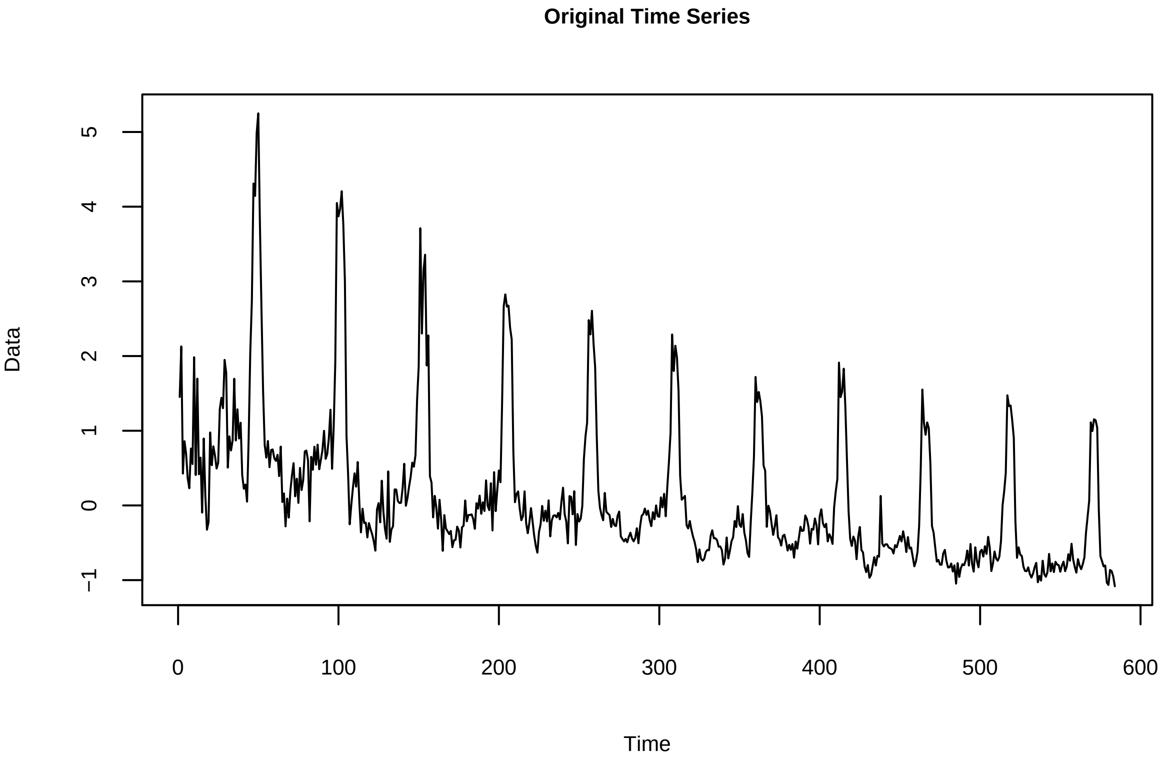
Midterm 2

Haoyang Xu, Dae Woong Ham

11/11/2017

Midterm 2 Report/Data Of Interest: Q3

Plot to see what time series looks like



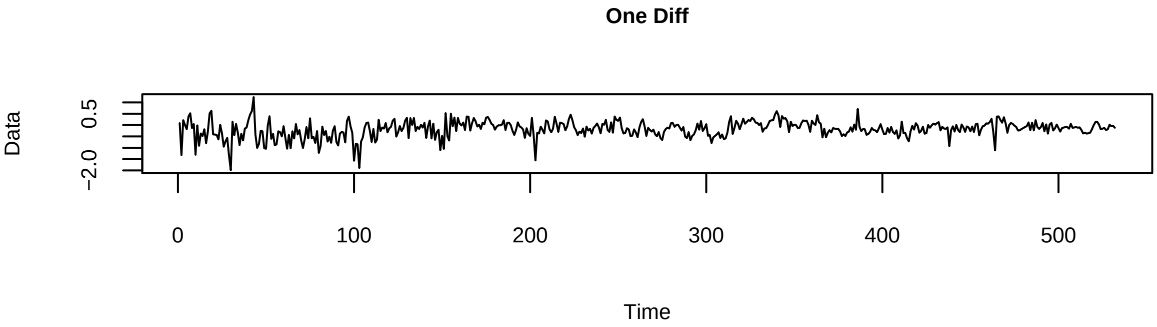
Observations:

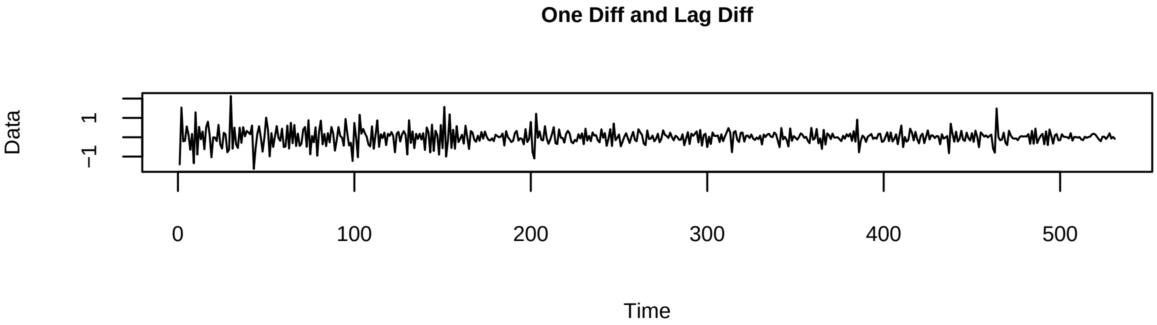
1) There is a clear seasonal trend at lag 52 since we can see every year it reaches it's highest point after around every 52 weeks or around 1 year approximately.
2) There is a very clear linear trend in the beginning of this time series that goes down. However, it is also important to note this linear trend stabilizes in the last few recent years.

