Time Series Analysis - Assignment 4

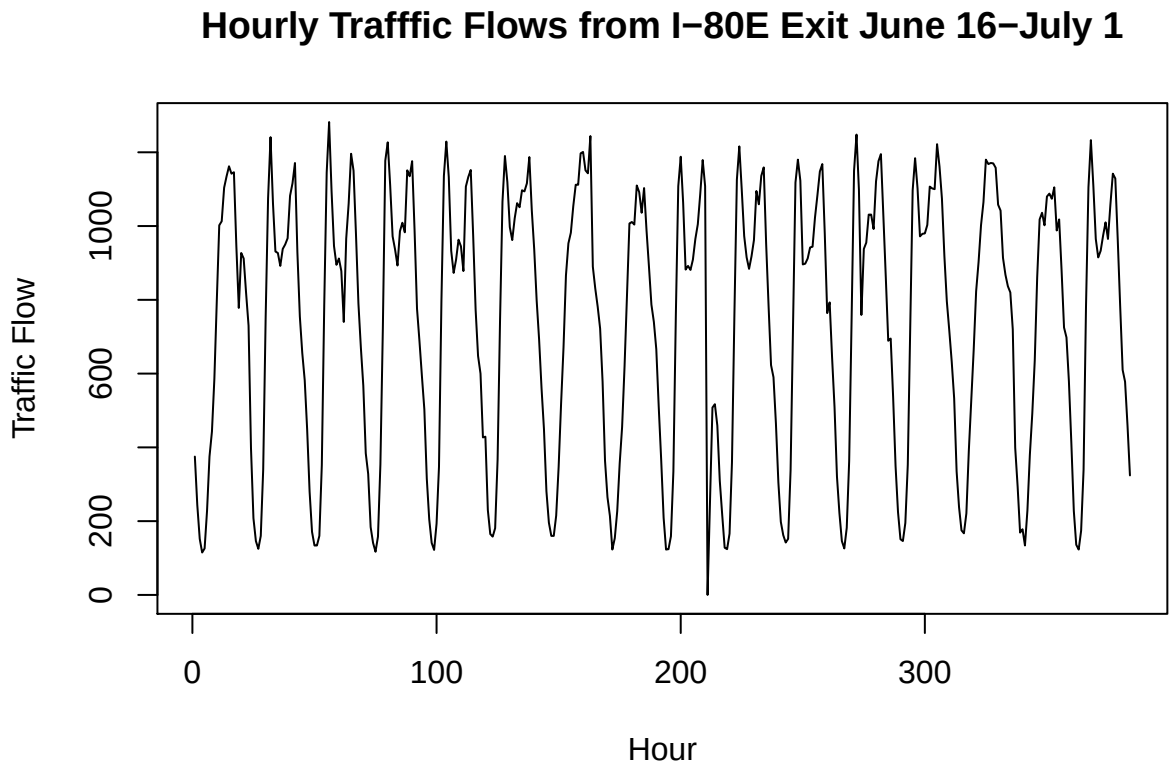
Matthew Dunne

July 14, 2018

Question 1

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

```
## [1] 384
```



Question 2

Split the dataset into a training dataset which includes 6/13/2013 - 6/30/2013 and Test dataset which includes 7/1/2013.

I split the data into the first 360 observartions for the training and the last 24 for the testing data (representing the data for July 1).

```
train<-window(data, start=1, end= 360)
test<-window(data, start=361, end= 384)
nrow(train)
```

