CptS575Project_Kumar_Patten

Ashutosh Kumar

12/5/2019

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date
```

```r
library(tidyr)
library(stringi)
library(data.table)
```

```
##
## Attaching package: 'data.table'

## The following objects are masked from 'package:lubridate':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday,
##     week, yday, year

## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```r
library(htree)
```

```
## Loading required package: parallel

## htree 2.0.0
```

```r
library(ggplot2)
library(ggthemes)
library(caret)
```

```
## Loading required package: lattice
```

```r
library(klaR)
```

```
## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```r
library(plyr)
```

```
## ------------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)

## ------------------------------------------------------------------------------

##
## Attaching package: 'plyr'

## The following object is masked from 'package:lubridate':
##
##     here

## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```r
library(ISLR)
library(lmtest)
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(MASS)
library(ggplot2)
library(DataExplorer)

setwd("~/Downloads/Fall 2019/Data Science 575/Project/Data")
all = read.csv("AllCompaniesNEW.csv")

all$mcap = abs(all$ALTPRC)*all$SHROUT
newdata = subset(all, all$mcap >= 3000000) # Choosing the dataset with companies having more than $3 bi

sandp = read.csv("New S&P 500 Additions.csv")

# Company.cusip has an extra digit compared to the original CUSIP. I remvoed this digit.


sandp$cusipn = as.character(sandp$Company.Cusip)
cusipLen = nchar(sandp$cusipn)
cusipTrunc = substr(sandp$cusipn, start=1, stop=(cusipLen-1))
sandp$cusip = cusipTrunc

#write.csv(sandp,"sandpnew.csv")
# Getting rid of observations without deletion date
sandprefined = subset(sandp, Deletion_Date != "NA")
#write.csv(sandprefined,"sandprefinednew.csv")
#sandprefined = sandprefined[c(4,5,22)]
#summary(sandprefined$Deletion_Date)

# Merging s&p addition and deletion into the list of all companies
m1=merge(sandprefined, newdata, by.x = "cusip", by.y = "CUSIP", all.x = FALSE)

# Creating a variable to denote if the particular company is listed in the recorded month or not.
m1$splisted = ifelse(m1$ALTPRCDT>m1$Addition_Date & m1$ALTPRCDT<m1$Deletion_Date, 1,0)
m3 = subset(m1, mcap<72000000)
#ary(m3)# We got rid of around 300 observations by this.

