```
library(h2o)
require(reshape2)
require(tidyquant)
require(RcppRoll)
require(TTR)
require(RCurl)
library(lubridate)
library(imputeTS)
library(ggplot2)
library(caret)
require(h2o)

memory.limit(size=90000)
```

#Data directory

```
data.path <- "C:/Users/daveh/Desktop/Data/Stocks"
```

#Data path

```
files <- dir(data.path, pattern = "*.txt") # get file names
```

 # create a data frame holding the file names, read files into a new data column

```
allstocks <- data_frame(files) %>%
  mutate(file_contents = map(files,  ~ read.table(file.path(data.path, .),
sep=",", header=TRUE)))

allstocks <- unnest(allstocks)

allstocks$files <- toupper(str_split_fixed(allstocks$files, fixed("."), n =
2)[,1])

names(allstocks)[1]<- "S&P500.members"

allstocks$Date<- as.Date(allstocks$Date)

write_csv(allstocks,"allstocks.csv")
```

#Read-in Data

```
stocks <- read_csv("allstocks.csv")
dim(stocks)
```

#Features creation, dates extraction, and other data manipulations

#Features creation and dates selection

```
stocks.df <-stocks %>%
           group_by(Date)%>%
           mutate("Months" = months(as.Date(Date,"%A")),
                  "Monthly.Returns" = Close/Open-1,
                  "Monthly.Stock.Changes" = (Open-Close)/Open,
                  "Monthly.Volume.Rolling.Average" =
RcppRoll::roll_mean(Volume),
                  "Monthly.log.change.(%)" = Close-Open/Open*100,
                  "Monthly.Open.Change" = Open-lag(Open),
```

```r
                  "Monthly.Close.Change" = Close - lag(Close),
                  "Monthly.High.Change" = High - lag(High),
                  "Monthly.Low.Change" = Low - lag(Low))

stocks.df

#Grouped by year/quarters
quarters <- stocks.df%>%
  group_by("Quarters" = quarters.Date(Date),
           "QTR.Year" =format(as.Date(Date, format="%d/%m/%Y"),"%Y"))%>%
  subset(Date >= "2009/01/01" & Date <=  "2010/12/31")%>%
  select(-Close, -Open, -High, -Low, -OpenInt, -Volume, -Date)

quarters <- na.replace(quarters, 0)
quarters

dim(quarters)

write_csv(quarters,"quarters.csv")

#GLM Model relative Importance
#Read-in data
qrt <- read_csv("quarters.csv")

# h2o GLM example
h2o.init(nthreads=8, max_mem_size='4G')

h2o.removeAll()

# Import the prostate dataset
all.stocks = h2o.importFile(path = "C:/Users/daveh/Documents/APAN 5420/EDA
VIII/quarters.csv")

# Split dataset giving the training dataset 75% of the data
all.stocks.split = h2o.splitFrame(data=all.stocks, ratios=0.75)


# Create a training set from the 1st dataset in the split
all.stocks.train = all.stocks.split[[1]]

# Create a testing set from the 2nd dataset in the split
all.stocks.test = all.stocks.split[[2]]

# Generate a GLM model using the training dataset. x represents
the predictor column, and y represents the target index
all.stocks.glm = h2o.glm(x = c("Monthly.log.change.(%)",
                               "Monthly.Volume.Rolling.Average",
                               "Monthly.Stock.Changes",
                               "Monthly.Open.Change",
                               "Monthly.Close.Change",
                               "Monthly.Returns",
                               "Monthly.Low.Change"),
                         y =   "Monthly.High.Change",
                         training_frame= all.stocks.train,
```

```
                              nfolds =10, family = "gaussian",
                              alpha=0.1)
summary(all.stocks.glm)

sum <- summary(all.stocks.glm)
```

	names <chr>	coefficients <chr>	sign <chr>
1	Monthly.Open.Change	110.974971	POS
2	Monthly.Close.Change	110.966096	POS
3	Monthly.Low.Change	110.893977	POS
4	Monthly.log.change.(%)	77.916239	POS
5	Monthly.Returns	0.000000	POS
6	Monthly.Stock.Changes	0.000000	POS
7	Monthly.Volume.Rolling.Average	0.000000	POS

	mean <chr>	sd <chr>
mae	39847.26	3563.0786
mean_residual_deviance	2.96092251E12	3.71111625E11
mse	2.96092251E12	3.71111625E11
null_deviance	3.84923903E17	4.7823623E16
r2	3.9191043E-4	7.710672E-6
residual_deviance	3.84768047E17	4.7804176E16
rmse	1714267.9	105375.83
rmsle	0.0	NaN

The MAE, RMSE, and MSE errors have a high disparity (outliers exist) in terms of value, but their behavior caters to the model benchmark or what is expected from the model. I cannot fully elaborate on their respective behaviors as being good, bad, negative, or positive at the moment.
R^2, null deviance, and residual deviance is not too bad concerning the model fit, but not very strong since it is a little bit too large. The closer to 1 the better…. #deviancegoals

