Second Harvest Practicum Project

Ben Foglia

6/5/2020

As part of a practicum project in Wake Forest University’s MS in Business Analytics program, I worked with Second Harvest Food Bank of Northwest North Carolina. I was given a csv of donor data dating back to 2004, and was tasked with forecasting future donations, and segmenting donors to improve future donation solicitation processes.

### Packages

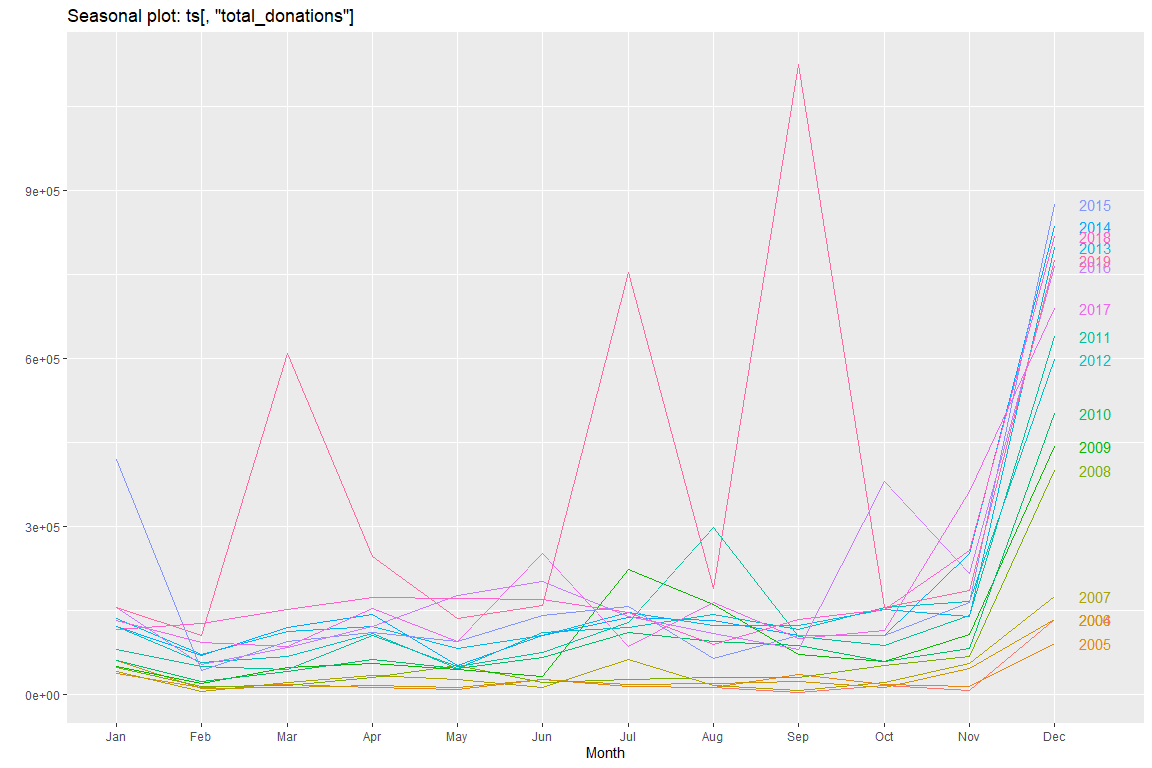
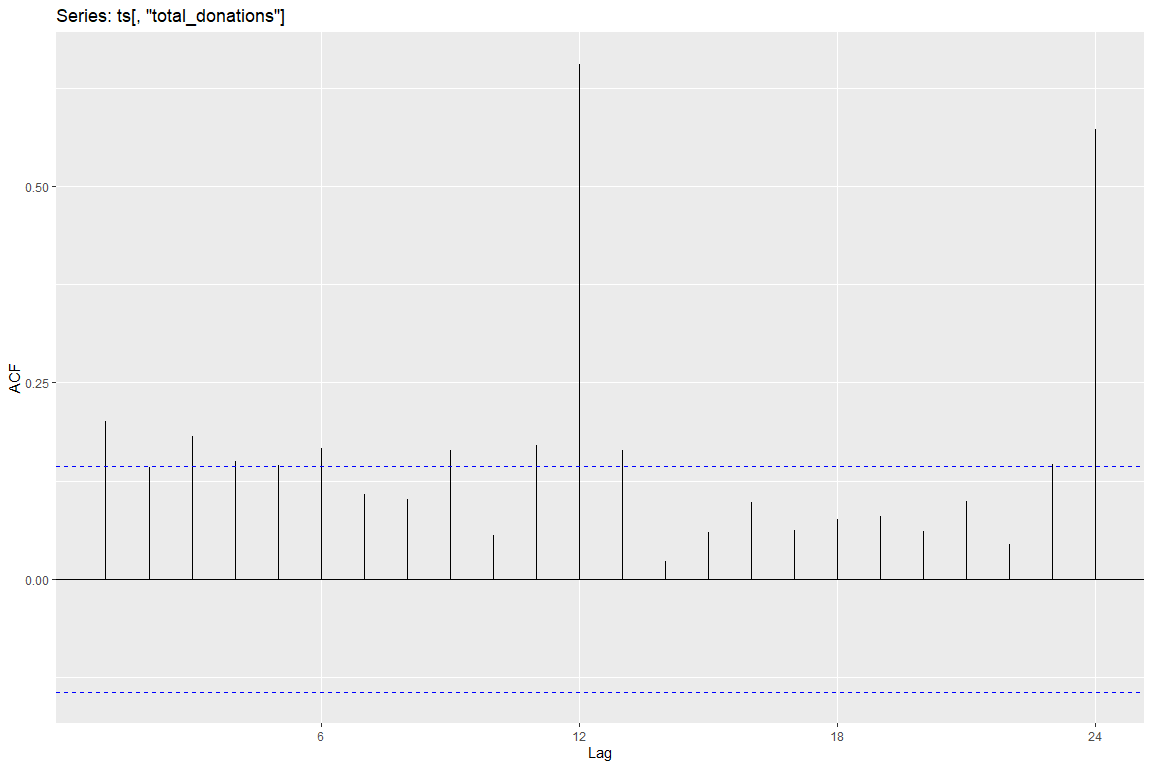
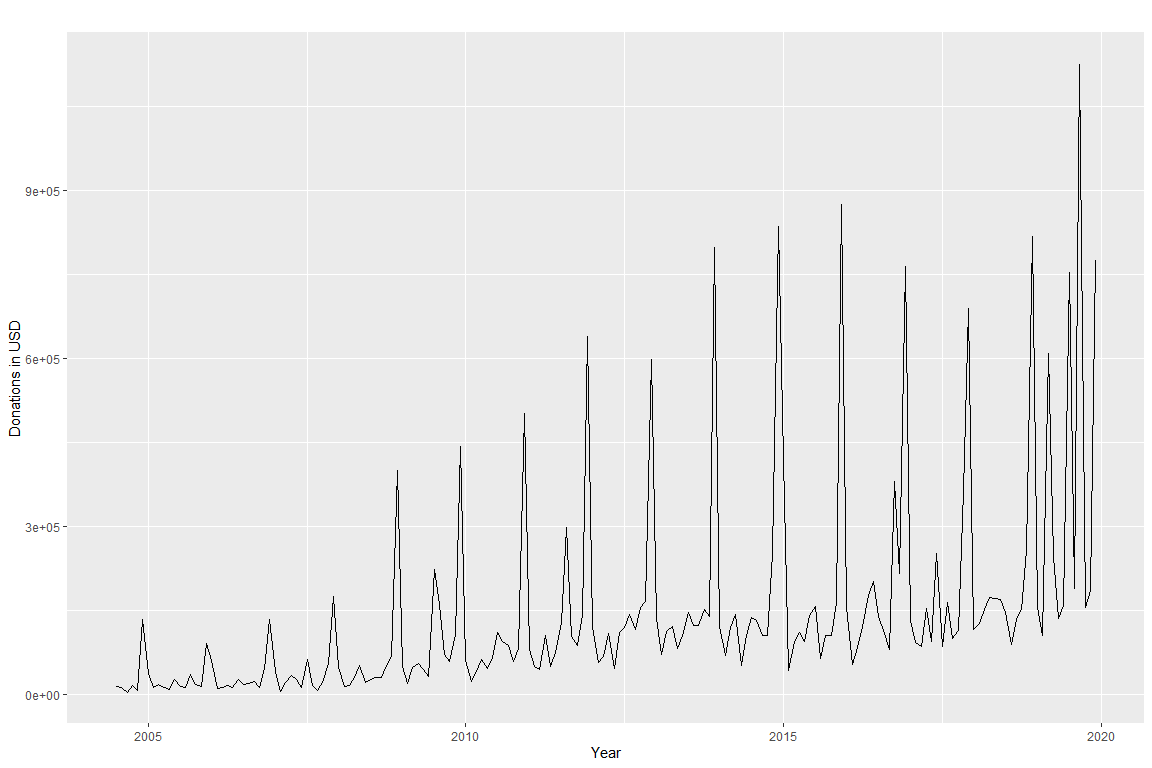
library(readr)  
library(ggplot2)  
library(forecast)  
library(dplyr)  
library(lubridate)  
library(rfm)  
library(RColorBrewer)  
library(knitr)

