Project 1 Group 5

Aaron Zalki, Mia Siracusa, Lidiia Tronina, John Suh, Henry Vasquez

6/21/2020

In the first part of this project, we want to perform fundamental analysis of the data.

### Import and Clean Data

First step is to import the data from excel. When we imported the excel file, R was not reading the dates correctly, so we converted the first column in the date format.

## SeriesInd group Var01   
## Min. :2011-05-06 Length:10572 Min. : 9.03   
## 1st Qu.:2013-01-29 Class :character 1st Qu.: 23.10   
## Median :2014-11-03 Mode :character Median : 38.44   
## Mean :2014-11-01 Mean : 46.98   
## 3rd Qu.:2016-08-05 3rd Qu.: 66.78   
## Max. :2018-05-01 Max. :195.18   
## NA's :854   
## Var02 Var03 Var05 Var07   
## Min. : 1339900 Min. : 8.82 Min. : 8.99 Min. : 8.92   
## 1st Qu.: 12520675 1st Qu.: 22.59 1st Qu.: 22.91 1st Qu.: 22.88   
## Median : 21086550 Median : 37.66 Median : 38.05 Median : 38.05   
## Mean : 37035741 Mean : 46.12 Mean : 46.55 Mean : 46.56   
## 3rd Qu.: 42486700 3rd Qu.: 65.88 3rd Qu.: 66.38 3rd Qu.: 66.31   
## Max. :480879500 Max. :189.36 Max. :195.00 Max. :189.72   
## NA's :842 NA's :866 NA's :866 NA's :866

Next step is to examine the data before converting it to a time series to see if there is any missing data or other problems with the data. By doing this we discovered a few problems that needed to be dealt with:

1. All variables except date are NA after 10/13/17.
2. There are several NA that are in the middle of the data set.
3. There are outliers in data that are far above the normal.
4. The date field has only workdays (Monday through Friday).

We removed all blank observations after 10/13/17. Other NA’s data were imputed using the median since the number of missing values was so small. We used median for each variable and group separately.

## SeriesInd group Var01   
## Min. :2011-05-06 Length:9732 Min. : 9.03   
## 1st Qu.:2012-12-10 Class :character 1st Qu.: 23.16   
## Median :2014-07-25 Mode :character Median : 38.40   
## Mean :2014-07-23 Mean : 46.98   
## 3rd Qu.:2016-03-01 3rd Qu.: 66.80   
## Max. :2017-10-13 Max. :195.18   
## Var02 Var03 Var05 Var07   
## Min. : 1339900 Min. : 8.82 Min. : 8.99 Min. : 8.92   
## 1st Qu.: 12521025 1st Qu.: 22.63 1st Qu.: 22.93 1st Qu.: 22.92   
## Median : 21086550 Median : 37.62 Median : 38.01 Median : 37.98   
## Mean : 37031871 Mean : 46.12 Mean : 46.55 Mean : 46.56   
## 3rd Qu.: 42464900 3rd Qu.: 65.97 3rd Qu.: 66.43 3rd Qu.: 66.39   
## Max. :480879500 Max. :189.36 Max. :195.00 Max. :189.72

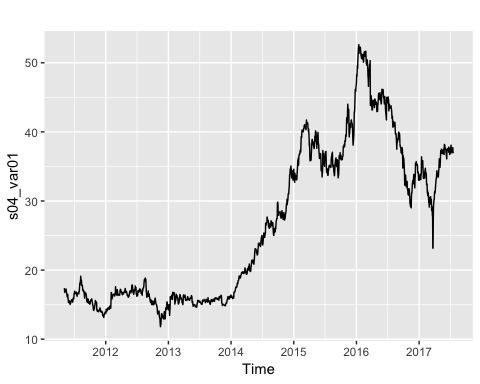
Time series objects were created for each group and variable separately. We used 261 days as our frequency, which is the approximate number of weekdays in a year.