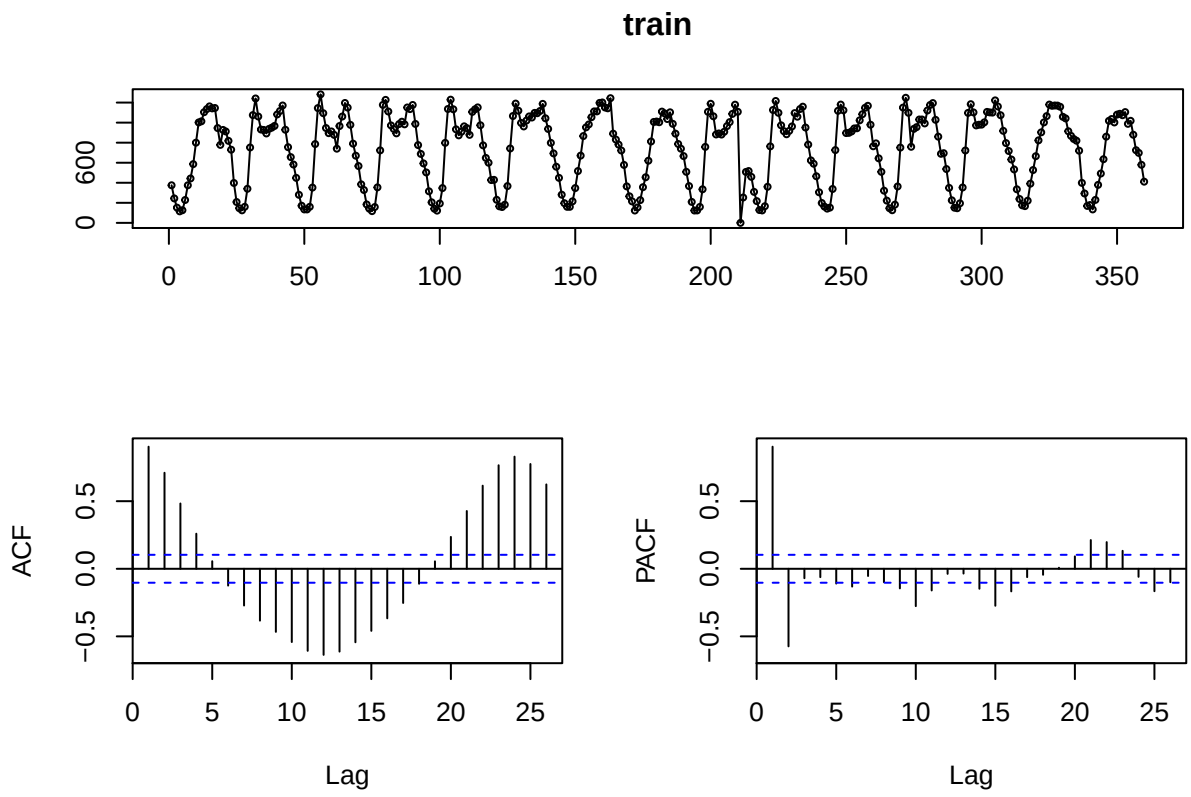
```
## [1] 360
```

```
nrow(test)
```

```
## [1] 24
```

Let us look at the ACF and PACF of the training data.

```
tsdisplay(train)
```



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 ACF of residuals and residual normality.

The auto.arima() function yields a model of ARIMA(2,0,3) with a non-zero mean.

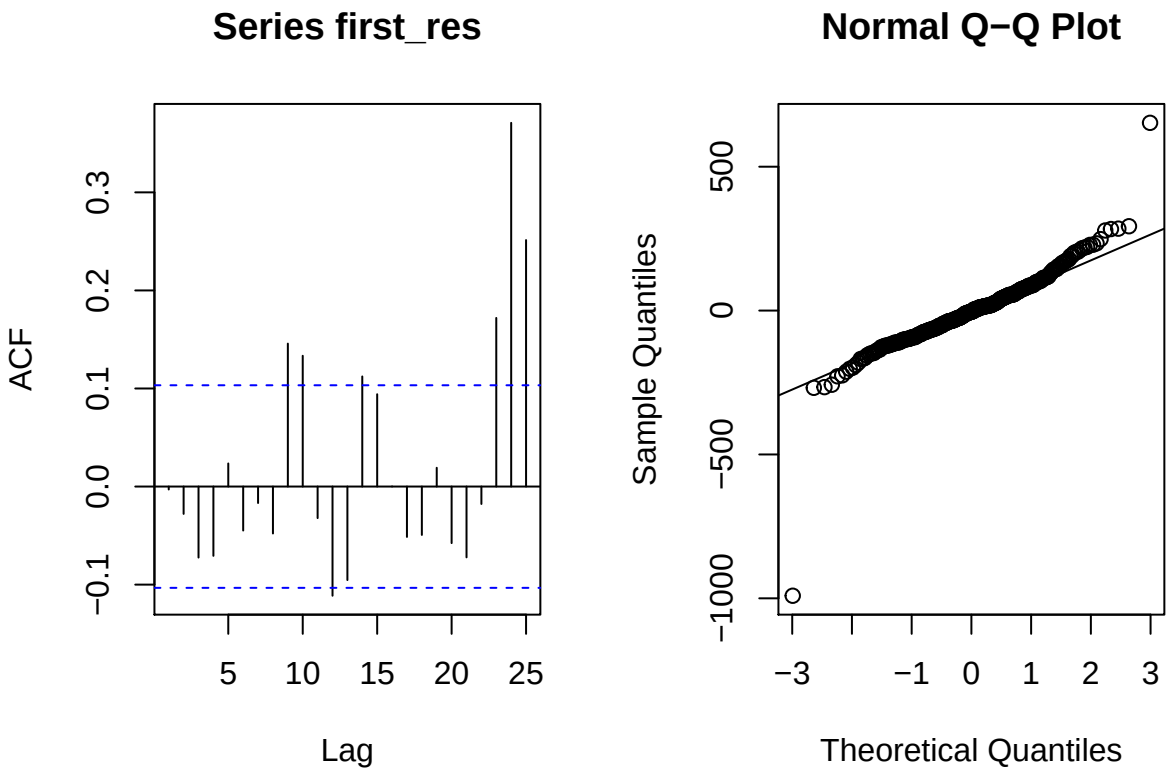
```
suppressMessages(library(fpp))
suppressMessages(library(TSA))
first<-auto.arima(train, seasonal=FALSE)
first_res<-first$residuals
first
```

```
## 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
```