### Head of csv file

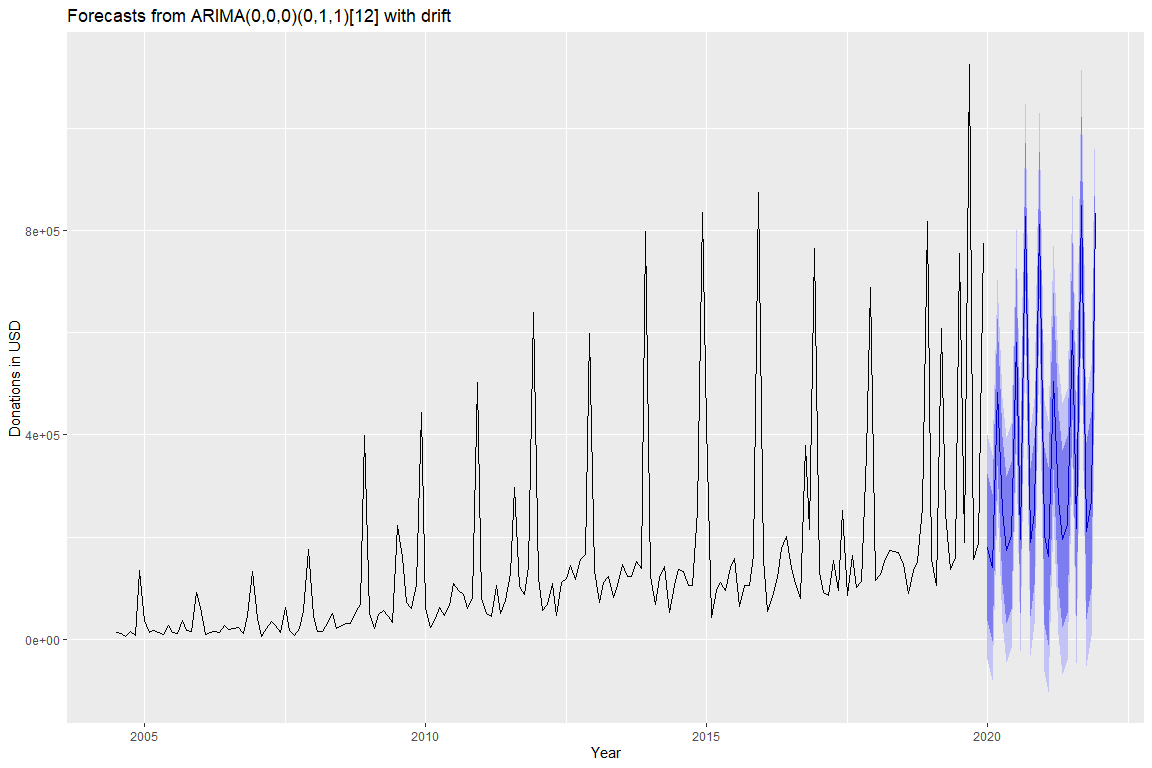
## # A tibble: 6 x 11  
## X1 `Unique Donor I~ `Gift Date` `Gift Fiscal Ye~ `Gift Amount`  
## <dbl> <dbl> <date> <chr> <dbl>  
## 1 1 1 2008-04-01 General 50   
## 2 2 1 2012-12-21 Annual Campaign~ 250   
## 3 3 1 2012-12-21 Annual Campaign~ 20   
## 4 4 1 2013-04-08 Annual Campaign~ 50   
## 5 5 1 2016-05-03 Annual Campaign~ 60   
## 6 6 2 2019-04-30 Annual Campaign~ 75.6  
## # ... with 6 more variables: `Donor's ZIP Code` <chr>, `Source of  
## # Donation` <chr>, `Method of Donation` <chr>, `Purpose of Donation` <chr>,  
## # `Tender Type` <chr>, Source <chr>

## Forecasting

I forecasted the total amount donated each month. I filtered down to only include non-capital individuals because corporations and capital individuals tend to donate randomly and only once. Filtering down removed these outliers. I then plotted a time series, autocorrelation function, and seasonal plot.



The time series shows a clear upward trend, and the spikes at lags 12 and 24 in the autocorrelation plot shows that there is indeed seasonality. Looking at the ACF and running auto.arima shows that this is an ARIMA(0,0,0)(0,1,1)[12] model, meaning there are significant spikes every 12 months, and those spikes decrease further along (lag 12 is the highest, lags 24, 36, etc. countinually decrease). We can now plot the forecast.



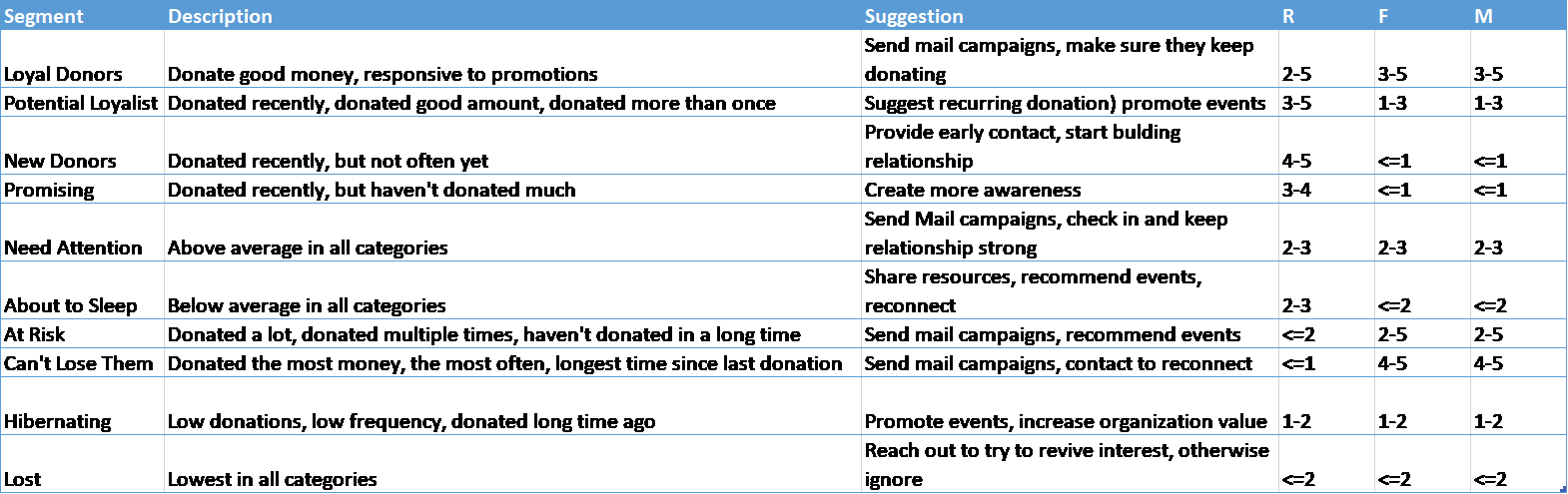
## Point Forecast Lo 80 Hi 80 Lo 95 Hi 95  
## Jan 2020 180604.6 36754.639 324454.7 -39394.96 400604.3  
## Feb 2020 140363.3 -3486.672 284213.3 -79636.27 360362.9  
## Mar 2020 483595.2 339745.145 627445.2 263595.55 703594.8  
## Apr 2020 252447.7 108597.733 396297.8 32448.14 472447.4  
## May 2020 174390.0 30540.026 318240.0 -45609.57 394389.6  
## Jun 2020 202585.2 58735.144 346435.2 -17414.45 422584.8