x=count(m3, 'cusip')
sum(x$freq)
```

```
## [1] 31618
```

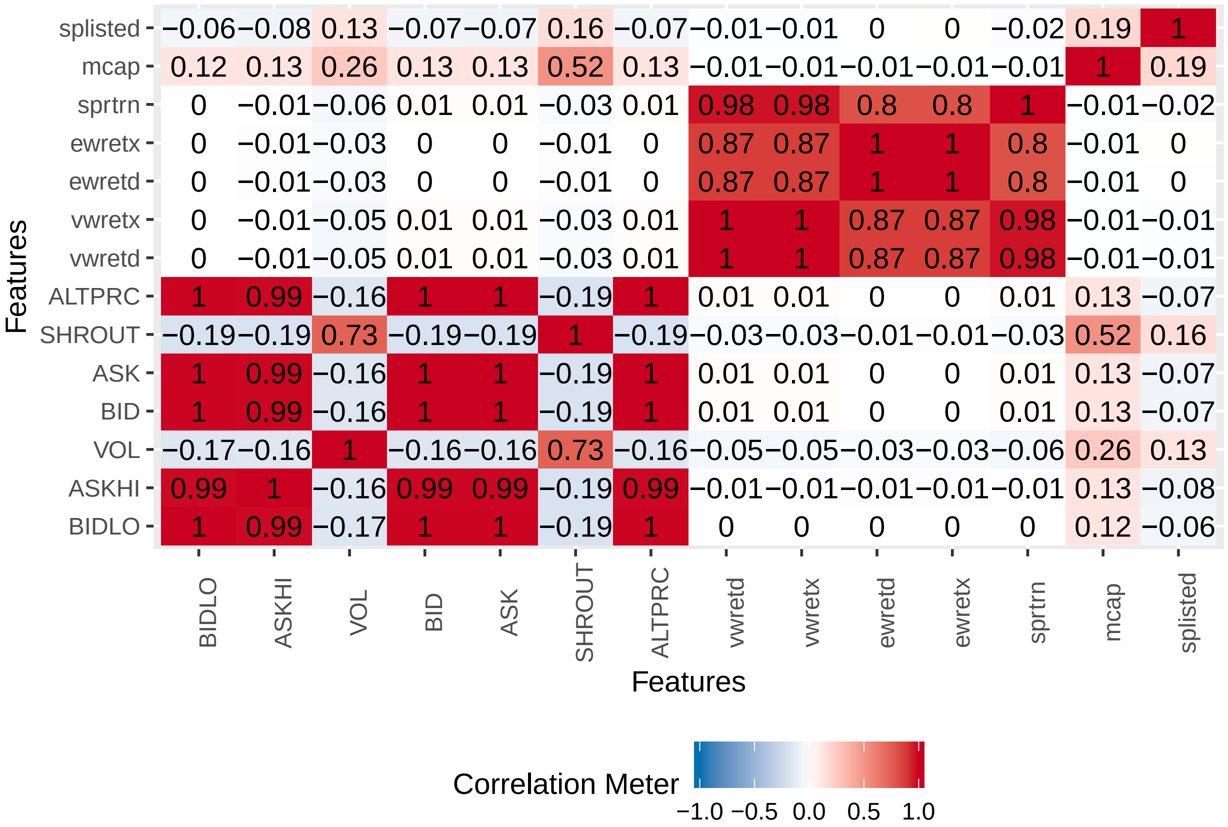
```
#Creating unique id for each company based on CUSIP
m3$cusip = as.factor(m3$cusip)
m3$cusip = as.numeric(m3$cusip)


# Dealing with missing data
# exclude variables v1, v2, v3
#m2 = m3[-c(2,3,4,7,8,9,10,11,12,13,14,15,16,17,18,19, 20, 21, 22,23,25,27,28,29,30,31,32, 33,34,35,36,
m2 = m3[c(1,24,37,38,40,42,43,44,45,49,50,51,52,53,54,55)]
m2 = na.exclude(m2)