In looking at the ACF of the residuals of this model we see there is still evidence of autocorrelation in the residuals. The Q-Q plot shows the residuals more or less fit the normal distribution, except at the extreme values.

```
first_res<-first$residuals
par(mfrow=c(1,2))
acf(first_res)
qqnorm(first_res)
qqline(first_res)
```



How might we change this? As a starting point for thinking about different parameters for the model remember that the PACF of the training data strongly suggests an AR term (p) of 2. Let us check if the MA(3) part is necessary. If we run auto.arima() with the constraing of p at maximum of 2 and q at maximum of 2 (one less than the previous model) it suggests that the MA component is not needed at that level.

```
second<-auto.arima(train, max.p=2, max.q=2, seasonal=FALSE)
second
```

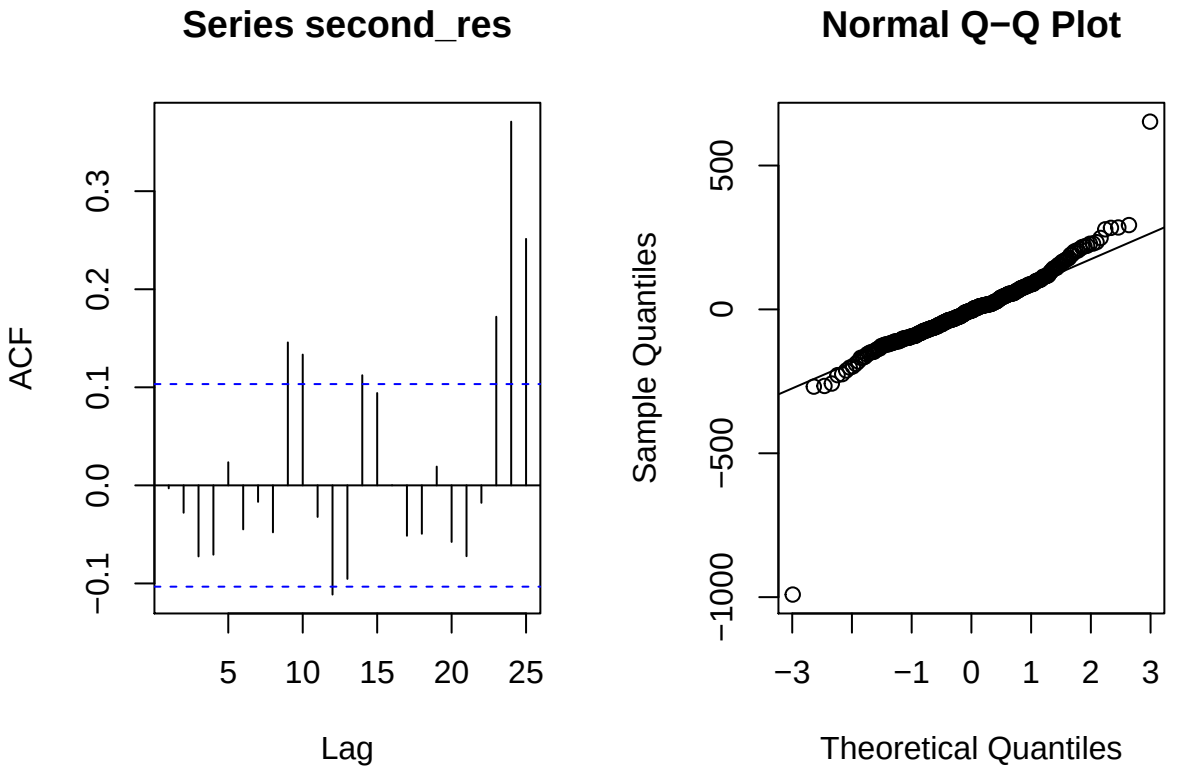
```
## Series: train
## ARIMA(2,0,2) with non-zero mean
##
```

```
## Coefficients:
##          ar1      ar2      ma1      ma2      mean
##       1.8308  -0.9072  -0.5916  -0.3254  746.3649
## s.e.  0.0229   0.0228   0.0488   0.0471    6.9120
##
## sigma^2 estimated as 13514:  log likelihood=-2222.26
## AIC=4456.52   AICc=4456.76   BIC=4479.83
```