The plot and corresponding table show us the predicted value (point forecast in the table), as well as the values for the 80% and 95% confidence intervals (lighter and darker shaded areas in the plot, respectively).

Initially, I was only going to use the last 10 years of data to forecast, in order to leave out 2008 and 2009, where donations dipped because of the recession, but after the COVID-19 crisis hit, I decided it would be better to have the recession in just in case.

## Donor Segmenting

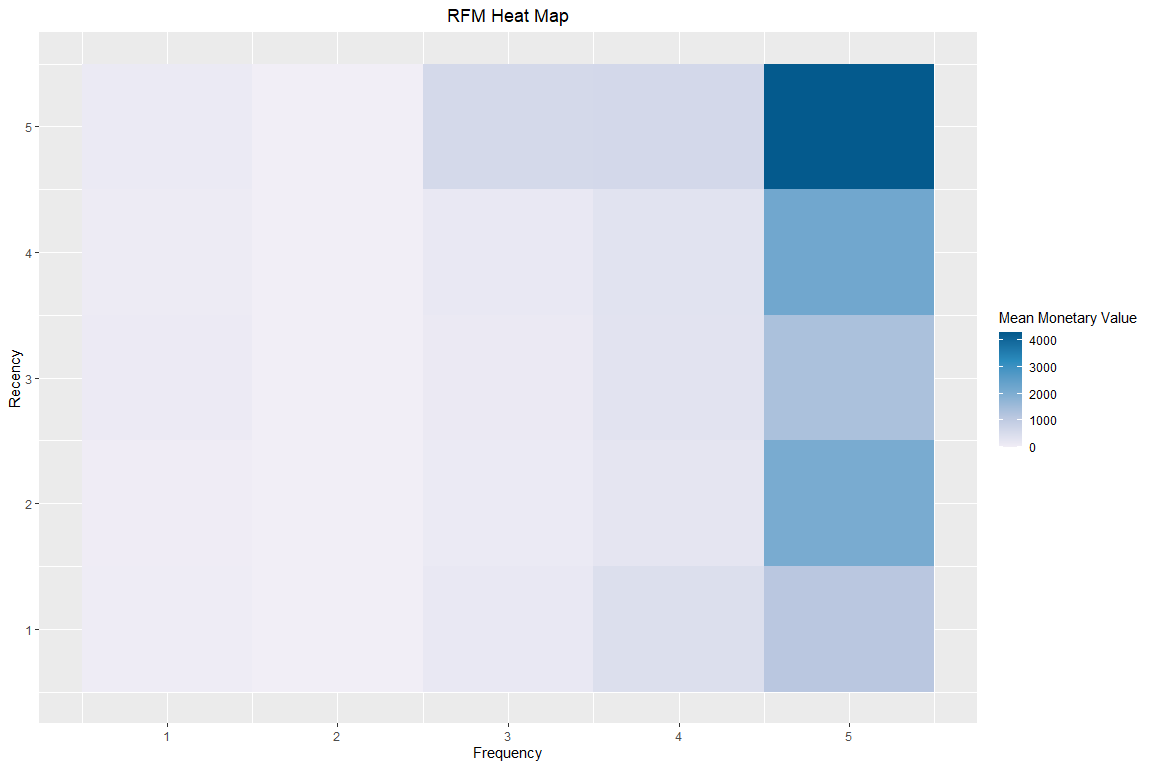
Second Harvest runs mailing campaigns throughout the year and sends a donation form to every one of their donors in the mail. This didn’t account for people who may have just donated or people that only donated once years ago and might not even be at that address anymore. To improve this, I ran an RFM analysis. This puts customers into segments based on recency, frequency, and monetary values. Customers are places into five equally sized bins for each of the three categories. An outline of the segments is shown below.



