Market Basket of Hand Picked Stocks

Janis Corona

2/15/2020

I used yahoo finance at finance.yahoo.com to grab a list of stocks I wanted to examine over time.

```
HandStocks <- read.csv('yahooStockBasket.csv', header=TRUE, sep=',',
                       na.strings=c('',' '))
```

Create the NYSE subset and the Nasdaq subset. There are also a few that are ‘other OTC’ labeled that I will exclude.

```
NYSE <- subset(HandStocks, HandStocks$stockExchange=='NYSE')
NASDAQ <- subset(HandStocks, HandStocks$stockExchange=='Nasdaq')
```

This is a youtube tutorial on quant finance from ’Quant Finance with R Part 1 intro and Data": This tutorial link.

The github repository for these tutorials are at: https://github.com/fdupuis659/Quant-Finance-with-R

The changes made are that the NYSE and NASDAQ stocks read in above will be used.

```
nyse <- as.character(NYSE$stock)
nasdaq <- as.character(NASDAQ$stock)

library(quantmod)
library(PerformanceAnalytics)

tickers <- nyse

weights <- rep(1/length(tickers), length(tickers))

NYSE_portfolioPrices <- NULL

for (ticker in tickers){
  NYSE_portfolioPrices <- cbind(NYSE_portfolioPrices,
getSymbols.yahoo(ticker,
  from = '2007-01-03', periodicity='daily', auto.assign=FALSE)[,4])

}
```

Check NAs not in data.

```
colSums(is.na(NYSE_portfolioPrices))
```

```
##    TGT.Close     HD.Close    JPM.Close    XOM.Close    CVX.Close 
MGM.Close 
##            0            0            0            0            0 
0 
##   TEVA.Close    HST.Close   FCAU.Close    WFC.Close    WWE.Close 
QSR.Close 
##            0            0          864            0            0 
2000 
## SCE.PB.Close     WM.Close      S.Close     GM.Close      F.Close 
JWN.Close 
##         1195            0            0          978            0 
0 
##    NUS.Close    AMC.Close    KSS.Close    LUV.Close    HMC.Close 
PCG.Close 
##            0         1753            0            0            0 
0 
##    NKE.Close    WMT.Close    TJX.Close     TM.Close      T.Close 
JNJ.Close 
##            0            0            0            0            0 
0 
##      C.Close    EPD.Close     VZ.Close    HRB.Close    AAP.Close 
SIG.Close 
##            0            0            0            0            0 
0 
##      M.Close   YELP.Close 
##            0         1301
```

There are some stocks with missing values and this is probably due to so far back the dates are pulled from 2007. Lets make a separate data set for those and remove them from this one. FCAU, QSR, SCE.PB, GM, AMC, and Yelp have many NAs.

```
NYSE_portfolioPrices_2007 <- NYSE_portfolioPrices[,-c(9,12,13,16,20,38)]
colSums(is.na(NYSE_portfolioPrices_2007))
```

```
##  TGT.Close   HD.Close  JPM.Close  XOM.Close  CVX.Close  MGM.Close 
TEVA.Close 
##          0          0          0          0          0          0 
0 
##  HST.Close  WFC.Close  WWE.Close   WM.Close    S.Close    F.Close 
JWN.Close 
##          0          0          0          0          0          0 
0 
##  NUS.Close  KSS.Close  LUV.Close  HMC.Close  PCG.Close  NKE.Close 
WMT.Close 
##          0          0          0          0          0          0 
0 
##  TJX.Close   TM.Close    T.Close  JNJ.Close    C.Close  EPD.Close 
VZ.Close 
##          0          0          0          0          0          0 
0
```

```
##  HRB.Close  AAP.Close  SIG.Close    M.Close
##          0          0          0          0

NYSE_portfolioPrices_2015 <-
NYSE_portfolioPrices[complete.cases(NYSE_portfolioPrices),]
```

So we have all data for NYSE portfolio prices since 2007 that excluded some stock not available in our list, and data on all the stocks in the list since December 2014.Lets do the same for the NASDAQ stocks.

```
tickers2 <- nasdaq

weights <- rep(1/length(tickers2), length(tickers2))

NASDAQ_portfolioPrices <- NULL

for (ticker in tickers2){
  NASDAQ_portfolioPrices <- cbind(NASDAQ_portfolioPrices,
getSymbols.yahoo(ticker,
  from = '2007-01-03', periodicity='daily', auto.assign=FALSE)[,4])

}
```