### Forecast

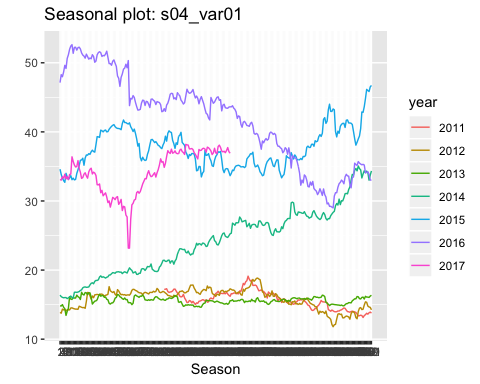
For each group and variable, we want to run at least 2 models, and see which has the better performance. Before running any models we will check the ACF and PACF plots, seasonal plot, time series decomposition plot to see what it can recommend for what type of model they suggest might be most appropriate.

### S04 – Forecast Var01

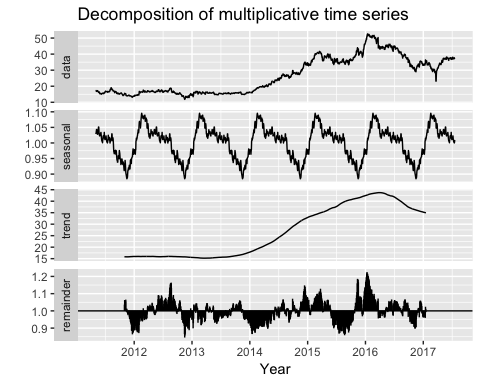
s04\_var01 <- data1%>%  
 filter(group =='S04' ) %>%  
 select(SeriesInd, Var01)  
s04\_var01<- ts(s04\_var01[,2], start = c(2011,88), frequency = 261)  
autoplot(s04\_var01)



ggseasonplot(s04\_var01)

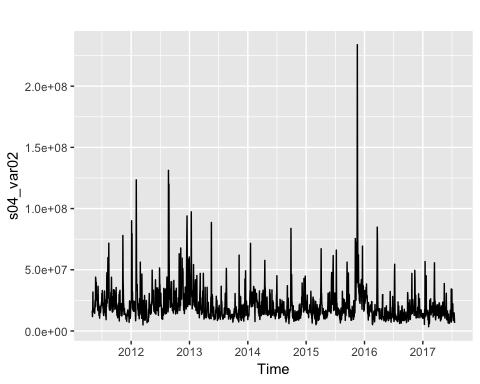


s04\_var01 %>% decompose(type = "multiplicative") %>%  
 autoplot() + xlab("Year")

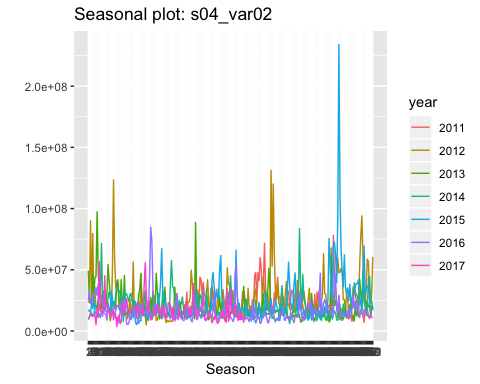


### S04 – Forecast Var02

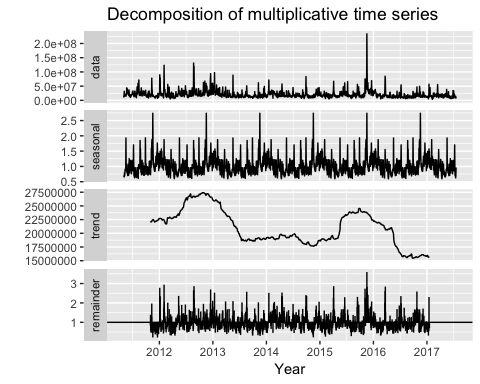
s04\_var02 <- data1%>%  
 filter(group =='S04' ) %>%  
 select(SeriesInd, Var02)  
s04\_var02<- ts(s04\_var02[,2], start = c(2011,88), frequency = 261)  
autoplot(s04\_var02)



ggseasonplot(s04\_var02)

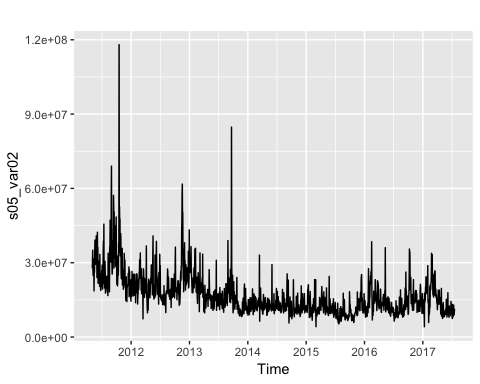


s04\_var02 %>% decompose(type = "multiplicative") %>%  
 autoplot() + xlab("Year")