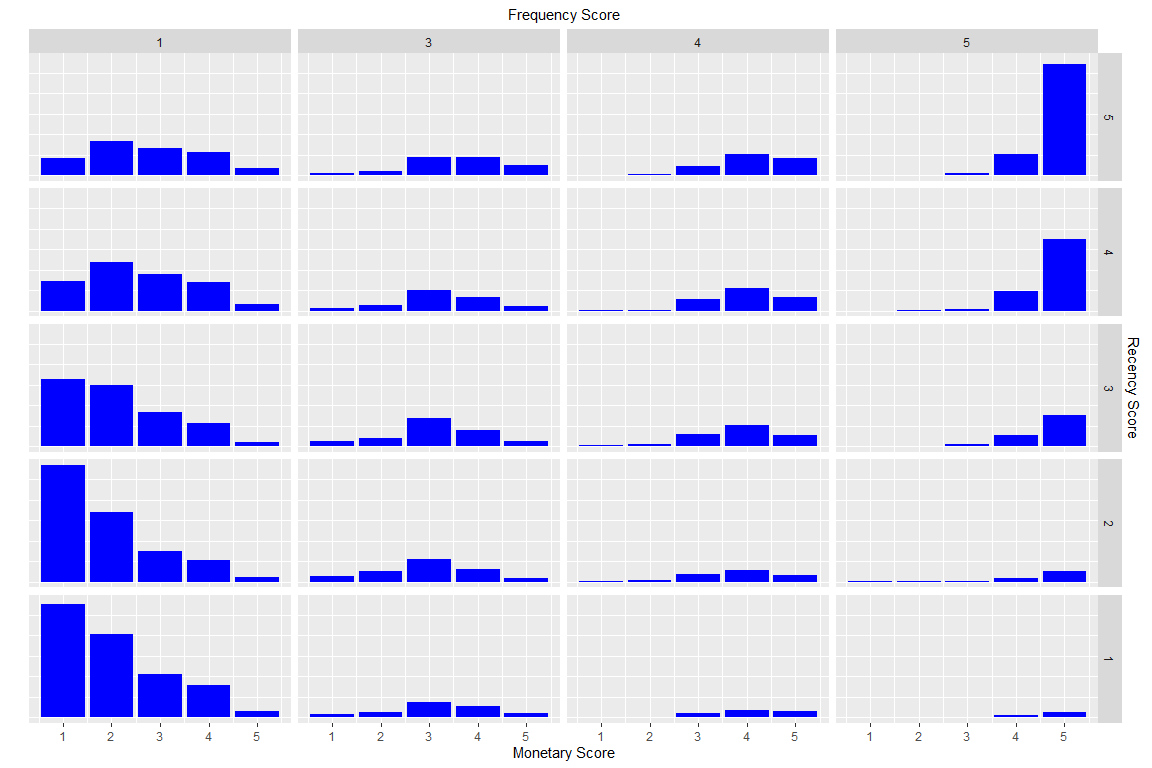
Again using only non-capital individuals, and using 12/31/2019 (the last day donation data is available) as our analysis date, we can begin to segment our donors and create plots.

## Classes 'rfm\_table\_order', 'tibble' and 'data.frame': 0 obs. of 6 variables:  
## $ rfm : tibble [41,922 x 9] (S3: tbl\_df/tbl/data.frame)  
## ..$ customer\_id : num 1 2 4 5 6 7 9 11 15 21 ...  
## ..$ date\_most\_recent : Date, format: "2016-05-03" "2019-04-30" ...  
## ..$ recency\_days : num 1337 245 3103 2891 286 ...  
## ..$ transaction\_count: num 5 1 1 2 9 1 27 2 19 1 ...  
## ..$ amount : num 430 75.6 100 20 185 ...  
## ..$ recency\_score : int 3 5 2 2 5 3 5 3 4 4 ...  
## ..$ frequency\_score : int 5 1 1 3 5 1 5 3 5 1 ...  
## ..$ monetary\_score : int 5 3 4 1 4 4 5 1 5 2 ...  
## ..$ rfm\_score : num 355 513 214 231 554 314 555 331 455 412 ...  
## $ analysis\_date : Date, format: "2019-12-31"  
## $ frequency\_bins: num 5  
## $ recency\_bins : num 5  
## $ monetary\_bins : num 5  
## $ threshold : tibble [5 x 6] (S3: tbl\_df/tbl/data.frame)  
## ..$ recency\_lower : num 0 385 1294 2277 3141  
## ..$ recency\_upper : num 385 1294 2277 3141 5661  
## ..$ frequency\_lower: num 1 2 2 3 5  
## ..$ frequency\_upper: num 2 2 3 5 333  
## ..$ monetary\_lower : num 0.05 21 36 86 251  
## ..$ monetary\_upper : num 21 36 86 251 1170788