Check NAs not in data.

```
colSums(is.na(NASDAQ_portfolioPrices))

##   FTR.Close  UBSI.Close   INO.Close  GRPN.Close  FFIN.Close  GOOG.Close
##           0           0           0        1221           0           0
##  ONCY.Close  ARWR.Close  COST.Close   AAL.Close CSSEP.Close  MSFT.Close
##           0           0           0           0        2891           0
##  DLTR.Close  KGJI.Close  AMZN.Close  ROST.Close  TMUS.Close  PBYI.Close
##           0           0           0           0          73        1337
##  NFLX.Close  HOFT.Close   SDC.Close  RRGB.Close  JBLU.Close
##           0           0        3195           0           0
```

There are also some stocks with missing values for NASDAQ pulled from 2007. Lets make a separate data set for those and remove them from this one. GRPN, CSSEP, TMUS, PBYI, and SDC are the stock with many NA values.

```
NASDAQ_portfolioPrices_2007 <- NASDAQ_portfolioPrices[,-c(4,11,17,18,21)]
colSums(is.na(NASDAQ_portfolioPrices_2007))

##  FTR.Close UBSI.Close  INO.Close FFIN.Close GOOG.Close ONCY.Close
ARWR.Close
##          0          0          0          0          0          0
0
## COST.Close  AAL.Close MSFT.Close DLTR.Close KGJI.Close AMZN.Close
ROST.Close
##          0          0          0          0          0          0
0
```

```
## NFLX.Close HOFT.Close RRGB.Close JBLU.Close
##          0          0          0          0

NASDAQ_portfolioPrices_2019 <-
NASDAQ_portfolioPrices[complete.cases(NASDAQ_portfolioPrices),]
```

So we have all data for NYSE portfolio prices since 2007 that excluded some stock not available in our list, and data on all the stocks in the list since September 2019 as that was the earliest date that all stocks had available data.

S&P benchmark

```
benchmarkPrices <- getSymbols.yahoo('^GSPC', from='2007-01-03',
periodicity='daily', auto.assign=FALSE)[,4]
```

Calculate daily change in each column.

```
benchmarkReturns <- na.omit(ROC(benchmarkPrices))
colSums(is.na(benchmarkReturns))

## GSPC.Close
##          0

NYSE_2007_portfolioReturns <- na.omit(ROC(NYSE_portfolioPrices_2007))
colSums(is.na(NYSE_2007_portfolioReturns))

##  TGT.Close   HD.Close  JPM.Close  XOM.Close  CVX.Close  MGM.Close
TEVA.Close
##          0          0          0          0          0          0
0
##  HST.Close  WFC.Close  WWE.Close   WM.Close    S.Close    F.Close
JWN.Close
##          0          0          0          0          0          0
0
##  NUS.Close  KSS.Close  LUV.Close  HMC.Close  PCG.Close  NKE.Close
WMT.Close
##          0          0          0          0          0          0
0
##  TJX.Close   TM.Close    T.Close  JNJ.Close    C.Close  EPD.Close
VZ.Close
##          0          0          0          0          0          0
0
##  HRB.Close  AAP.Close  SIG.Close    M.Close
##          0          0          0          0

NYSE_2015_portfolioReturns <- na.omit(ROC(NYSE_portfolioPrices_2015))
colSums(is.na(NYSE_2015_portfolioReturns))

##    TGT.Close     HD.Close    JPM.Close    XOM.Close    CVX.Close
MGM.Close
##            0            0            0            0            0
0
```

```
##   TEVA.Close    HST.Close   FCAU.Close    WFC.Close    WWE.Close
QSR.Close
##            0            0            0            0            0
0
## SCE.PB.Close     WM.Close      S.Close     GM.Close      F.Close
JWN.Close
##            0            0            0            0            0
0
##    NUS.Close    AMC.Close    KSS.Close    LUV.Close    HMC.Close
PCG.Close
##            0            0            0            0            0
0
##    NKE.Close    WMT.Close    TJX.Close     TM.Close      T.Close
JNJ.Close
##            0            0            0            0            0
0
##      C.Close    EPD.Close     VZ.Close    HRB.Close    AAP.Close
SIG.Close
##            0            0            0            0            0
0
##      M.Close   YELP.Close
##            0            0
```

```
NASDAQ_2007_portfolioReturns <- na.omit(ROC(NASDAQ_portfolioPrices_2007))
colSums(is.na(NASDAQ_2007_portfolioReturns))
```

```
##  FTR.Close UBSI.Close  INO.Close FFIN.Close GOOG.Close ONCY.Close
ARWR.Close
##          0          0          0          0          0          0
0
## COST.Close  AAL.Close MSFT.Close DLTR.Close KGJI.Close AMZN.Close
ROST.Close
##          0          0          0          0          0          0
0
## NFLX.Close HOFT.Close RRGB.Close JBLU.Close
##          0          0          0          0
```

```
NASDAQ_2019_portfolioReturns <- na.omit(ROC(NASDAQ_portfolioPrices_2019))
colSums(is.na(NASDAQ_2019_portfolioReturns))
```

```
##   FTR.Close  UBSI.Close   INO.Close  GRPN.Close  FFIN.Close  GOOG.Close
##           0           0           0           0           0           0
##  ONCY.Close  ARWR.Close  COST.Close   AAL.Close CSSEP.Close  MSFT.Close
##           0           0           0           0           0           0
##  DLTR.Close  KGJI.Close  AMZN.Close  ROST.Close  TMUS.Close  PBYI.Close
##           0           0           0           0           0           0
##  NFLX.Close  HOFT.Close   SDC.Close  RRGB.Close  JBLU.Close
##           0           0           0           0           0
```

```
NYSE_2007_portfolioReturn <- Return.portfolio(NYSE_2007_portfolioReturns)
NYSE_2015_portfolioReturn <- Return.portfolio(NYSE_2015_portfolioReturns)
```

```
NASDAQ_2007_portfolioReturn <- Return.portfolio(NASDAQ_2007_portfolioReturns)
NASDAQ_2019_portfolioReturn <- Return.portfolio(NASDAQ_2019_portfolioReturns)
```