m4 = m2[-c(1,2)] # Final working dataset
summary(m4)
```

```
##      BIDLO             ASKHI              VOL                BID        
##  Min.   : -80.88   Min.   :   1.23   Min.   :      810   Min.   :   1.00  
##  1st Qu.:  23.45   1st Qu.:  26.90   1st Qu.:   180479   1st Qu.:  25.10  
##  Median :  35.17   Median :  39.84   Median :   375156   Median :  37.51  
##  Mean   :  44.04   Mean   :  50.18   Mean   :   728988   Mean   :  47.13  
##  3rd Qu.:  52.45   3rd Qu.:  59.75   3rd Qu.:   754642   3rd Qu.:  56.19  
##  Max.   :1051.90   Max.   :1092.34   Max.   :40940055   Max.   :1075.05  
##       ASK              SHROUT            ALTPRC            vwretd         
##  Min.   :   1.24   Min.   :   4624   Min.   : -80.88   Min.   :-0.184648  
##  1st Qu.:  25.21   1st Qu.: 123133   1st Qu.:  25.16   1st Qu.:-0.017553  
##  Median :  37.65   Median : 183716   Median :  37.57   Median : 0.012950  
##  Mean   :  47.27   Mean   : 303631   Mean   :  47.20   Mean   : 0.008519  
##  3rd Qu.:  56.38   3rd Qu.: 326145   3rd Qu.:  56.27   3rd Qu.: 0.037932  
##  Max.   :1077.53   Max.   :4484000   Max.   :1075.05   Max.   : 0.114030  
##      vwretx              ewretd             ewretx         
##  Min.   :-0.186136   Min.   :-0.20522   Min.   :-0.206835  
##  1st Qu.:-0.018432   1st Qu.:-0.01907   1st Qu.:-0.020191  
##  Median : 0.011475   Median : 0.01348   Median : 0.011157  
##  Mean   : 0.006909   Mean   : 0.01046   Mean   : 0.008883  
##  3rd Qu.: 0.036089   3rd Qu.: 0.03970   3rd Qu.: 0.037053  
##  Max.   : 0.112619   Max.   : 0.22504   Max.   : 0.224085  
##      sprtrn               mcap             splisted     
##  Min.   :-0.169425   Min.   : 3000099   Min.   :0.0000  
##  1st Qu.:-0.017396   1st Qu.: 4349871   1st Qu.:0.0000  
##  Median : 0.010674   Median : 6487394   Median :1.0000  
##  Mean   : 0.006767   Mean   : 9941891   Mean   :0.6515  
##  3rd Qu.: 0.032549   3rd Qu.:11326008   3rd Qu.:1.0000  
##  Max.   : 0.111588   Max.   :71980807   Max.   :1.0000  
```

```
head(m4)
```

```
##   BIDLO ASKHI     VOL   BID   ASK SHROUT ALTPRC    vwretd    vwretx
## 1 40.51 41.40  603768 41.24 41.25 165546  41.26  0.070518  0.068146
## 2 24.94 40.40 1730469 40.35 40.36 165319  40.37  0.000685 -0.001737
## 3 32.55 35.47  376574 35.46 35.47 164937  35.47  0.002448  0.000182
## 4 41.27 41.98  831265 41.97 41.98 165546  41.98  0.011814  0.010579
## 5 33.51 36.18  474552 34.38 34.39 171152  34.40 -0.027200 -0.028272
## 6 29.08 32.18  685295 30.23 30.24 174246  30.24  0.001670  0.000506
##      ewretd    ewretx    sprtrn    mcap splisted
## 1  0.078191  0.075547  0.065991 6830428        1
## 2  0.005557  0.003629 -0.004128 6673928        1
## 3  0.007333  0.005435  0.000505 5850315        1
## 4  0.039819  0.038583  0.002699 6949621        1
## 5 -0.019073 -0.020095 -0.031041 5887629        1
## 6 -0.022319 -0.023436  0.006201 5269199        1
```

```
plot_correlation(m4)
```



```
m4$splisted = as.factor(m4$splisted)
datanew = m4

set.seed(123)

trainIndex=createDataPartition(datanew$splisted, p=0.8)$Resample1
train=datanew[trainIndex, ]
test=datanew[-trainIndex, ]

head(train)
```

```
##   BIDLO ASKHI     VOL   BID   ASK SHROUT ALTPRC    vwretd    vwretx
## 1 40.51 41.40  603768 41.24 41.25 165546  41.26  0.070518  0.068146
## 2 24.94 40.40 1730469 40.35 40.36 165319  40.37  0.000685 -0.001737
## 3 32.55 35.47  376574 35.46 35.47 164937  35.47  0.002448  0.000182
## 4 41.27 41.98  831265 41.97 41.98 165546  41.98  0.011814  0.010579
## 5 33.51 36.18  474552 34.38 34.39 171152  34.40 -0.027200 -0.028272
## 6 29.08 32.18  685295 30.23 30.24 174246  30.24  0.001670  0.000506
##      ewretd    ewretx    sprtrn    mcap splisted
## 1  0.078191  0.075547  0.065991 6830428        1
## 2  0.005557  0.003629 -0.004128 6673928        1
## 3  0.007333  0.005435  0.000505 5850315        1
## 4  0.039819  0.038583  0.002699 6949621        1
## 5 -0.019073 -0.020095 -0.031041 5887629        1
## 6 -0.022319 -0.023436  0.006201 5269199        1
```

