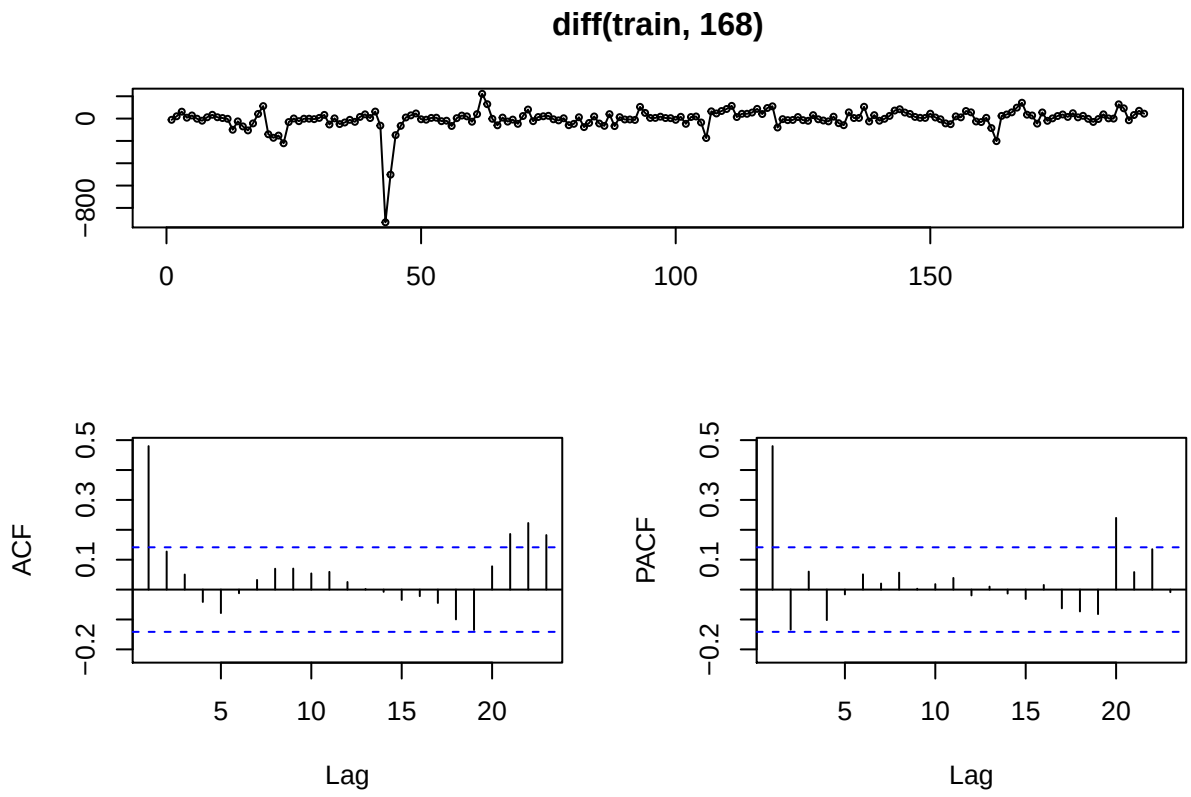


Although ARIMA(4,0,3) fits the test data much better thatn ARIMA(2,0,3). The residual plots of ARIMA(4,0,3) still show that same issues as ARIMA(2,0,3). These show that our model needs further improvement. We clearly see the seaonality in our train data and residuals ACF shows significant autocorrelation in lag=23 and 24. We need to consider sARIMA model.

(10 points) Question 4:

Build a day of the week seasonal ARIMA(p,d,q)(P,D,Q) model using the training dataset and R auto.arima() function.

```
# use day of the week: s=24*7=168
tsdisplay(diff(train,168))
```




```
# use day of the week: s=24*7=168
fit2 <- auto.arima(ts(train, frequency=168),seasonal=T)
```

```
## Warning: The chosen seasonal unit root test encountered an error when testing for the second differe
## From stl(): series is not periodic or has less than two periods
## 1 seasonal differences will be used. Consider using a different unit root test.
```

```
fit2
```