To find out more on the Return.portfolio function, use: * ? $Return.portfolio$

Some side information about a few financial algorithms:

- **CAPM**: formula for expected return with calculated risk on an asset or stock.
- **ALPHA**: risk adjustment metric for performances compares to an index and shows how much better that index is beat by your benchmark.
- **BETA**: measure of volatility with <1 => less risky and >1 => more risky.
- **SHARPE RATIO**: risk metric for every standard deviation unit, how much return is achieved, gives risk & reward, and most widely used metric with finance managers.

This section shows portfolio returns on the NYSE since 2007 stocks.

The number of trading days is 252 days a year.

```
CAPM.beta(NYSE_2007_portfolioReturn, benchmarkReturns, 0.035/252)

## [1] 0.8787609

CAPM.jensenAlpha(NYSE_2007_portfolioReturn, benchmarkReturns, 0.035/252)

## [1] -0.03233876

SharpeRatio(NYSE_2007_portfolioReturn, 0.035/252)

##                               portfolio.returns
## StdDev Sharpe (Rf=0%, p=95%):      -0.002775803
## VaR Sharpe (Rf=0%, p=95%):         -0.001834287
## ES Sharpe (Rf=0%, p=95%):          -0.001057602

table.AnnualizedReturns(NYSE_2007_portfolioReturn)

##                           portfolio.returns
## Annualized Return                    0.0105
## Annualized Std Dev                   0.1815
## Annualized Sharpe (Rf=0%)            0.0581

table.CalendarReturns(NYSE_2007_portfolioReturn)

##       Jan  Feb  Mar  Apr  May Jun  Jul  Aug  Sep  Oct  Nov  Dec
## 2007  0.6 -0.4  0.1  0.3  0.6 0.0  0.5  1.4  1.4 -2.9  1.4 -0.4
## 2008  1.2 -2.6  3.5  1.3 -0.1 0.1 -0.1 -0.5 -0.6  1.9 -8.8  2.2
## 2009 -1.9 -1.0  1.8 -0.3  3.0 0.5  0.0 -1.7 -2.2 -2.4  1.2 -0.8
## 2010  1.1  1.1  0.7 -1.8 -1.4 0.4  0.5  2.6  0.4 -0.3  1.1  0.0
## 2011  1.1 -1.1  0.5 -0.4 -1.9 1.4 -0.6 -1.1 -1.6 -2.2 -0.3 -0.4
## 2012  0.9  0.6  0.4 -0.5 -2.3 1.3 -0.9  0.3  0.5  2.2  0.1  1.5
## 2013  0.4  0.3 -0.2 -0.8 -1.0 0.5  1.5 -0.7  0.8  0.2 -0.2  0.2
## 2014 -0.4  0.4  0.7  0.3  0.4 0.5 -0.3  0.1 -1.2  0.6 -1.0 -0.4
## 2015 -1.9 -0.1 -1.0  1.5  0.2 0.7 -0.2 -2.4  0.5  0.4  0.7 -0.7
## 2016  1.0  1.8 -0.1 -0.4  0.1 0.4 -0.5  0.3  1.0 -1.1  0.0 -0.4
## 2017 -0.4  0.6 -0.4 -0.5  1.0 1.4  0.5  0.5 -0.2  0.1 -0.2 -0.3
## 2018  0.0 -0.7  1.3 -0.3  0.7 1.0 -0.6  0.3  0.1  1.0  1.1  1.2
## 2019 -0.2  0.4  1.0 -0.8 -1.5 0.6 -1.9 -0.1 -0.7  0.6 -0.5  0.3
## 2020 -2.8  0.1   NA   NA   NA  NA   NA   NA   NA   NA   NA   NA
##      portfolio.returns
## 2007               2.5
## 2008              -3.0
## 2009              -3.8
## 2010               4.6
## 2011              -6.6
## 2012               4.2
## 2013               1.2
## 2014              -0.3
## 2015              -2.4
## 2016               2.1
## 2017               2.1
## 2018               5.1
## 2019              -3.0
## 2020              -2.7
```