This is a more parsimonious model with an AICc and BIC nearly identical to the first model.

Given the similarity in AICc and BIC, the ACF and Q-Q plots of the residuals of this second model are what one might expect: the same basic autocorrelation in the ACF plot and the same basic Q-Q plot.

```
second_res<-second$residuals
par(mfrow=c(1,2))
acf(second_res)
qqnorm(second_res)
qqline(second_res)
```



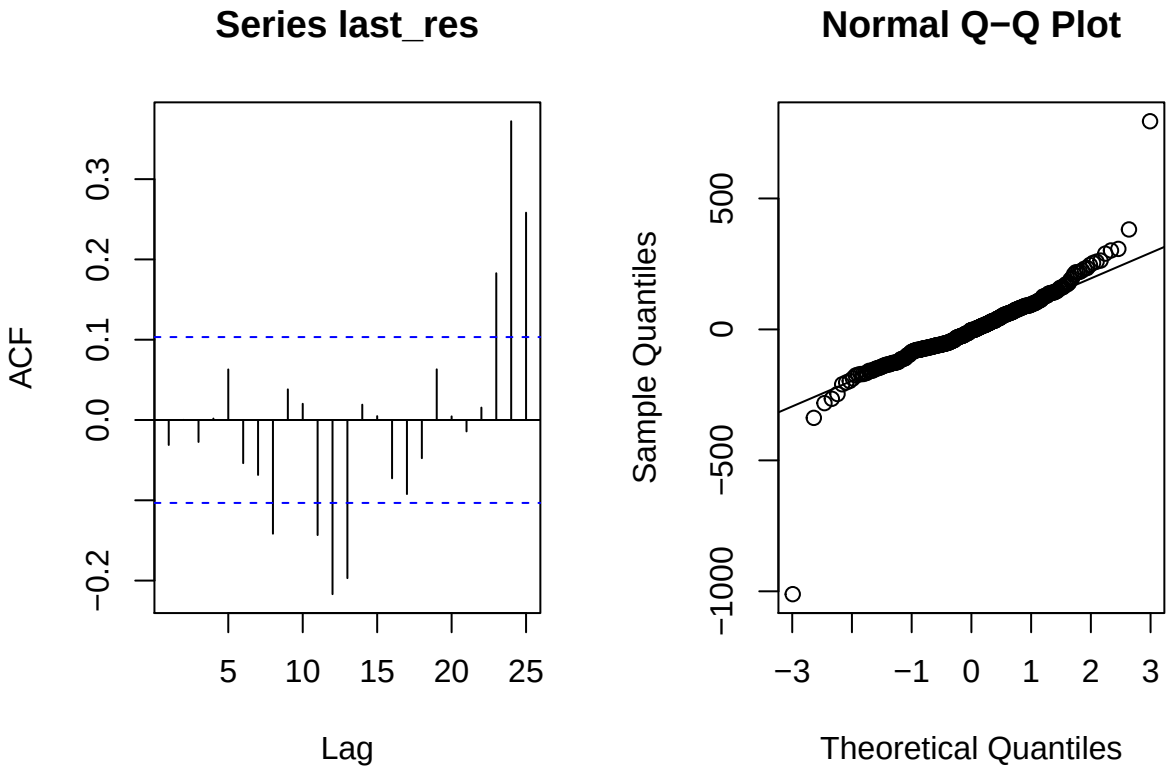
Interestingly, if we decrease the q all the way down to 0 the AICc and BIC do not change that much. Likewise, the ACF and Q-Q plots are very similar.

```
last<-auto.arima(train, max.p=2, max.q=0, seasonal=FALSE)
last
```

```
## Series: train
## ARIMA(2,0,0) with non-zero mean
##
## Coefficients:
```

```
##          ar1      ar2      mean
##       1.4390  -0.5890  739.0216
## s.e.  0.0425   0.0426   43.0839
##
## sigma^2 estimated as 15307:  log likelihood=-2245.08
## AIC=4498.16   AICc=4498.28   BIC=4513.71
```

```
last_res<-last$residuals
par(mfrow=c(1,2))
acf(last_res)
qqnorm(last_res)
qqline(last_res)
```