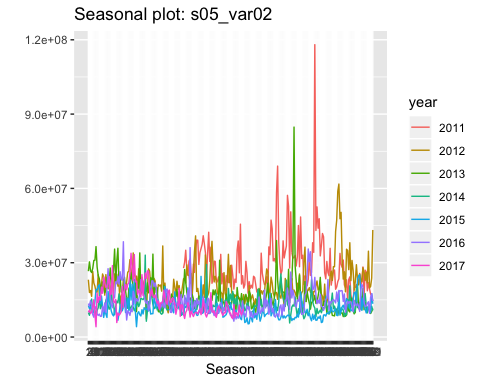


### S05 – Forecast Var02

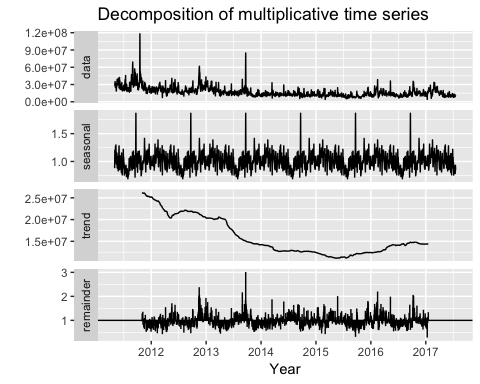
s05\_var02 <- data1%>%  
 filter(group =='S05' ) %>%  
 select(SeriesInd, Var02)  
s05\_var02<- ts(s05\_var02[,2], start = c(2011,88), frequency = 261)  
autoplot(s05\_var02)



ggseasonplot(s05\_var02)

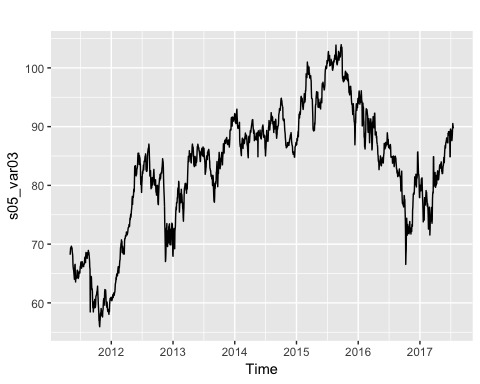


s05\_var02 %>% decompose(type = "multiplicative") %>%  
 autoplot() + xlab("Year")

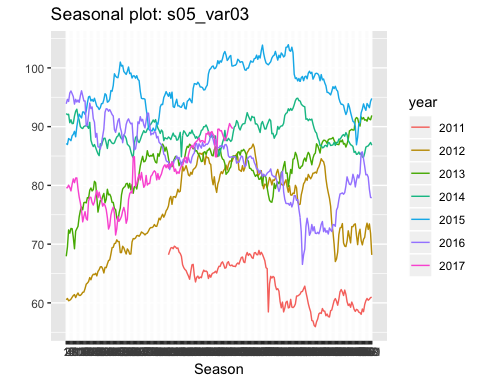


### S05 – Forecast Var03

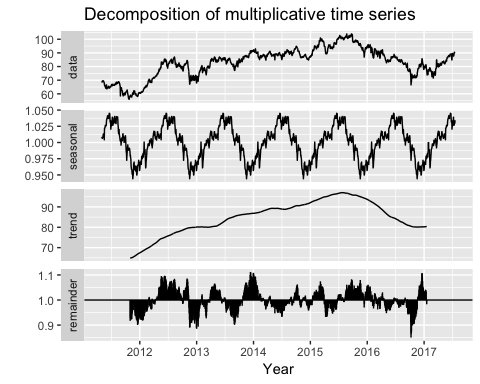
s05\_var03 <- data1%>%  
 filter(group =='S05' ) %>%  
 select(SeriesInd, Var03)  
s05\_var03<- ts(s05\_var03[,2], start = c(2011,88), frequency = 261)  
autoplot(s05\_var03)



ggseasonplot(s05\_var03)