This section shows the NYSE 2015 stock portfolio return.

```
CAPM.beta(NYSE_2015_portfolioReturn, benchmarkReturns, 0.035/252)
```

```
## [1] 0.9089286
```

```
CAPM.jensenAlpha(NYSE_2015_portfolioReturn, benchmarkReturns, 0.035/252)
```

```
## [1] -0.05055032
```

```
SharpeRatio(NYSE_2015_portfolioReturn, 0.035/252)
```

```
##                                portfolio.returns
## StdDev Sharpe (Rf=0%, p=95%):      -0.012065988
## VaR Sharpe (Rf=0%, p=95%):         -0.006999366
## ES Sharpe (Rf=0%, p=95%):          -0.004045242
```

```
table.AnnualizedReturns(NYSE_2015_portfolioReturn)
```

```
##                            portfolio.returns
## Annualized Return                    -0.0062
## Annualized Std Dev                    0.1535
## Annualized Sharpe (Rf=0%)            -0.0404
```

```
table.CalendarReturns(NYSE_2015_portfolioReturn)
```

```
##       Jan  Feb  Mar  Apr May Jun  Jul  Aug  Sep  Oct  Nov  Dec
## 2014   NA   NA   NA   NA  NA  NA   NA   NA   NA   NA   NA -0.4
## 2015 -1.7  0.1 -0.8  1.2 0.0 0.4  0.0 -2.9  2.6 -0.5  0.0 -0.7
## 2016  0.6  1.8 -0.6 -0.7 0.5 0.4 -0.6 -0.6  1.0  0.2 -0.5 -0.5
## 2017  0.2   NA  0.0 -0.1 1.4 0.3 -0.1  0.9  0.1  0.1 -0.2 -0.5
## 2018  0.2 -1.3  1.2 -0.4 0.4 0.1 -1.1  0.1 -0.2  0.7  0.7  1.0
## 2019 -0.2  0.5  1.2 -0.7 0.4  NA   NA   NA   NA   NA   NA   NA
## 2020   NA -4.0   NA   NA  NA  NA   NA   NA   NA   NA   NA   NA
##      portfolio.returns
## 2014              -0.4
## 2015              -2.5
## 2016               0.8
## 2017               2.1
## 2018               1.4
## 2019               1.2
## 2020              -4.0
```