Conclusion:

In order to further analyze this time series we must try to fit a model, but this is currently not stationary so we will difference it by lag 52 to deal with the seasonality noted in the first observation and difference it once more to deal clear linear trend noted in the second observation. Plot this differenced data and compare it perhaps by just differencing it by lag 52 to see which one makes our data more stationary.

Decide if Differencing Helps:

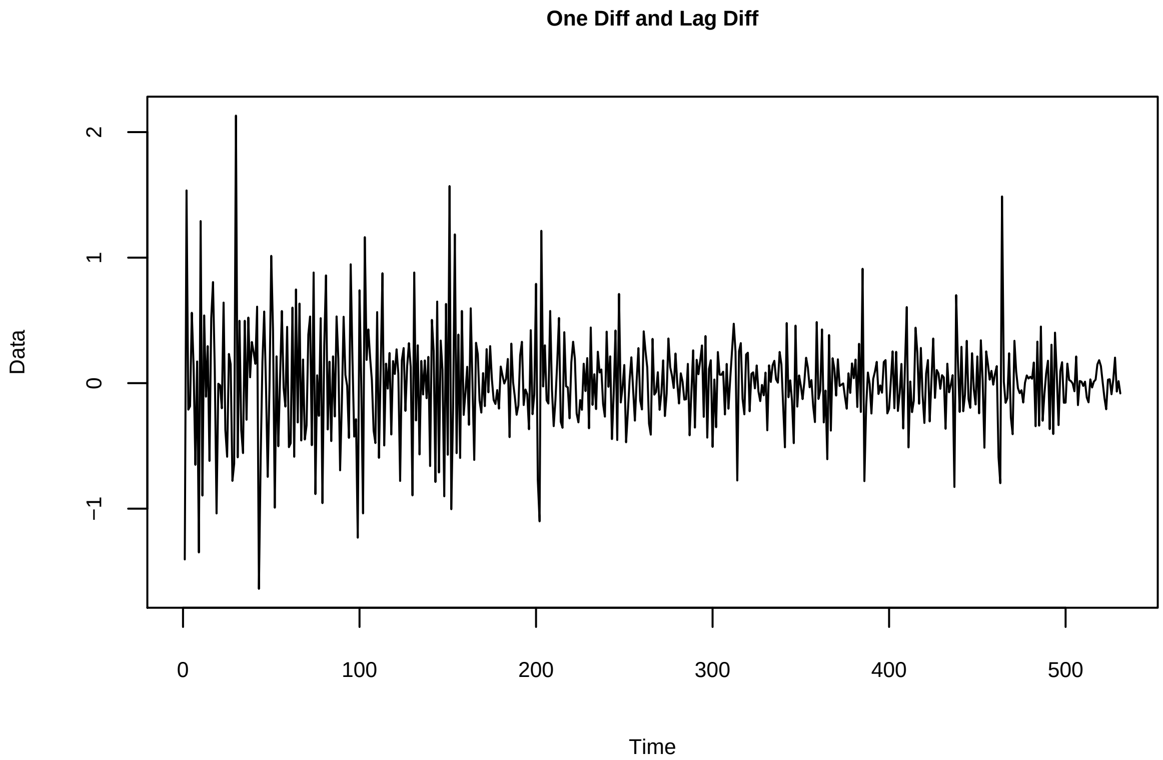


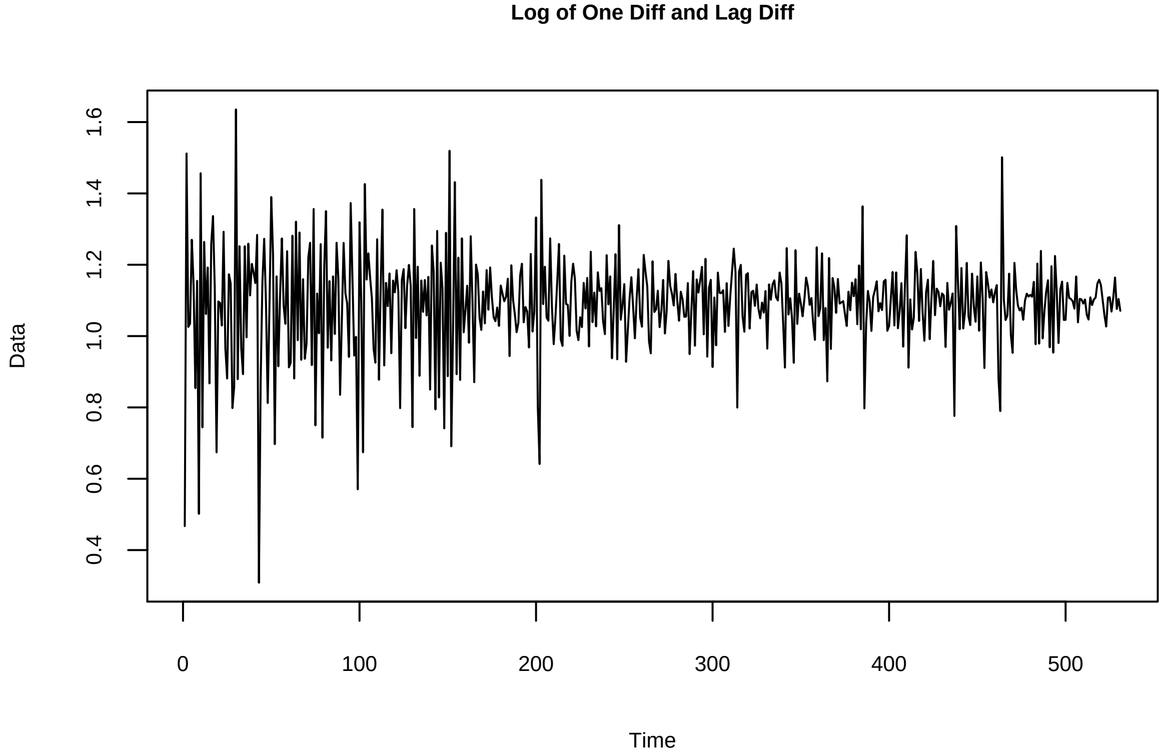


Observation:

1) It is indicative that differencing it once and with lag 52 is much more stationary and what we want to fit a possible ARMA function

2) There is a bit of worry in the beginning of the second plot because there seems to be much stronger noise in the beginning of this time series, and this makes sense too because we can see in the original time series how there is bigger peaks hence greater variance. The way to deal with greater variance is to usually take the log so we will see if the log of this difference will improve it to stationarity more. *Note: Because there are negative values we will take the log of the values shifted up by 3.