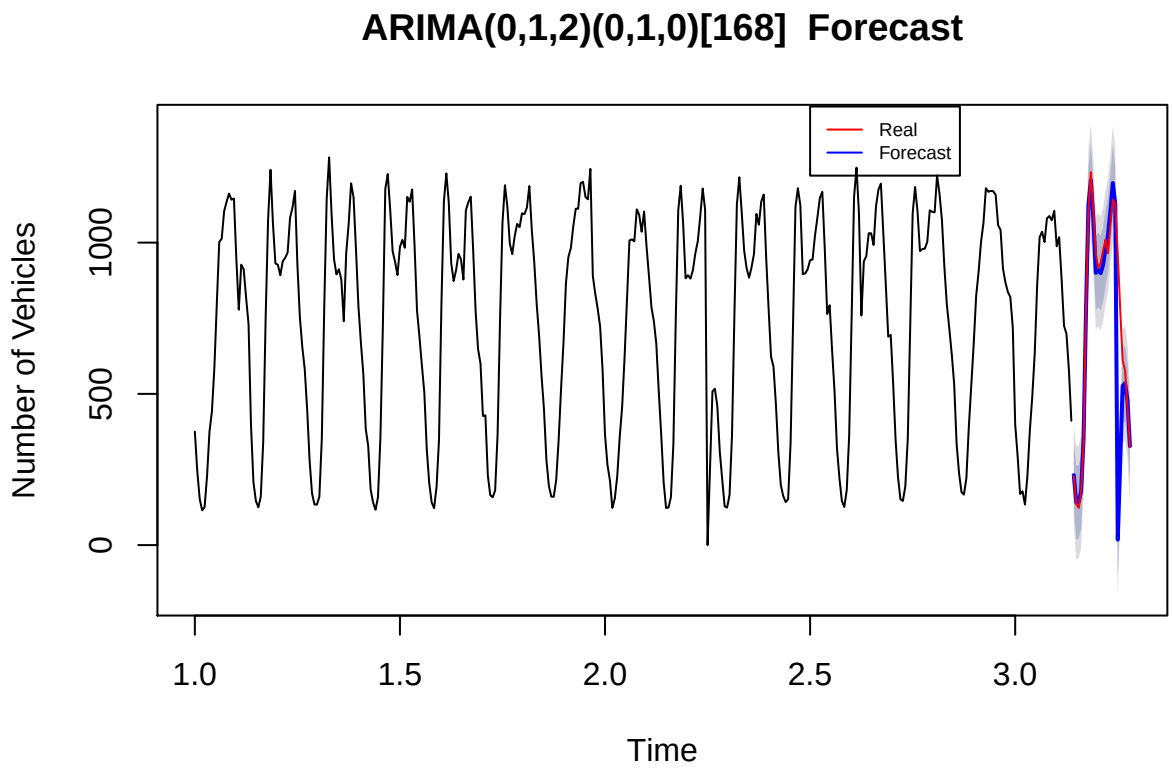
```
## Series: ts(train, frequency = 168)
## ARIMA(0,1,2)(0,1,0)[168]
##
## Coefficients:
##           ma1      ma2
##       -0.4741  -0.4853
## s.e.   0.0593   0.0586
##
## sigma^2 estimated as 7081:  log likelihood=-1121.66
## AIC=2249.31   AICc=2249.44   BIC=2259.07
```

(10 points) Question 5:

Use the model ARIMA(p,d,q)(P,D,Q) from Question 4 to forecast for July 1st (which is a Monday). Plot your result.

```r
# forecast for July 1

plot(forecast(fit2, 24), xlab="Time",
     ylab="Number of Vehicles",main="ARIMA(0,1,2)(0,1,0)[168]  Forecast")
lines(x=c(time(forecast(fit2, 24)$mean)), y =ts(test, frequency = 168) , col="red")
legend(2.5, 1450, legend=c("Real", "Forecast"),
       col=c("red", "blue"), lty=1,cex=0.6)
```



We can see that the fit is much better.

(10 points) Question 6:

Build a hour of the day seasonal ARIMA(p,d,q)(P,D,Q) model using the training dataset and R auto.arima() function.

```r
# use hour of the day: s=24
fit3 <- auto.arima(ts(train, frequency=24),seasonal=T)
fit3
```

```
## Series: ts(train, frequency = 24)
## ARIMA(2,0,2)(2,1,0)[24]
##
## Coefficients:
##          ar1      ar2      ma1      ma2     sar1     sar2
##       1.5965  -0.6895  -0.6772  -0.1650  -0.4151  -0.3254
```