```
head(test)
```

```
##    BIDLO ASKHI    VOL   BID   ASK SHROUT ALTPRC    vwretd    vwretx
## 12 30.07 32.89 482879 32.19 32.20 174246  32.20  0.020223  0.018015
## 15 30.01 34.11 303671 33.03 33.04 169933  33.04  0.074021  0.072626
## 20 37.88 41.64 512065 41.52 41.53 171152  41.52 -0.010454 -0.012270
## 33 34.00 37.22 398566 36.86 36.87 174008  36.86  0.040206  0.038203
## 34 39.75 41.51 490777 39.82 39.83 210520  39.83 -0.025727 -0.027904
## 35 35.48 39.84 353144 39.23 39.24 171152  39.22  0.056017  0.053679
##       ewretd    ewretx    sprtrn    mcap splisted
## 12  0.006647  0.004858  0.021030 5610721        1
## 15  0.052873  0.051512  0.082983 5614586        1
## 20 -0.005112 -0.007204 -0.017396 7106231        1
## 33  0.033697  0.032156  0.037655 6413935        1
## 34 -0.017278 -0.019029 -0.031298 8385012        1
## 35  0.054123  0.052497  0.054893 6712581        1
```

```
## check the balance
print(table(datanew$splisted))
```

```
##
##     0     1
## 10683 19971
```

```
print(table(train$splisted))
```

```
##
##     0     1
##  8547 15977
```

```
####
library(randomForest)
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
model_1 = randomForest(splisted~ASKHI+ VOL+SHROUT+ALTPRC+vwretd+ewretd+I(ASKHI^2)+I(VOL^2), data = trai

print(model_1)
```

```
##
## Call:
##  randomForest(formula = splisted ~ ASKHI + VOL + SHROUT + ALTPRC +      vwretd + ewretd + I(ASKHI^2)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 2
##
##         OOB estimate of  error rate: 22.38%
## Confusion matrix:
##      0     1 class.error
## 0 4963  3584   0.4193284
## 1 1905 14072   0.1192339
```

```
pred = predict(model_1, data = test)


testPred=predict(model_1, newdata=test, type="class")
tab_test = table(testPred, test$splisted)
caret::confusionMatrix(tab_test)
```

```
## Confusion Matrix and Statistics
##
##
## testPred    0    1
##        0 1260  426
##        1  876 3568
##
##                Accuracy : 0.7876
##                  95% CI : (0.7771, 0.7978)
##     No Information Rate : 0.6515
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5081
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.5899
##             Specificity : 0.8933
##          Pos Pred Value : 0.7473
##          Neg Pred Value : 0.8029
##              Prevalence : 0.3485
##          Detection Rate : 0.2055
##    Detection Prevalence : 0.2750
##       Balanced Accuracy : 0.7416
##
##        'Positive' Class : 0
##
```

```r
trainPred=predict(model_1, newdata = train, type = "class")
tab_train = table(trainPred, train$splisted)
caret::confusionMatrix(tab_train)
```

```
## Confusion Matrix and Statistics
##
##
## trainPred     0     1
##         0  8419   213
##         1   128 15764
##
##                Accuracy : 0.9861
##                  95% CI : (0.9846, 0.9875)
##     No Information Rate : 0.6515
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9695
##
##  Mcnemar's Test P-Value : 5.394e-06
##
##             Sensitivity : 0.9850
##             Specificity : 0.9867
##          Pos Pred Value : 0.9753
##          Neg Pred Value : 0.9919
##              Prevalence : 0.3485
##          Detection Rate : 0.3433
##    Detection Prevalence : 0.3520
##       Balanced Accuracy : 0.9858
##
##        'Positive' Class : 0
##
```