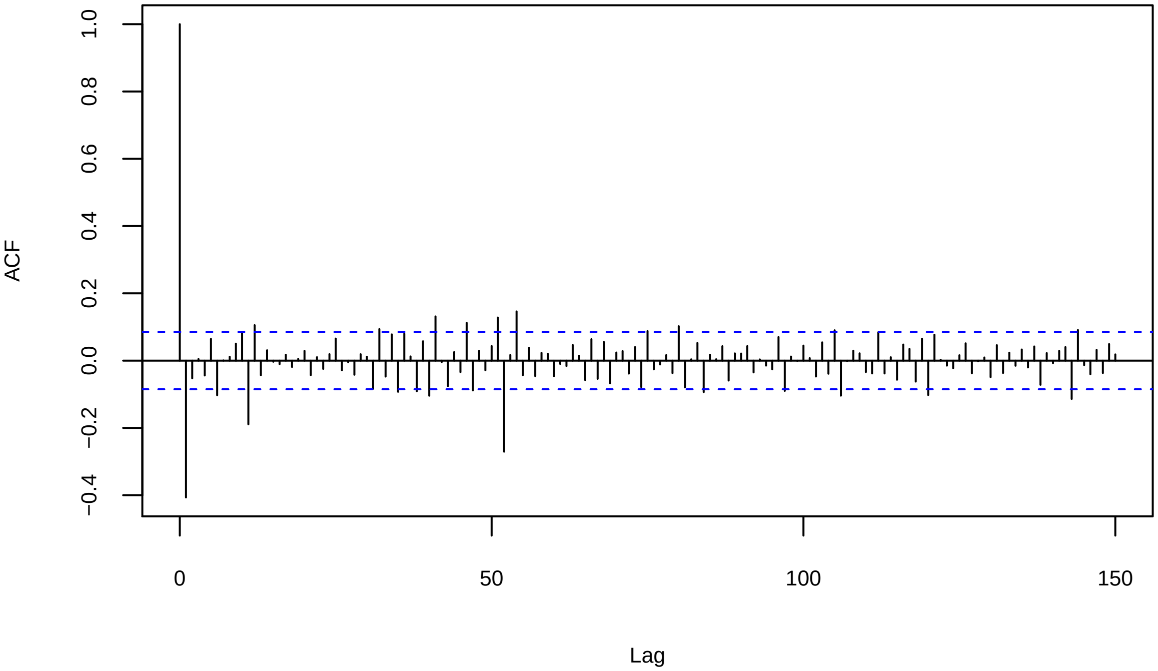


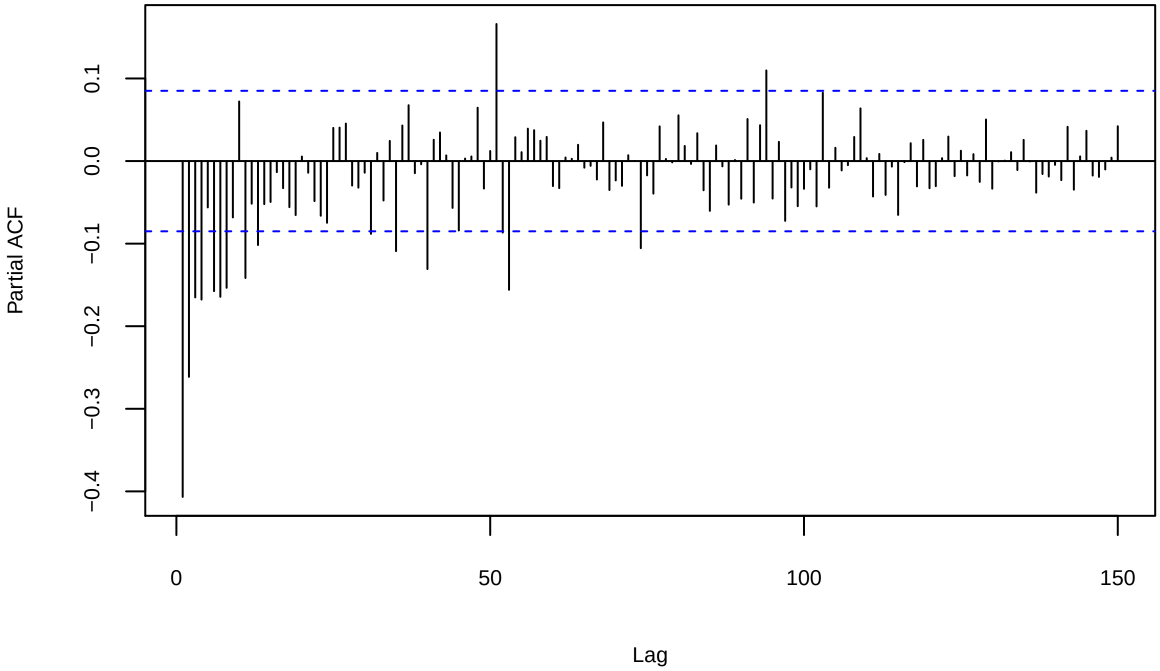


Conclusion:

1) The log did not help that much we will stick with the main one diff and lag diff data to fit models.
2) To see exactly which model might be appropriate, take a look at the ACF and PACF

Getting to Stationarity





Observation:

1) A lot of the ACF values dies after lag 1 but it is very important to realize that at lag 52 and the lags surrounding lag 52 the ACF values are actually greater than just normal white noise ACF values.