The next section shows the NASDAQ portfolio return for 2007.

```
CAPM.beta(NASDAQ_2007_portfolioReturn, benchmarkReturns, 0.035/252)
```

```
## [1] 0.9353663
```

```
CAPM.jensenAlpha(NASDAQ_2007_portfolioReturn, benchmarkReturns, 0.035/252)
```

```
## [1] 0.06146454
```

```
SharpeRatio(NASDAQ_2007_portfolioReturn, 0.035/252)
```

```
##                                portfolio.returns
## StdDev Sharpe (Rf=0%, p=95%):         0.02579675
## VaR Sharpe (Rf=0%, p=95%):            0.01619773
## ES Sharpe (Rf=0%, p=95%):             0.00964793
```

```
table.AnnualizedReturns(NASDAQ_2007_portfolioReturn)
```

```
##                           portfolio.returns
## Annualized Return                    0.1071
## Annualized Std Dev                   0.2249
## Annualized Sharpe (Rf=0%)            0.4761
```

```
table.CalendarReturns(NASDAQ_2007_portfolioReturn)
```

```
##       Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
## 2007  0.3 -0.3  0.3  0.0 -0.3 -0.5 -0.5  1.6  2.3 -2.7 -0.3 -0.7
## 2008 -0.9 -3.0  4.5  1.5 -0.7  1.0 -0.6 -1.4 -0.8  2.4 -9.0  1.7
## 2009 -0.2  0.9  0.8 -1.8  3.3  0.0 -0.3 -2.2 -2.2 -2.3  0.9 -1.1
## 2010 -0.8  2.0 -0.1 -2.5 -1.4  1.0  1.2  4.2 -1.3 -0.8  0.5 -1.3
## 2011  0.7 -1.3  0.9 -0.2 -1.8  1.6 -0.4 -1.0 -1.3 -1.6  1.4 -0.9
## 2012  0.0  1.2 -0.4  0.3 -2.6  2.3 -1.0  0.0  0.4 -1.1  0.3  2.0
## 2013  0.4  0.7 -1.5 -1.4 -0.4  1.7  1.5 -1.1  2.1 -0.1  0.4  0.5
## 2014 -2.5  0.0  1.8  0.7  0.3  2.2 -0.4  0.1 -1.6  1.7 -1.4 -0.4
## 2015  1.7  0.6 -0.6  1.0  0.5  0.8  0.7 -5.0  1.0  0.4  1.5 -1.5
## 2016  0.4  3.2  1.2  1.7  0.0  1.0  0.8 -0.1  1.5 -0.8 -0.4 -1.1
## 2017 -0.2  0.4 -0.1  0.8  0.6 -0.3  0.5  0.3  0.5  0.1 -0.6 -0.8
## 2018 -2.0 -0.6  2.0  0.4  1.3 -0.5  0.2  0.1  0.5  2.6  0.3  2.2
## 2019 -1.9  0.6  1.6 -0.4 -1.3  1.3 -1.3 -0.3 -0.4  0.6 -0.7  0.0
## 2020  1.3 -0.2   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA
##      portfolio.returns
## 2007              -0.9
## 2008              -5.8
## 2009              -4.2
## 2010               0.5
## 2011              -3.8
## 2012               1.4
## 2013               2.8
## 2014               0.5
## 2015               1.1
## 2016               7.4
## 2017               1.0
## 2018               6.6
## 2019              -2.1
## 2020               1.1
```

The next section shows the NASDAQ portfolio return for 2019.

```
CAPM.beta(NASDAQ_2019_portfolioReturn, benchmarkReturns, 0.035/252)

## [1] 0.8379404

CAPM.jensenAlpha(NASDAQ_2019_portfolioReturn, benchmarkReturns, 0.035/252)

## [1] 0.1472888

SharpeRatio(NASDAQ_2019_portfolioReturn, 0.035/252)

##                               portfolio.returns
## StdDev Sharpe (Rf=0%, p=95%):        0.04272244
## VaR Sharpe (Rf=0%, p=95%):           0.03789039
## ES Sharpe (Rf=0%, p=95%):            0.03073317

table.AnnualizedReturns(NASDAQ_2019_portfolioReturn)

##                           portfolio.returns
## Annualized Return                    0.1882
## Annualized Std Dev                   0.2466
## Annualized Sharpe (Rf=0%)            0.7633

table.CalendarReturns(NASDAQ_2019_portfolioReturn)
```

```
##       Jan  Feb Mar Apr May Jun Jul Aug  Sep Oct Nov Dec portfolio.returns
## 2019   NA   NA  NA  NA  NA  NA  NA  NA -1.7 0.2 0.5 3.1               2.0
## 2020 -1.3 -0.7  NA  NA  NA  NA  NA  NA   NA  NA  NA  NA              -1.9
```

```
library(dplyr)
library(quantmod)
library(PerformanceAnalytics)
library(imputeTS)
library(PortfolioAnalytics)
library(ROI)
library(ROI.plugin.quadprog)
library(ROI.plugin.glpk)
```