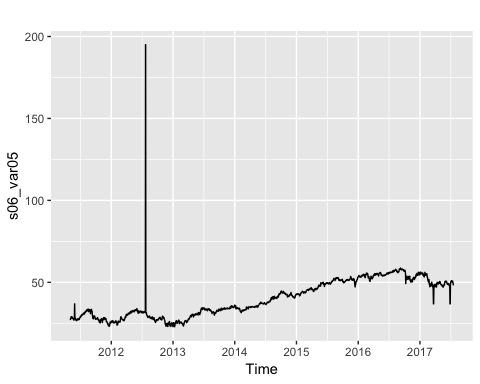


s05\_var03 %>% decompose(type = "multiplicative") %>%  
 autoplot() + xlab("Year")

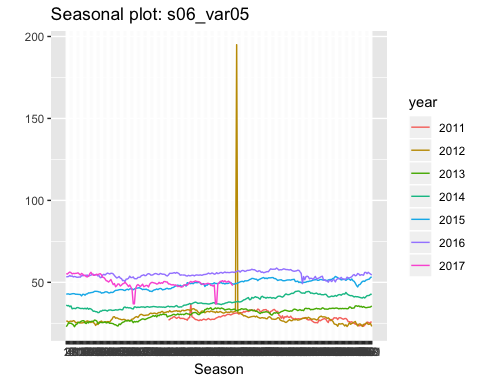


### S06 – Forecast Var05

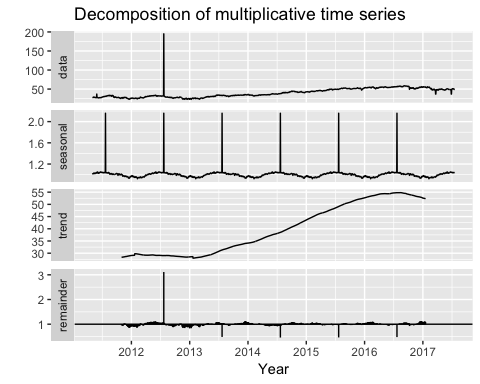
s06\_var05 <- data1%>%  
 filter(group =='S06' ) %>%  
 select(SeriesInd, Var05)  
s06\_var05<- ts(s06\_var05[,2], start = c(2011,88), frequency = 261)  
autoplot(s06\_var05)



ggseasonplot(s06\_var05)

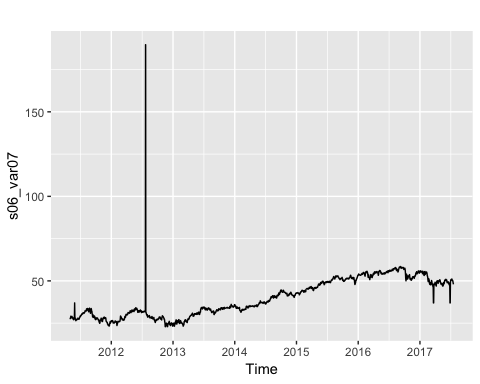


s06\_var05 %>% decompose(type = "multiplicative") %>%  
 autoplot() + xlab("Year")

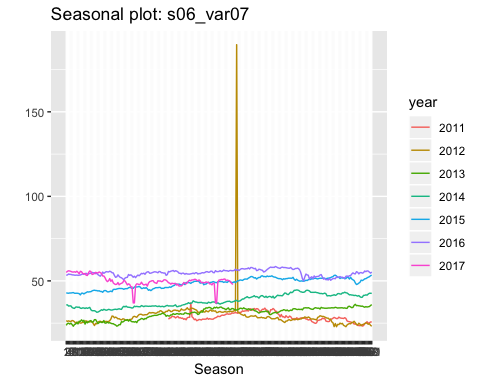


### S06 – Forecast Var07

s06\_var07 <- data1%>%  
 filter(group =='S06' ) %>%  
 select(SeriesInd, Var07)  
s06\_var07<- ts(s06\_var07[,2], start = c(2011,88), frequency = 261)  
autoplot(s06\_var07)



ggseasonplot(s06\_var07)



s06\_var07 %>% decompose(type = "multiplicative") %>%  
 autoplot() + xlab("Year")