*structure of newly created rfm tibble*



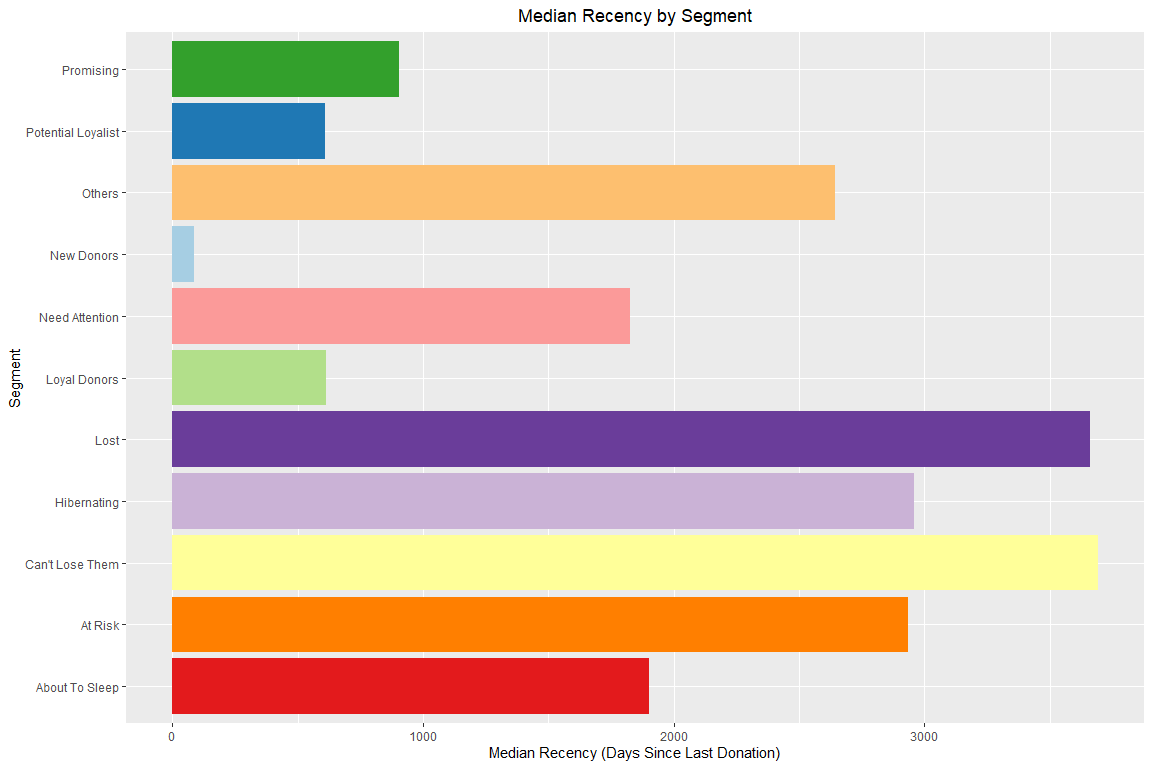
The heatmap shows us how much money we would expect someone to donate base on their recency and frequency scores. Our chart shows us that the most money is coming in from our most recent donors. This should be seen as a positive. If there were big money donors with very low recency scores, we would be seriously concerned about losing a large amount of donations.