2) The PACF also shows a behavior where after lag 2 the pacf values die down to white noise. This is not very evidential and is actually a rough estimate. We can see from the PACF that there are many values after lag 2 that also pop out of the blue bands, but there is clearly a huge drop from lag 2 to lag 3 PACF value so a rough guess of after lag 2 will be used.

Conclusion:

From the two observations it seems that MA(1), AR(2), and perhaps seasonal ARMA with period = 52 is also suitable. I will try out all permutations of these along with MA(1) inside the seasonal or not also. Therefore try out the following models for the original dataset.

**After all this I will try to further diagnose whether which AR(2), AR(3), AR(4), AR(5), is more suitable since the PACF as stated above was not easy to guess where the PACF dies out. My guess was lag 2 but after realizing the AR coefficients are significant I will do further diagnosis on greater AR(p) values

Model Candidates

Note: to undo the differencing put difference of 1 in each seasonal and ARMA component then run the SARIMA fit onto original data

Model a): SARIMA(0, 1, 1), (0, 1, 0)x52 (MA(1) x Seasonal_diff)

Model b): SARIMA(0, 1, 1), (0, 1, 1)x52 (MA(1) x Seasonal_diff(MA(1)))

Model c): SARIMA(2, 1, 0), (0, 1, 0)x52 (AR(2) x Seasonal_diff)

Model d): SARIMA(2, 1, 0), (0, 1, 1)x52 (AR(2) x Seasonal_diff(MA(1)))

Model e): SARIMA(2, 1, 1), (0, 1, 0)x52 (ARMA(2, 1) x Seasonal_diff)

Model f): SARIMA(2, 1, 1), (0, 1, 1)x52 (ARMA(2, 1) x Seasonal_diff(MA(1)))

*Note: If AR(2) turns out to be significant I will do further diagnosis on different AR(p) parameters.

```
##        ma1
## -0.8412863
```

```
##        ma1       sma1
## -0.7309543 -0.3705598
```

```
##        ar1        ar2
## -0.5183898 -0.2631758
```

```
##        ar1        ar2       sma1
## -0.5312759 -0.2050288 -0.4490145
```

```
##         ar1         ar2         ma1
##  0.25773405  0.08358546 -0.95981919
```

```
##        ar1        ar2        ma1       sma1
##  0.2871687  0.1687070 -0.9710702 -0.4053686
```

Observations:

1) All my coefficients including the AR(2) coefficients are significant in the sense that they are far from 0. Therefore, it seems at a very initial glance all these models are potential candidates for further diagnosis.
2) It is also noteworthy to observe that the when the ARMA(2,1) was fitted neither of the MA nor the AR component dominated.

Conclusion:

It seems that because every coefficients are relatively significant we do not yet know whether which model will be the best.

**All in all this requires further diagnostics such as AIC/BIC and cross validation. Note we will not run tsdiag() simply because it takes up too much space to check for 6 different models with each 3 different plots.

Diagnostics - AIC/BIC

```
##        AIC      BIC
## a 368.5938 377.1433
## b 321.2389 334.0631
## c 423.6292 436.4535
## d 353.7674 370.8665
## e 343.7007 360.7998
## f 285.5767 306.9505
```

Conclusion:

1) The AIC values has a very solid winner of model f, where it has by far the lowest AIC value. However, it is important to realize that AIC does not penalize the number of paramters as much as BIC, and therefore model f), having the most number of parameters, could just be over fitting.
2) The BIC values suggest that model f is actually still the best. This multiple attestion for both AIC and BIC telling us model f) is the best can give us perhaps the first hypothesis for model selection. Although AIC and BIC are not enough to go on for model selection, the fact that model f) was a clear winner in both cases shows that model f) is our top candidate.

**Since AIC and BIC are not satisfactory measures of model selection we will do further diagnoistics to decide whether model f is really the best using cross validation.

Diagnoistics

```
##           Model A  Model B  Model C Model D  Model E  Model F
## CV Error 19.65223 18.87806 17.76692 15.8176 18.63931 7.078616
```

Note on Cross Validation: The way I did my cross validation was by first making sure that each testing data had 104 data points left aside and I created 4 out of sample predictions where each training data grew bigger for each experiment. Moreover, I made sure each of my training data were actually disjoint from each other to give full power to each cross validation. It turned out that model d and model f were very similar in the cross validation error so I tried these two models again with a different out of sample size. Specifically I chose to test these two models on only more recent years, by making them predict only the last 4 years rather than the last 8 years.

```
## [1] "Model D CV Error More Recent: 8.27340932333264"
```

```
## [1] "Model F CV Error More Recent: 7.07861647566474"
```

Comment: It seems that the best model is model f) (after repeating it with more experiments) which has all the MA(1) and AR(2) component. This indeed matches our initial hypothesis from the AIC and BIC diagnostics. However, as stated above the AR(2) was a pure guess above from the PACF and actually could have been AR(3) or AR(4), or higher since those lag pacf values were also out of the blue bands. Now to make sure each AR coefficients are indeed significant and cannot be reduced or expanded I will check a few values for different AR(p) and see how they perform relative to the AR(2) by using cross validation.

So try out the same form of model e) but with AR(1), AR(2), AR(3), AR(4), AR(5) and see which gives the best cross validation error

Further Diagnoistics - Cross Validation (Choosing the AR)

```
##               AR1      AR2      AR3      AR4      AR5
## CV Error 16.19871 15.89597 15.88661 15.93655 15.99381
```

Comment: So it turns out the initial AR(2) although a close guess was not the best fit for our original time series. A better model would be the AR(3) as the cross validation tells us. It is also important to note that if we changed AR(3) and increased the parameter even more to AR(4) and AR(5) this just shows how our errors are getting bigger so AR(3) was the right place to stop.

Now that we have diagnosed almost all possible models the only thing we have not checked is whether the MA(1) was fully justified. It seems from the ACF that MA(2) is not a possibility because the ACF at lag2 died down extremely steeply and very convincingly. Therefore, unlike the PACF where there is some confusion between which AR(p) is necessary, there is no theoretical basis to justify MA(2). However, computing power is cheap and since we want to 100% make sure MA(1) is better let's do a final comparison between our final last model compared to that same model but with MA(2).