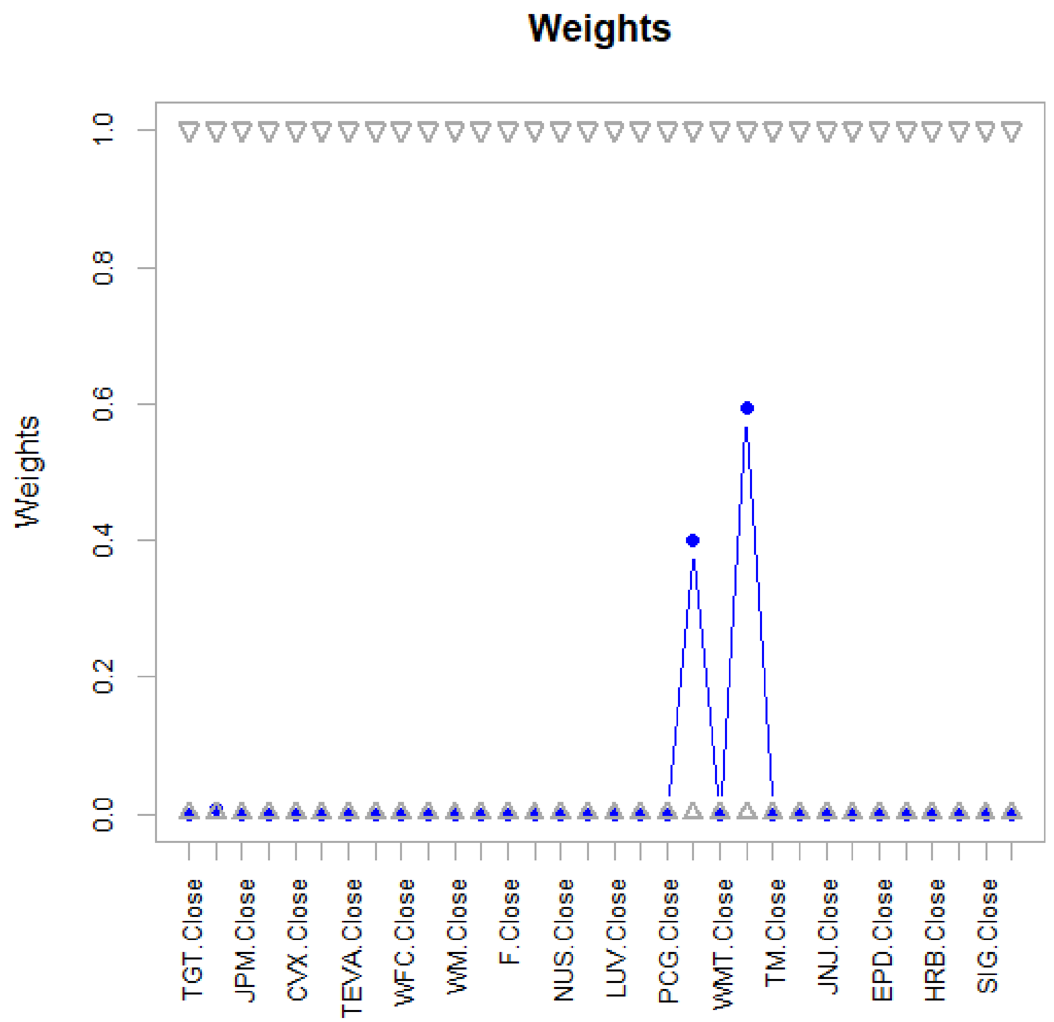
Calculate daily change in each column.

```
benchmarkReturns <- na.omit(ROC(benchmarkPrices))
```