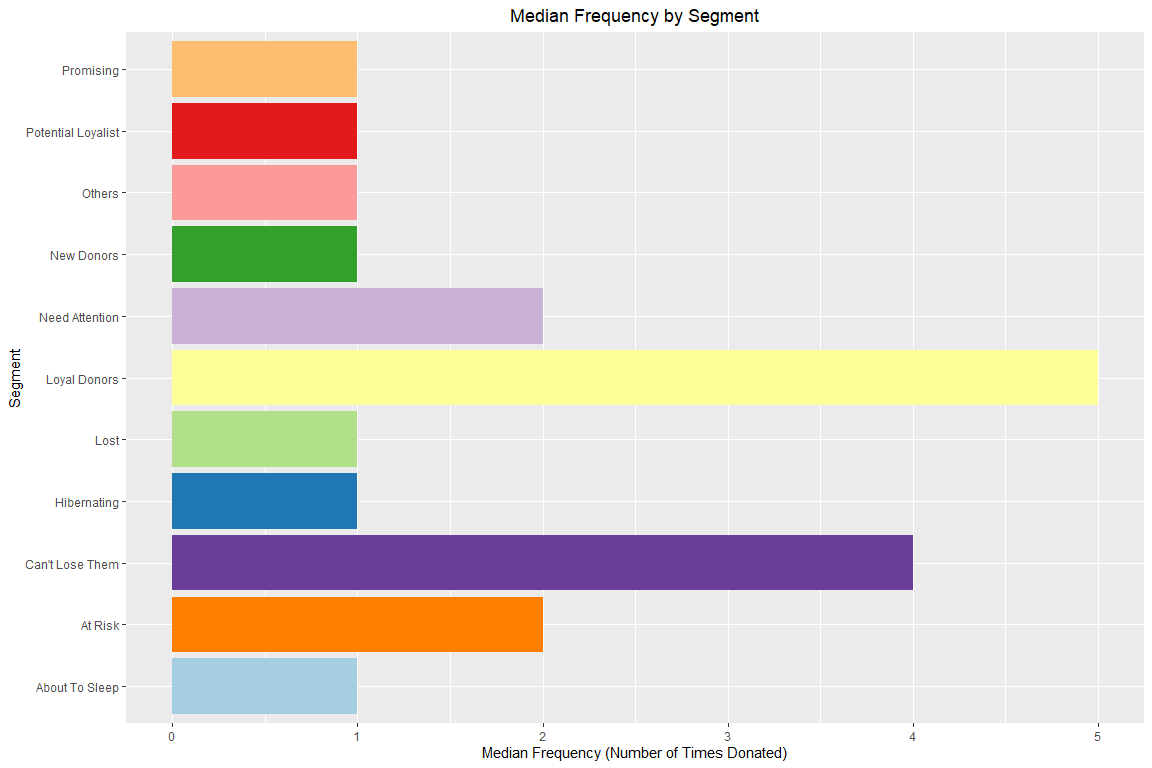


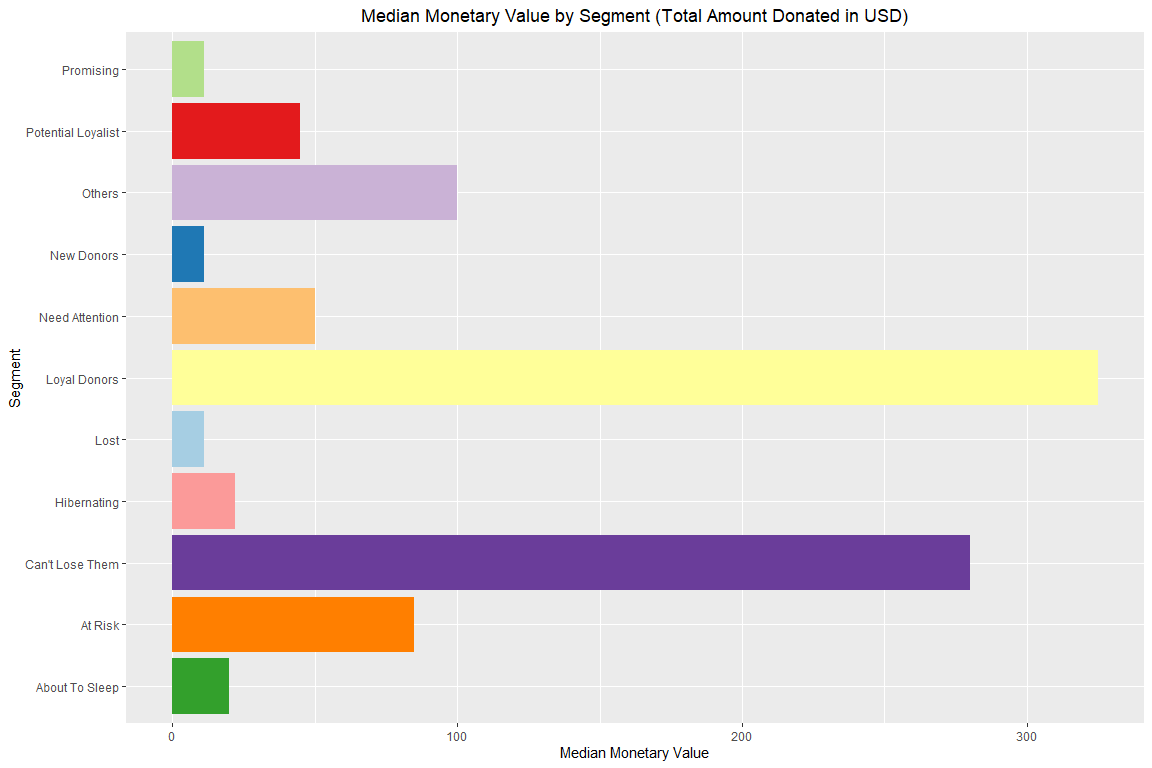
The bar charts further show us that the most recent and most frequent donors are donating the most money.

Now that we have the distriubution of the scores, let’s take a look at what the actual values in each segment are. I’ve created a chart of the amount of donors in each segment and 3 bar plots showing the median recency, frequency, and monetary values of each segment.

## # A tibble: 11 x 2  
## Segment Count  
## <chr> <int>  
## 1 Loyal Donors 11772  
## 2 Hibernating 6569  
## 3 Others 6031  
## 4 Potential Loyalist 5857  
## 5 At Risk 3269  
## 6 About To Sleep 3128  
## 7 Lost 2772  
## 8 Need Attention 870  
## 9 Promising 729  
## 10 Can't Lose Them 495  
## 11 New Donors 430







These graphs confirm what the segment chart originally showed us (i.e. “Loyal Donors” and “Can’t Lose Them” donate the most), but now we have exact values for each segment, rather than just the five bins.

## Conclusion

With this information, my hope is that Second Harvest can effictively market to their most crucial donors, especially in a time like this, where certain segments may not be able to donate during the pandemic. Hopefully it can use the RFM analysis and forecast together to make sure that they get the most they expected out of each segment.