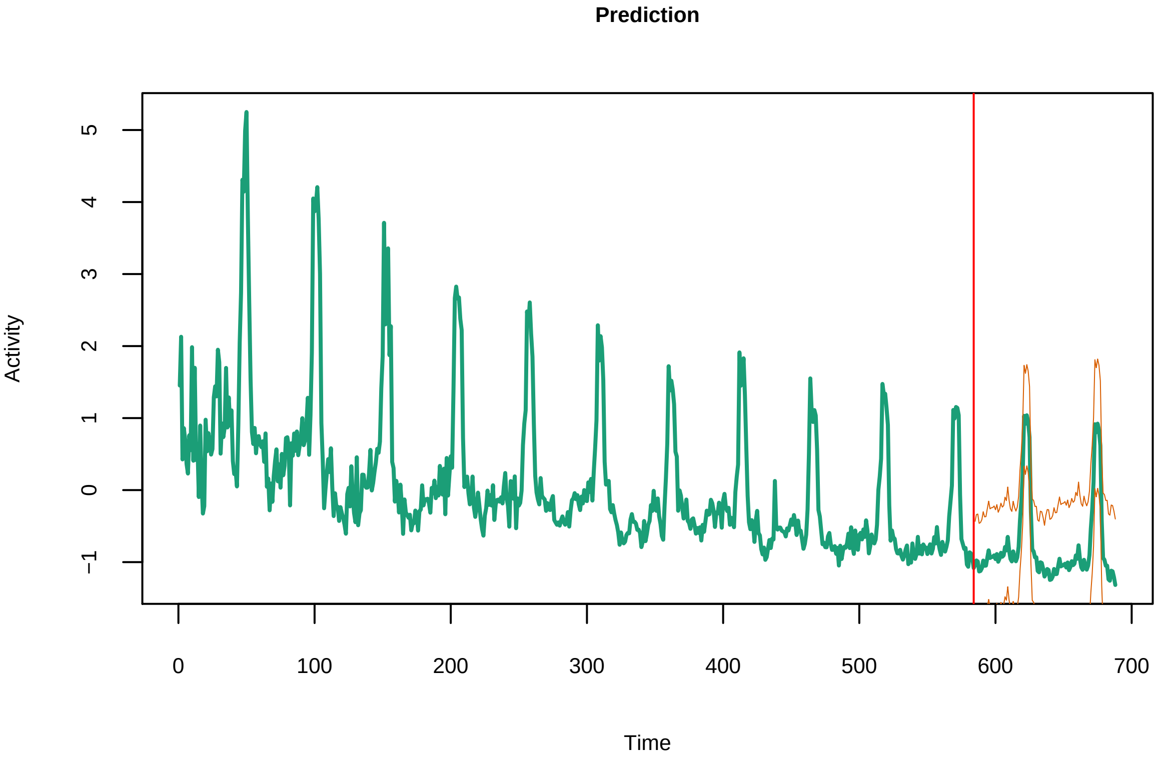
Last Diagnoistics - Cross Validation (Choosing MA(1) or MA(2))

```
## [1] "MA(1) CV Error: 15.8866099883747"
```

```
## [1] "MA(2) CV Error: 19.0353620383772"
```

It is just as we expected MA(2) does an absolutely terrible job compared to MA(1) of our final form. With all this diagnostics, the model that is most appropriate is the: SARIMA(3, 1, 1), (0, 1, 1)x52

**Now that we have our model let's predict and plot our predictions.



Appendix

Q1

```r
library(RColorBrewer)
pal <- brewer.pal(8, "Dark2")

#Helper functions
plot_func = function(data, xlab = "Time", ylab = "Data", type = "l", main = "Graph") {
  plot(x = 1:length(data), y = data, type = type, xlab = xlab, ylab = ylab, main = main)
}

cv_error <- function(data, p = 0, q = 0, d = 1, p1 = 0, q1 = 0, d1 = 1, s = 52, l = 104, n = 4) {
  set.seed(5)
  len = length(data)
  sum_error = vector(length = 0)
  k = len - l*n
  for (i in 1:n) {
    initial = k + (i-1)*l
    final = initial + l - 1
    train_data = data[1:(initial - 1)]
    arima.fit = arima(x = train_data, order = c(p, d, q), seasonal = list(order = c(p1, d1, q1), period
    predic = predict(arima.fit, n.ahead = l)$pred
    actual = data[initial:final]
    sum_error[i] = sum((predic - actual)^2)
  }
  return(sum(sum_error)/n)
}

plot_predic = function(model, data) {
  predic = predict(model, n.ahead = 104)
  new_data = c(data, predic$pred)
  lower_bound = predic$pred - 1.96*predic$se
  upper_bound = predic$pred + 1.96*predic$se
  mfrow = c(1,1)
  plot(x = 1:length(new_data), y = new_data, type = "l", col = pal[1], lwd = 2, ylab = "Activity", xlab
  abline(v = length(data), col = "red")
  lines(lower_bound, col=pal[2], lwd = 0.5)
  lines(upper_bound, col=pal[2], lwd = 0.5)
}
```

Data Extraction and Cleanup

```r
q1_df = read.csv("~/Downloads/q1_train.csv", header = T)
q1_df[, 2] = q1_df[, 2]
q1_df[, 3] = 1:length(q1_df[, 1])
colnames(q1_df)[3] = "Time"
data = q1_df[, 2]
t = q1_df[, 3]
```

Plot to see what time series looks like

```r
par(mfrow = c(1, 1))
plot_func(data, main = "Original Time Series")
```

Decide if Differencing Helps:

```r
par(mfrow = c(2, 1))
annual_diff = diff(q1_df[, 2], lag = 52)
diff_annual_diff = diff(annual_diff)
plot(x = 1:length(annual_diff), y = annual_diff, type = "l", ylab = "Data", xlab = "Time", main = "One 
plot(x = 1:length(diff_annual_diff), y = diff_annual_diff, type = "l",ylab = "Data", xlab = "Time", mai
```