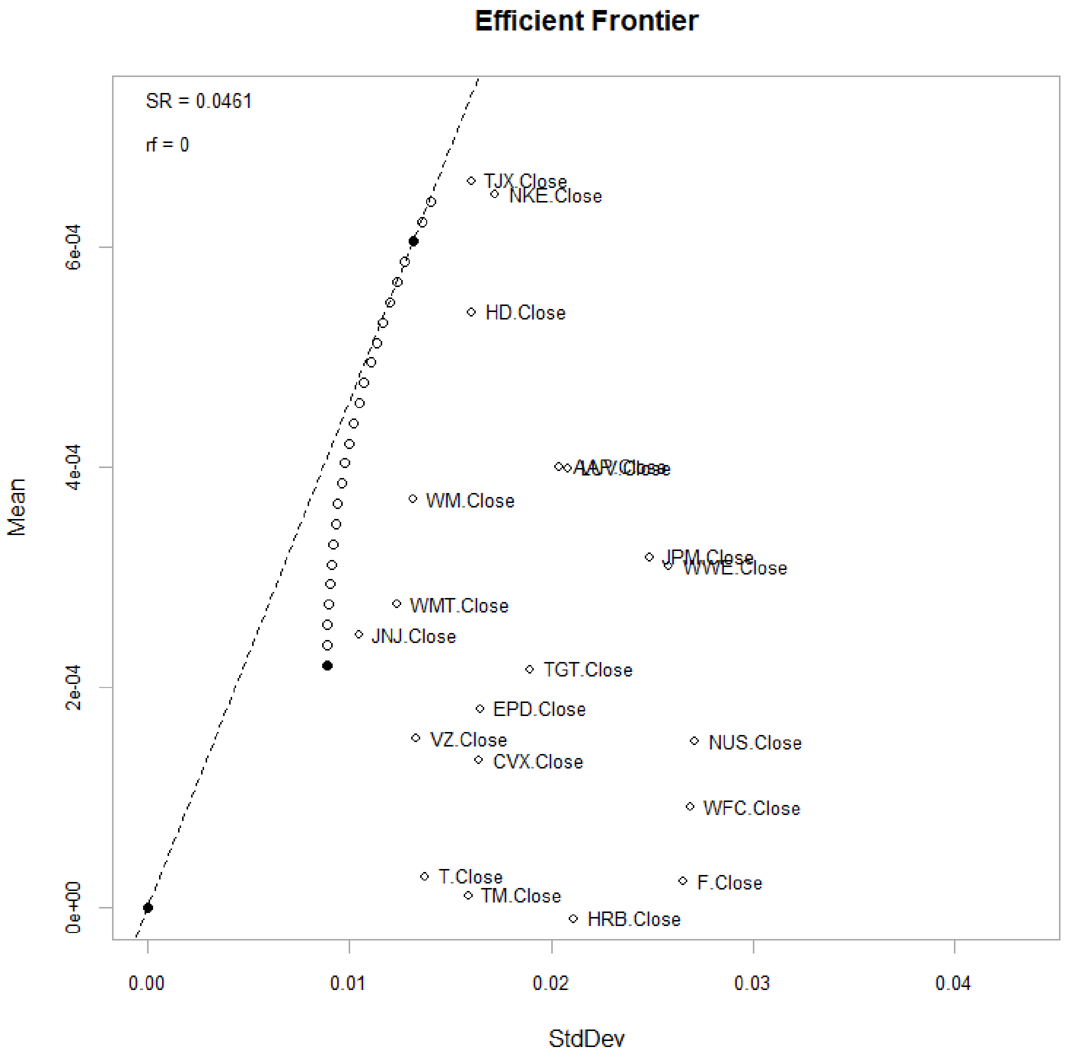
- NYSE_2007_portfolioReturn
- NYSE_2015_portfolioReturn
- NASDAQ_2007_portfolioReturn
- NASDAQ_2019_portfolioReturn

NYSE_2007_portfolioReturn:

```
portNYSE_2007 <- portfolio.spec(colnames(NYSE_2007_portfolioReturns))

portNYSE_2007 <- add.constraint(portNYSE_2007, type="weight_sum", min_sum=1,
max_sum=1)
portNYSE_2007 <- add.constraint(portNYSE_2007, type="box") #, min=.10,
max=.40)
portNYSE_2007 <- add.objective(portNYSE_2007, type="return", name="mean")
portNYSE_2007 <- add.objective(portNYSE_2007, type="risk", name="StdDev")

optPort <- optimize.portfolio(NYSE_2007_portfolioReturns, portNYSE_2007,
                              optimize_method = "ROI", trace=TRUE)

chart.Weights(optPort)
```