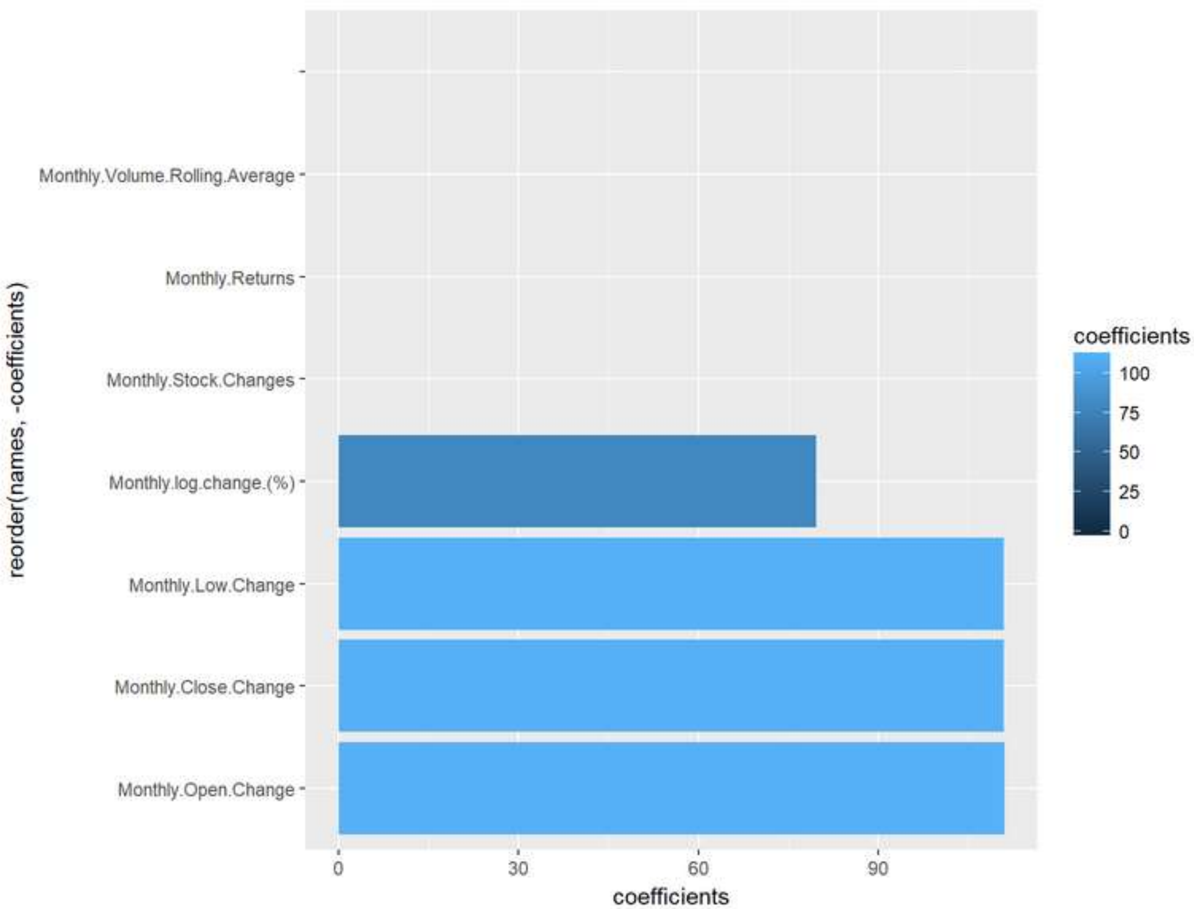
```
S&P500.members Months              Monthly.Returns       Monthly.Stock.Changes
LMT  :403      December :117665   Min.   :-9.940e-01   Min.   :-2.346e+03
DB   :401      March    :113485   1st Qu.:-1.217e-02   1st Qu.:-1.275e-02
ENLK:397       October  :113325   Median : 0.000e+00   Median : 0.000e+00
FLWS:397       June     :112317   Mean   : 3.057e-03   Mean   :-3.057e-03
RGR  :397      August   :111591   3rd Qu.: 1.275e-02   3rd Qu.: 1.217e-02
SJW  :397      September:110323   Max.   : 2.346e+03   Max.   : 9.940e-01
```

Strongest correlations came from the predictor variable (Montly.Returns) and the response variable (Monthly.Stock.Changes). They hit their max returns or experience change in September and their min returns and changes in the month of December, but not for the same stock members concerning both situations.

```
# Predict using the GLM model and the testing dataset
pred <- h2o.predict(all.stocks.glm, newdata=all.stocks.test)
pred


# View a summary of the prediction with a probability of TRUE
summary(all.stocks.train, exact_quantiles=TRUE)

#Plot the GLM model variable importance
ggplot(sum, aes(x = reorder(names, -coefficients),
                        y =coefficients, fill = coefficients)) +
                        geom_bar(stat = "identity")+
                        coord_flip()
```



The Random Forest model relative importance in regards to monthly closing changes was from four major contributors (open, high, log, and low changes), which is the driver for the changes that the monthly closing variable experiences. The other variables represent the outliers for this feature.