Check ACF/PACF for Model Candidates

```r
new_data = diff_annual_diff
acf(new_data, lag.max = 150)
pacf(new_data, lag.max = 150)
```

Fitting our model

```r
data = q1_df[, 2]
q1_modela = arima(x = data, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 0),
    period = 52))
q1_modelb = arima(x = data, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 52))
q1_modelc = arima(x = data, order = c(4, 1, 0), seasonal = list(order = c(0, 1, 0),
    period = 52))
q1_modeld = arima(x = data, order = c(4, 1, 0), seasonal = list(order = c(0, 1, 1),
    period = 52))
q1_modele = arima(x = data, order = c(4, 1, 1), seasonal = list(order = c(0, 1, 0),
    period = 52))
q1_modelf = arima(x = data, order = c(4, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 52))
coef(q1_modela); coef(q1_modelb); coef(q1_modelc); coef(q1_modeld); coef(q1_modele); coef(q1_modelf)
```

Diagnostics - AIC/BIC

```r
aic = c(AIC(q1_modela), AIC(q1_modelb), AIC(q1_modelc), AIC(q1_modeld), AIC(q1_modele), AIC(q1_modelf))
bic = c(BIC(q1_modela), BIC(q1_modelb), BIC(q1_modelc), BIC(q1_modeld), BIC(q1_modele), BIC(q1_modelf))

matrix(c(aic, bic), nrow = 6, dimnames = list(c("a", "b", "c", "d", "e", "f"), c("AIC", "BIC")))
```

Diagonistics - Cross Validation

```
error_a = cv_error(data = data, q = 1)
error_b = cv_error(data = data, q = 1, q1 = 1)
error_c = cv_error(data = data, p = 4)
error_d = cv_error(data = data, p = 4, q1 = 1)
error_e = cv_error(data = data, p = 4, q = 1)
error_f = cv_error(data = data, p = 4, q = 1, q1 = 1)
matrix(c(error_a, error_b, error_c, error_d, error_e, error_f), nrow = 1, dimnames = list(c("CV Error")
```

General Function to Plot

```
winner_model = q1_modela
par(mfrow = c(1,1))
plot_predic(winner_model, data)
pre = predict(winner_model, n.ahead = 104)
p = pre$pred
se = pre$se
lower = p - 1.96*se
upper = p + 1.96*se
pred_df = data.frame(lower, p, upper)
write.csv(pred_df, file = "Q1_DaeWoong_Ham_26222439.txt", row.names = F)
```

Q2

Data Extraction and Cleanup

```
q2_df = read.csv("~/Downloads/q2_train.csv", header = T)
q2_df[, 2] = q2_df[, 2]
q2_df[, 3] = 1:length(q2_df[, 1])
colnames(q2_df)[3] = "Time"
data = q2_df[, 2]
t = q2_df[, 3]
```

Plot to see what time series looks like

```
par(mfrow = c(1,1))
plot_func(data, main = "Original Time Series")
```

Decide if Differencing Helps

```
diff_52 = diff(data, lag = 52)
diff_52_1 = diff(diff_52)
par(mfrow = c(2, 1))
plot_func(diff_52, ylab = "Diff(data, 52)", main = "Differenced with lag")
plot_func(diff_52_1, ylab = "Diff(Diff(data, 52))", main = "Differenced once and with lag")
```

Check ACF/PACF for Model Candidates

```
new_data = diff_52_1
par(mfrow = c(2, 1))
acf(new_data, lag.max = 150)
pacf(new_data, lag.max = 150)
```

Fitting Our Models

```
q2_modela = arima(x = data, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 0),
    period = 52))
q2_modelb = arima(x = data, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 52))
q2_modelc = arima(x = data, order = c(3, 1, 0), seasonal = list(order = c(0, 1, 0),
    period = 52))
q2_modeld = arima(x = data, order = c(3, 1, 0), seasonal = list(order = c(0, 1, 1),
    period = 52))
q2_modele = arima(x = data, order = c(3, 1, 1), seasonal = list(order = c(0, 1, 0),
    period = 52))
q2_modelf = arima(x = data, order = c(3, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 52))
coef(q2_modela); coef(q2_modelb); coef(q2_modelc); coef(q2_modeld); coef(q2_modele); coef(q2_modelf)
```

Diagnostics - AIC/BIC

```
aic = c(AIC(q2_modela), AIC(q2_modelb), AIC(q2_modelc), AIC(q2_modeld), AIC(q2_modele), AIC(q2_modelf))
bic = c(BIC(q2_modela), BIC(q2_modelb), BIC(q2_modelc), BIC(q2_modeld), BIC(q2_modele), BIC(q2_modelf))

matrix(c(aic, bic), nrow = 6, dimnames = list(c("a", "b", "c", "d", "e", "f"), c("AIC", "BIC")))
```

Diagonistics - Cross Validation

```
error_a = cv_error(data = data, q = 1)
error_b = cv_error(data = data, q = 1, q1 = 1)
error_c = cv_error(data = data, p = 3)
error_d = cv_error(data = data, p = 3, q1 = 1)
error_e = cv_error(data = data, p = 3, q = 1)
error_f = cv_error(data = data, p = 3, q = 1, q1 = 1)

matrix(c(error_a, error_b, error_c, error_d, error_e, error_f), nrow = 1, dimnames = list(c("CV Error")
```

Further Diagonistics - Cross Validation (Choosing the AR)

```
error_ar1 = cv_error(data = data, p = 1, q = 1)
error_ar2 = cv_error(data = data, p = 2, q = 1)
error_ar3 = cv_error(data = data, p = 3, q = 1)
error_ar4 = cv_error(data = data, p = 4, q = 1)
error_ar5 = cv_error(data = data, p = 5, q = 1)
```

```
matrix(c(error_ar1, error_ar2, error_ar3, error_ar4, error_ar5), nrow = 1, dimnames = list(c("CV Error")
```