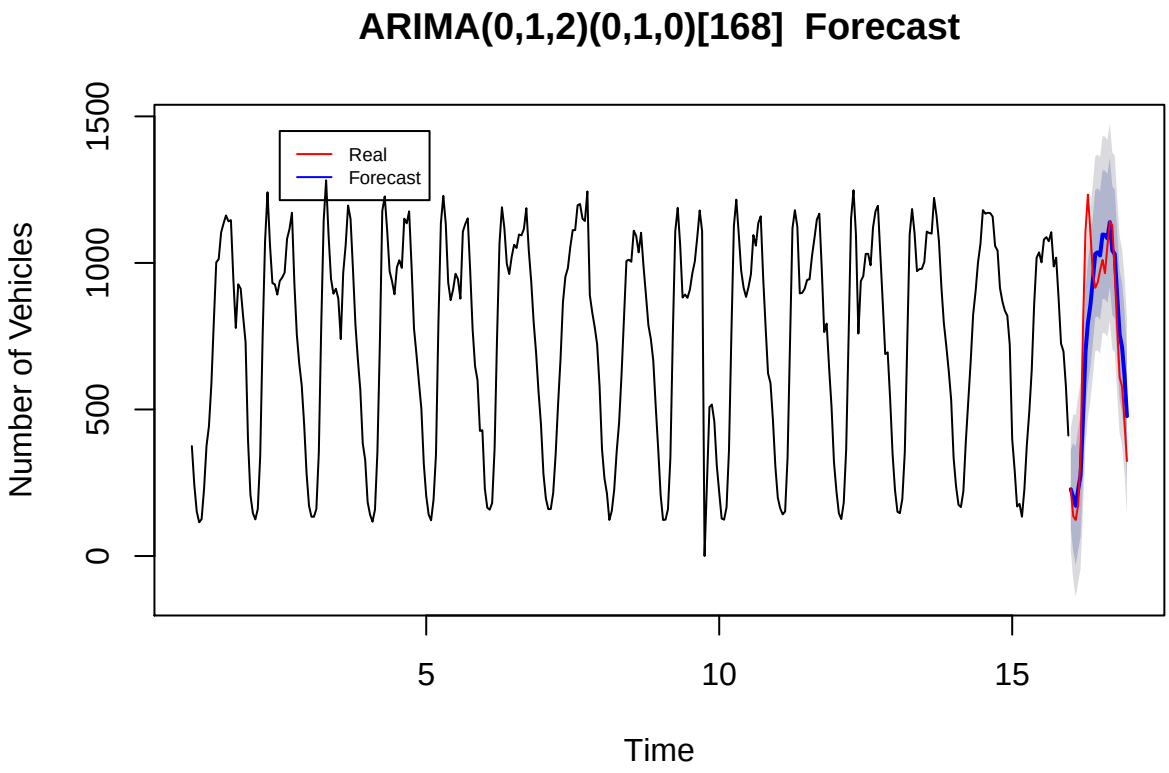
```
## s.e.  0.0639    0.0521    0.0815    0.0694    0.0565    0.0535
##
## sigma^2 estimated as 11243:  log likelihood=-2045.28
## AIC=4104.56   AICc=4104.9   BIC=4131.28
```

(10 points) Question 7:

Use the ARIMA(p,d,q)(P,D,Q) model from Question 6 to forecast for July 1st (which is a Monday). Plot your result.

```
# forecast for July 1

plot(forecast(fit3, 24), xlab="Time",
     ylab="Number of Vehicles",main="ARIMA(0,1,2)(0,1,0)[168]  Forecast")
lines(x=c(time(forecast(fit3, 24)$mean)), y =ts(test, frequency = 24) , col="red")
legend(2.5, 1450, legend=c("Real", "Forecast"),
       col=c("red", "blue"), lty=1,cex=0.6)
```



(5 points) Question 8:

Compare the forecast of the models from Questions 5 and 7 for July 1 8:00, 9:00, 17:00 and 18:00, which model is better (Questions 4 or 6)?

The sum of squared error(SSE) of these 4 forecast points from Q5 model and Q7 model are calculated for comparison. Whichever has the smaller SSE is a better model.

```r
# selecting forecast from both model at July 1 8:00, 9:00, 17:00 and 18:00
forecast_fit2<-forecast(fit2, 24)$mean[c(8,9,17,18)]
forecast_fit3<-forecast(fit3, 24)$mean[c(8,9,17,18)]
# selecting real test data at July 1 8:00, 9:00, 17:00 and 18:00
test_4<-test[c(8,9,17,18)]
cat('SSE of the 4 time points from Q5 model is', sum((forecast_fit2-test_4)^2),'\n')
```

```
## SSE of the 4 time points from Q5 model is 4604.173
```

```r
cat('SSE of the 4 time points from Q7 model is', sum((forecast_fit3-test_4)^2))
```

```
## SSE of the 4 time points from Q7 model is 265420.8
```

Clearly the Q5 model has a much smaller SSE at this 4 time points. We can also see from the forecast plot that Q5 model fit much better than Q7 model with the real data. Hence Q5 Model is a better model for forecasting.