## Code

library(readr)  
library(ggplot2)  
library(forecast)  
library(dplyr)  
library(lubridate)  
library(rfm)  
library(RColorBrewer)  
library(knitr)  
  
Donor\_Report <- read\_csv("F:/Practicum/donor\_report.csv",   
 col\_types = cols(`Gift Amount` = col\_number(),   
 `Gift Date` = col\_date(format = "%m/%d/%Y")))  
head(Donor\_Report)  
  
donors <- Donor\_Report %>% mutate(year = year(`Gift Date`), month = month(`Gift Date`)) %>%  
 group\_by(year, month) %>% filter(`Source of Donation` == "Individual") %>%   
 filter(`Source of Donation` != "Capital Individual") %>%  
 summarise(total\_donations = sum(`Gift Amount`))  
  
ts <- ts(donors, start = c(2004, 7), frequency = 12)   
  
autoplot(ts[,"total\_donations"], xlab = "Year", ylab = "Donations in USD")  
  
ggAcf(ts[,"total\_donations"])  
  
ggseasonplot(ts[,"total\_donations"], year.labels = TRUE)  
  
forecast <- auto.arima(ts[, 'total\_donations'])  
autoplot(forecast(forecast)) + xlab("Year") + ylab("Donations in USD")  
  
forecastdata <- forecast(forecast)  
forecasttable <- (as.data.frame(forecastdata))  
head(forecasttable)  
  
img <- "F:/rfm\_segments.png"  
include\_graphics(img)  
  
donorrfm <- Donor\_Report %>%   
 filter(`Source of Donation` == "Individual") %>%   
 filter(`Source of Donation` != "Capital Individual")  
  
analysis\_date <- as\_date("2019-12-31", tz = "EST")  
  
rfm <- rfm\_table\_order(donorrfm, customer\_id = `Unique Donor ID`,   
 order\_date = `Gift Date`, revenue = `Gift Amount`,   
 analysis\_date = analysis\_date)  
str(rfm)  
  
rfm\_heatmap(rfm)  
rfm\_bar\_chart(rfm)  
  
segment\_names <- c("Loyal Donors", "Potential Loyalist",  
 "New Donors", "Promising", "Need Attention", "About To Sleep",  
 "At Risk", "Can't Lose Them", "Hibernating", "Lost")  
recency\_lower <- c(2, 3, 4, 3, 2, 2, 1, 1, 1, 1)  
recency\_upper <- c(5, 5, 5, 4, 3, 3, 2, 1, 2, 1)  
frequency\_lower <- c(3, 1, 1, 1, 2, 1, 2, 4, 1, 1)  
frequency\_upper <- c(5, 3, 1, 1, 3, 2, 5, 5, 2, 1)  
monetary\_lower <- c(3, 1, 1, 1, 2, 1, 2, 4, 1, 1)  
monetary\_upper <- c(5, 3, 1, 1, 3, 2, 5, 5, 2, 1)  
rfm\_segments <- rfm\_segment(rfm, segment\_names = segment\_names, recency\_lower = recency\_lower, recency\_upper = recency\_upper,  
 frequency\_lower = frequency\_lower, frequency\_upper = frequency\_upper, monetary\_lower = monetary\_lower, monetary\_upper = monetary\_upper)  
  
rfm\_segments %>%  
 count(segment) %>%  
 arrange(desc(n)) %>%  
 rename(Segment = segment, Count = n)  
  
data <-   
 rfm\_segments %>%  
 group\_by(segment) %>%  
 select(segment, recency\_days) %>%  
 summarize(median(recency\_days)) %>%  
 rename(segment = segment, med\_recency = `median(recency\_days)`) %>%  
 arrange(med\_recency)   
  
n\_fill <- nrow(data)  
  
ggplot(data, aes(segment, med\_recency)) +  
 geom\_bar(stat = "identity", fill = brewer.pal(n=n\_fill, name = "Paired")) +  
 xlab("Segment") + ylab("Median Recency (Days Since Last Donation)") +  
 ggtitle("Median Recency by Segment") +  
 coord\_flip() +  
 theme(  
 plot.title = element\_text(hjust = 0.5)  
 )  
  
data <-   
 rfm\_segments %>%  
 group\_by(segment) %>%  
 select(segment, transaction\_count) %>%  
 summarize(median(transaction\_count)) %>%  
 rename(segment = segment, med\_frequency = `median(transaction\_count)`) %>%  
 arrange(med\_frequency)   
  
n\_fill <- nrow(data)  
  
ggplot(data, aes(segment, med\_frequency)) +  
 geom\_bar(stat = "identity", fill = brewer.pal(n = n\_fill, name = "Paired")) +  
 xlab("Segment") + ylab("Median Frequency (Number of Times Donated)") +  
 ggtitle("Median Frequency by Segment") +  
 coord\_flip() +  
 theme(  
 plot.title = element\_text(hjust = 0.5)  
 )  
  
data <-   
 rfm\_segments %>%  
 group\_by(segment) %>%  
 select(segment, amount) %>%  
 summarize(median(amount)) %>%  
 rename(segment = segment, med\_monetary = `median(amount)`) %>%  
 arrange(med\_monetary)   
  
n\_fill <- nrow(data)  
  
ggplot(data, aes(segment, med\_monetary)) +  
 geom\_bar(stat = "identity", fill = brewer.pal(n = n\_fill, name = "Paired")) +  
 xlab("Segment") + ylab("Median Monetary Value") +  
 ggtitle("Median Monetary Value by Segment (Total Amount Donated in USD)") +  
 coord\_flip() +  
 theme(  
 plot.title = element\_text(hjust = 0.5)  
 )