General Function to Plot

```
winner_model = q2_modelf
par(mfrow = c(1,1))
plot_predic(winner_model, data)
pre = predict(winner_model, n.ahead = 104)
p = pre$pred
se = pre$se
lower = p - 1.96*se
upper = p + 1.96*se
pred_df = data.frame(lower, p, upper)
write.csv(pred_df, file = "Q2_DaeWoong_Ham_26222439.txt", row.names = F)
```

Q3(Report)

Data Extraction and Cleanup

```
q3_df = read.csv("~/Downloads/q3_train.csv", header = T)
q3_df[, 2] = q3_df[, 2]
q3_df[, 3] = 1:length(q3_df[, 1])
colnames(q3_df)[3] = "Time"
data = q3_df[, 2]
t = q3_df[, 3]
```

Plot to see what time series looks like

```
par(mfrow = c(1, 1))
plot_func(data, main = "Original Time Series")
```

Decide if Differencing Helps:

```
annual_diff = diff(q3_df[, 2], lag = 52)
diff_annual_diff = diff(annual_diff)
par(mfrow = c(2, 1))
plot(x = 1:length(annual_diff), y = annual_diff, type = "l", ylab = "Data", xlab = "Time", main = "One 
plot(x = 1:length(diff_annual_diff), y = diff_annual_diff, type = "l",ylab = "Data", xlab = "Time", mai
```

```
plot(x = 1:length(diff_annual_diff), y = diff_annual_diff, type = "l",ylab = "Data", xlab = "Time", mai
plot(x = 1:length(diff_annual_diff), y = log(diff_annual_diff + 3), type = "l",ylab = "Data", xlab = "T
```

Getting to Stationarity

```
acf(diff_annual_diff, lag.max = 150, main = NA, cex.lab = 0.7, cex.axis = 0.7)
pacf(diff_annual_diff, lag.max = 150, main = NA, cex.lab = 0.7, cex.axis = 0.7)
```

Fitting Our Models

```
q3_modela = arima(x = data, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 0),
    period = 52))
q3_modelb = arima(x = data, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 52))
q3_modelc = arima(x = data, order = c(2, 1, 0), seasonal = list(order = c(0, 1, 0),
    period = 52))
q3_modeld = arima(x = data, order = c(2, 1, 0), seasonal = list(order = c(0, 1, 1),
    period = 52))
q3_modele = arima(x = data, order = c(2, 1, 1), seasonal = list(order = c(0, 1, 0),
    period = 52))
q3_modelf = arima(x = data, order = c(2, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 52))
coef(q3_modela); coef(q3_modelb); coef(q3_modelc); coef(q3_modeld); coef(q3_modele); coef(q3_modelf)
```

Diagnostics - AIC/BIC

```
aic = c(AIC(q3_modela), AIC(q3_modelb), AIC(q3_modelc), AIC(q3_modeld), AIC(q3_modele), AIC(q3_modelf))
bic = c(BIC(q3_modela), BIC(q3_modelb), BIC(q3_modelc), BIC(q3_modeld), BIC(q3_modele), BIC(q3_modelf))

matrix(c(aic, bic), nrow = 6, dimnames = list(c("a", "b", "c", "d", "e", "f"), c("AIC", "BIC")))
```

Diagonistics - Cross Validation

```
error_a = cv_error(data = data, q = 1)
error_b = cv_error(data = data, q = 1, q1 = 1)
error_c = cv_error(data = data, p = 2)
error_d = cv_error(data = data, p = 2, q1 = 1)
error_e = cv_error(data = data, p = 2, q = 1)
error_f = cv_error(data = data, p = 2, q = 1, q1 = 1, n = 2)
matrix(c(error_a, error_b, error_c, error_d, error_e, error_f), nrow = 1, dimnames = list(c("CV Error")
```

```
paste("Model D CV Error More Recent:", cv_error(data = data, p = 2, q1 = 1, n = 2))
paste("Model F CV Error More Recent:", cv_error(data = data, p = 2, q = 1, q1 = 1, n = 2))
```

Further Diagonistics - Cross Validation (Choosing the AR)

```
error_ar1 = cv_error(data = data, p = 1, q = 1, q1 = 1)
error_ar2 = cv_error(data = data, p = 2, q = 1, q1 = 1)
error_ar3 = cv_error(data = data, p = 3, q = 1, q1 = 1)
error_ar4 = cv_error(data = data, p = 4, q = 1, q1 = 1)
error_ar5 = cv_error(data = data, p = 5, q = 1, q1 = 1)

matrix(c(error_ar1, error_ar2, error_ar3, error_ar4, error_ar5), nrow = 1, dimnames = list(c("CV Error"
```

Last Diagnoistics - Cross Validation (Choosing MA(1) or MA(2))

```r
paste("MA(1) CV Error:", error_ar3)
paste("MA(2) CV Error:", cv_error(data = data, p = 3, q = 2, q1 = 1))
```

General Function to Plot

```r
winner_model = arima(x = data, order = c(3, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 52))
par(mfrow = c(1,1))
plot_predic(winner_model, data)
pre = predict(winner_model, n.ahead = 104)
p = pre$pred
se = pre$se
lower = p - 1.96*se
upper = p + 1.96*se
pred_df = data.frame(lower, p, upper)
write.csv(pred_df, file = "Q3_DaeWoong_Ham_26222439.txt", row.names = F)
```

Q4

Data Extraction and Cleanup

```r
q4_df = read.csv("~/Downloads/q4_train.csv", header = T)
q4_df[, 2] = q4_df[, 2]
q4_df[, 3] = 1:length(q4_df[, 1])
colnames(q4_df)[3] = "Time"
data = q4_df[, 2]
t = q4_df[, 3]
```

Plot to see what time series looks like

```r
par(mfrow = c(1, 1))
plot_func(data, main = "Original Time Series")
```

Decide if Differencing Helps:

```r
annual_diff = diff(q4_df[, 2], lag = 52)
diff_annual_diff = diff(annual_diff)
par(mfrow = c(2, 1))
plot(x = 1:length(annual_diff), y = annual_diff, type = "l", ylab = "Data", xlab = "Time", main = "One
plot(x = 1:length(diff_annual_diff), y = diff_annual_diff, type = "l",ylab = "Data", xlab = "Time", mai
```