For the sake of simplicity and parsimony, I would select this last model which is no longer an ARIMA model but is actually just an AR(2) model.

Question 4

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

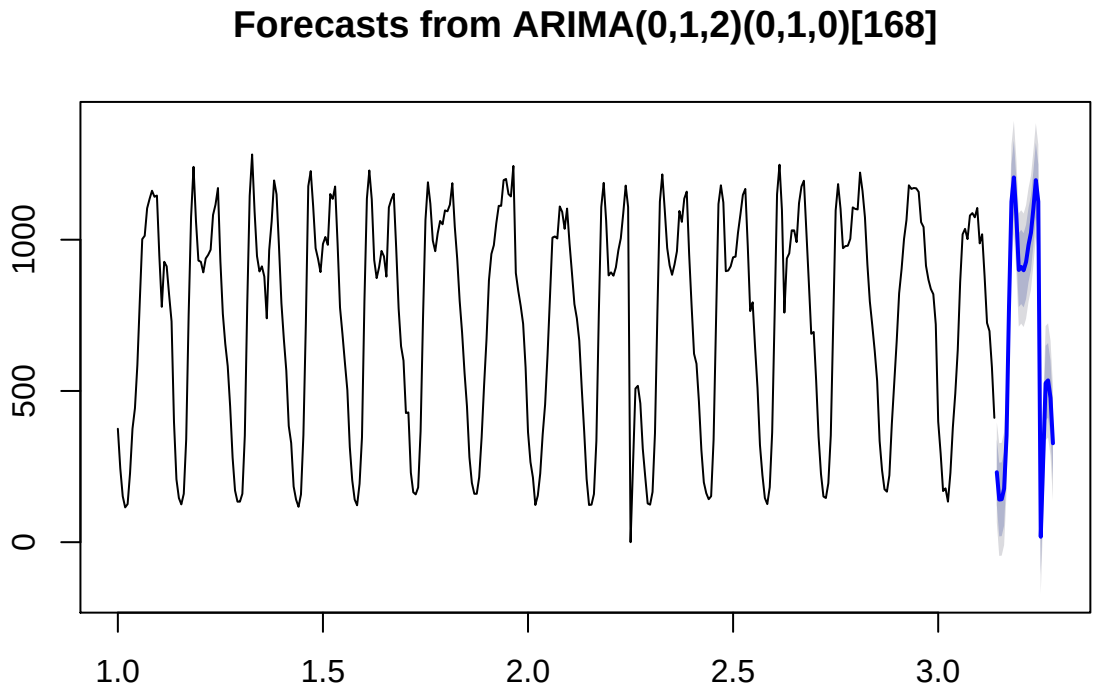
```
#set the frequency in the ts() object to 168 to represent a weekly data
day_data<-ts(train, frequency = 168)
#set the seasonal argument to TRUE
day_seasonal<-auto.arima(day_data, seasonal = TRUE)
day_seasonal
```

```
## Series: day_data
## 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
```

Question 5

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

```
plot(forecast(day_seasonal,h=24))
```



Question 6

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

```
#set the frequency in the ts() object to 24 to represent a daily data
hour_data<-ts(train, frequency = 24)
```