Check ACF/PACF for Model Candidates

```r
new_data = diff_annual_diff
acf(new_data, lag.max = 150)
pacf(new_data, lag.max = 150)
```

Fitting Our Models

```r
data = q4_df[, 2]
q4_modela = arima(x = data, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 0),
    period = 52))
q4_modelb = arima(x = data, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 52))
q4_modelc = arima(x = data, order = c(4, 1, 0), seasonal = list(order = c(0, 1, 0),
    period = 52))
q4_modeld = arima(x = data, order = c(4, 1, 0), seasonal = list(order = c(0, 1, 1),
    period = 52))
q4_modele = arima(x = data, order = c(4, 1, 1), seasonal = list(order = c(0, 1, 0),
    period = 52))
q4_modelf = arima(x = data, order = c(4, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 52))
coef(q4_modela); coef(q4_modelb); coef(q4_modelc); coef(q4_modeld); coef(q4_modele); coef(q4_modelf)
```

Diagnostics - AIC/BIC

```r
aic = c(AIC(q4_modela), AIC(q4_modelb), AIC(q4_modelc), AIC(q4_modeld), AIC(q4_modele), AIC(q4_modelf))
bic = c(BIC(q4_modela), BIC(q4_modelb), BIC(q4_modelc), BIC(q4_modeld), BIC(q4_modele), BIC(q4_modelf))

matrix(c(aic, bic), nrow = 6, dimnames = list(c("a", "b", "c", "d", "e", "f"), c("AIC", "BIC")))
```

Diagnoistics - Cross Validation

```r
error_a = cv_error(data = data, q = 1)
error_b = cv_error(data = data, q = 1, q1 = 1)
error_c = cv_error(data = data, p = 4)
error_d = cv_error(data = data, p = 4, q1 = 1)
error_e = cv_error(data = data, p = 4, q = 1)
error_f = cv_error(data = data, p = 4, q = 1, q1 = 1)

matrix(c(error_a, error_b, error_c, error_d, error_e, error_f), nrow = 1, dimnames = list(c("CV Error")
```

General function to plot

```r
#winner model f
winner_model = q4_modelb

par(mfrow = c(1,1))
plot_predic(winner_model, data)
pre = predict(winner_model, n.ahead = 104)
```

```
p = pre$pred
se = pre$se
lower = p - 1.96*se
upper = p + 1.96*se
pred_df = data.frame(lower, p, upper)
write.csv(pred_df, file = "Q4_DaeWoong_Ham_26222439.txt", row.names = F)
```

Q5

Data Extraction and Cleanup

```
q5_df = read.csv("~/Downloads/q5_train.csv", header = T)
q5_df[, 2] = q5_df[, 2]
q5_df[, 3] = 1:length(q5_df[, 1])
colnames(q5_df)[3] = "Time"
t = q5_df[, 3]
data = q5_df[, 2]
```

Plot to see what time series looks like

```
par(mfrow = c(1, 1))
plot_func(data, main = "Original Time Series")
```

Decide if Differencing Helps:

```
diff_52 = diff(data, lag = 52)
diff_52_1 = diff(diff_52)
par(mfrow = c(2, 1))
plot_func(diff_52, ylab = "Diff(data, 52)", main = "Differenced with lag")
plot_func(diff_52_1, ylab = "Diff(Diff(data, 52))", main = "Differenced once and with lag")
```

Check ACF/PACF for Model Candidates

```
new_data = diff_52_1
par(mfrow = c(2, 1))
acf(new_data, lag.max = 150)
pacf(new_data, lag.max = 150)
```

Fitting Our Models

```
q5_modela = arima(x = data, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 0),
    period = 52))
q5_modelb = arima(x = data, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 52))
q5_modelc = arima(x = data, order = c(5, 1, 0), seasonal = list(order = c(0, 1, 0),
    period = 52))
```

```
q5_modeld = arima(x = data, order = c(5, 1, 0), seasonal = list(order = c(0, 1, 1),
    period = 52))
q5_modele = arima(x = data, order = c(5, 1, 1), seasonal = list(order = c(0, 1, 0),
    period = 52))
q5_modelf = arima(x = data, order = c(5, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 52))

#Compare coefficients
a = coef(q5_modela)
b = coef(q5_modelb)
c_coef = coef(q5_modelc)
d = coef(q5_modeld)
e = coef(q5_modele)
f = coef(q5_modelf)
a; b; c_coef; d; e; f
```

Diagnostics - AIC/BIC

```
aic = c(AIC(q5_modela), AIC(q5_modelb), AIC(q5_modelc), AIC(q5_modeld), AIC(q5_modele), AIC(q5_modelf))
bic = c(BIC(q5_modela), BIC(q5_modelb), BIC(q5_modelc), BIC(q5_modeld), BIC(q5_modele), BIC(q5_modelf))

matrix(c(aic, bic), nrow = 6, dimnames = list(c("a", "b", "c", "d", "e", "f"), c("AIC", "BIC")))
```

Diagnoistics - Cross Validation

```
error_a = cv_error(data = data, q = 1)
error_b = cv_error(data = data, q = 1, q1 = 1)
error_c = cv_error(data = data, p = 5)
error_d = cv_error(data = data, p = 5, q1 = 1)
error_e = cv_error(data = data, p = 5, q = 1)
error_f = cv_error(data = data, p = 5, q = 1, q1 = 1)

matrix(c(error_a, error_b, error_c, error_d, error_e, error_f), nrow = 1, dimnames = list(c("CV Error")
```

General Function to Plot

```
winner_model = q5_modela
par(mfrow = c(1,1))
plot_predic(winner_model, data)
pre = predict(winner_model, n.ahead = 104)
p = pre$pred
se = pre$se
lower = p - 1.96*se
upper = p + 1.96*se
pred_df = data.frame(lower, p, upper)
write.csv(pred_df, file = "Q5_DaeWoong_Ham_26222439.txt", row.names = F)
```