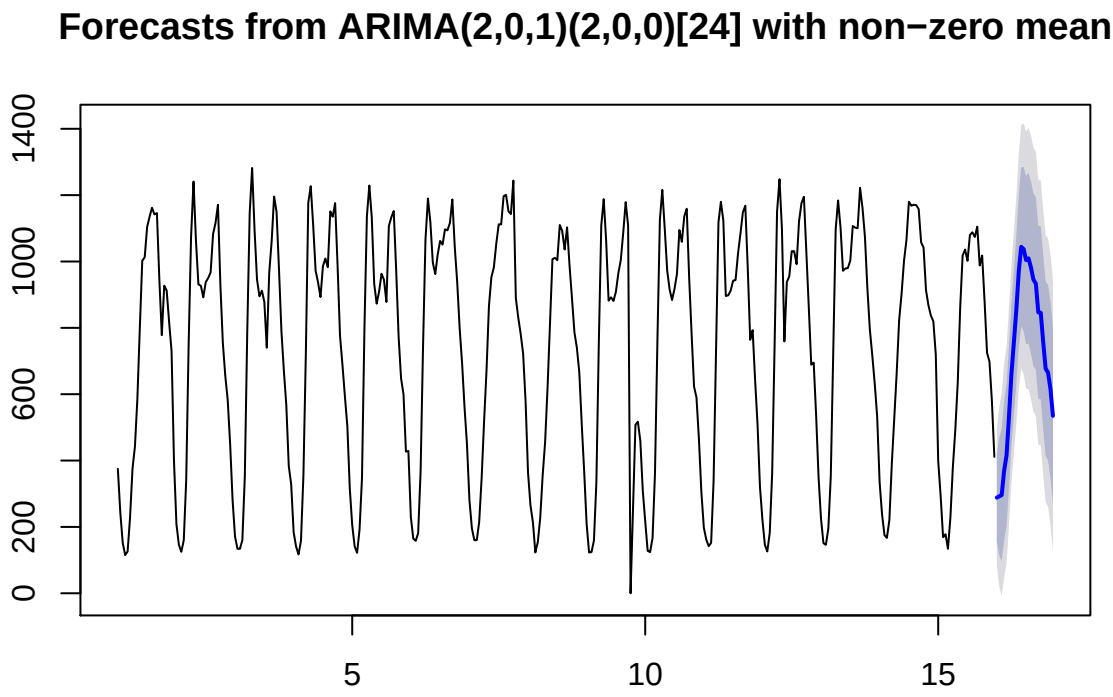
```r
#set the seasonal argument to TRUE
hour_seasonal<-auto.arima(hour_data, seasonal = TRUE)
hour_seasonal
```

```
## Series: hour_data
## ARIMA(2,0,1)(2,0,0)[24] with non-zero mean
##
## Coefficients:
##          ar1      ar2      ma1    sar1    sar2      mean
##       1.7922  -0.8685  -0.9146  0.4866  0.1010  743.7286
## s.e.  0.0299   0.0291   0.0257  0.0555  0.0557   13.6793
##
## sigma^2 estimated as 10737:  log likelihood=-2184.12
## AIC=4382.23   AICc=4382.55   BIC=4409.43
```

Question 7

Use the ARMIA(p,d,q)(P,Q,D)s model from part 8 to forecast for July 1 (which is a Monday). Plot your result.

```r
plot(forecast(hour_seasonal,h=24))
```



Question 8

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

Normally for a model diagnostic we would plot the residuals to see if they exhibit any pattern or run and autocorrelation test (e.g. the Ljung-Box Test). However, since we are focusing on only four points (constituting morning and afternoon rush hour for July 1) we do not have enough residuals for these tests to have much meaning. Instead, we will simply do a sum of squared error test.

```
#the forecast for the day of the week seasonal model
forecast_day<-forecast(day_seasonal,h=24)$mean
#the forecst for the hour of the day seasonal model
forecast_hour<-forecast(hour_seasonal,h=24)$mean
#take only the relevant observations from those forecasts
sub_forecast_day<-forecast_day[c(8, 9, 17, 18)]
sub_forecast_hour<-forecast_hour[c(8, 9, 17, 18)]
#take only the relevant observations from the test data
sub_test<-test[c(8,9,17,18)]
```

The sum of squared error for the model from part 5 is:

```
sum((sub_forecast_day-sub_test)^2)
```

```
## [1] 4604.173
```

The sum of squared error for the model from part 7 is:

```
sum((sub_forecast_hour-sub_test)^2)
```

```
## [1] 415855.7
```

Clearly the model from part 5, which is a model based on a day of the week seasonal model, performs better from the perspective of sum of squared error. Also, a simple eyeball test of the plotted forecast shows that the day of the week seasonal model lines up better with the pattern of the training data and what we see in the actual values for July 1. The hour of the day seasonal model forecast looks like a continuation in shape and diminution of the previous two days, which is a departure from the actual data.