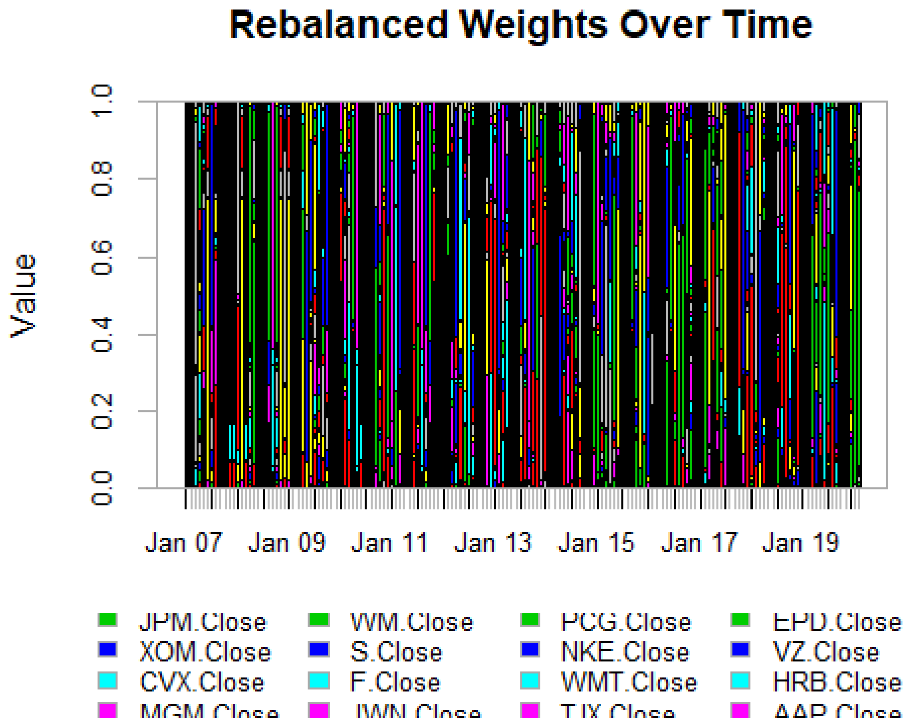


```
ef <- extractEfficientFrontier(optPort, match.col = "StdDev", n.portfolios =
25,
                         risk_aversion = NULL)

## Warning: executing %dopar% sequentially: no parallel backend registered

chart.EfficientFrontier(ef,
                        match.col = "StdDev", n.portfolios = 25, xlim = NULL,
ylim = NULL,
                        cex.axis = 0.8, element.color = "darkgray", main =
"Efficient Frontier",
                        RAR.text = "SR", rf = 0, tangent.line = TRUE,
cex.legend = 0.8,
                        chart.assets = TRUE, labels.assets = TRUE, pch.assets
= 21,
                        cex.assets = 0.8)
```

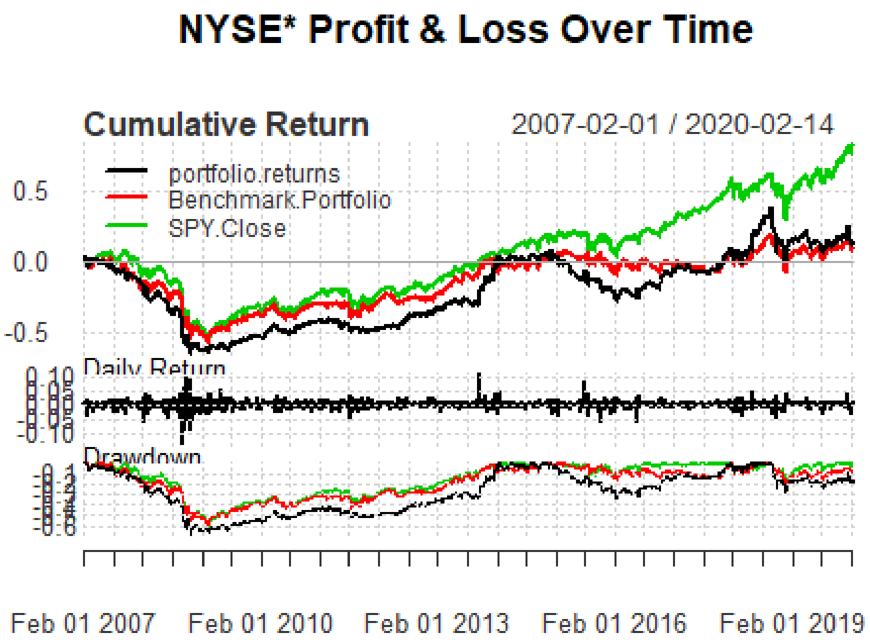


```r
rp <- random_portfolios(portNYSE_2007, 10000, "sample")

opt_rebal <- optimize.portfolio.rebalancing(NYSE_2007_portfolioReturns,
                                            portNYSE_2007,
                                            optimize_method="random",
                                            rp=rp,
                                            rebalance_on="months",
                                            training_period=1,
                                            rolling_window=10)

equal_weight <- rep(1 / ncol(NYSE_2007_portfolioReturns),
                    ncol(NYSE_2007_portfolioReturns))

benchmark <- Return.portfolio(NYSE_2007_portfolioReturns,
                              weights = equal_weight)

colnames(benchmark) <- "Benchmark Portfolio"
```

```
sp500prices <- getSymbols.yahoo("SPY", from='2007-01-03', periodicity =
'daily', auto.assign=FALSE)[,4]
sp500Rets <- na.omit(ROC(sp500prices))
sp500Rets <- as.xts(sp500Rets)

chart.Weights(opt_rebal, main="Rebalanced Weights Over Time")
```



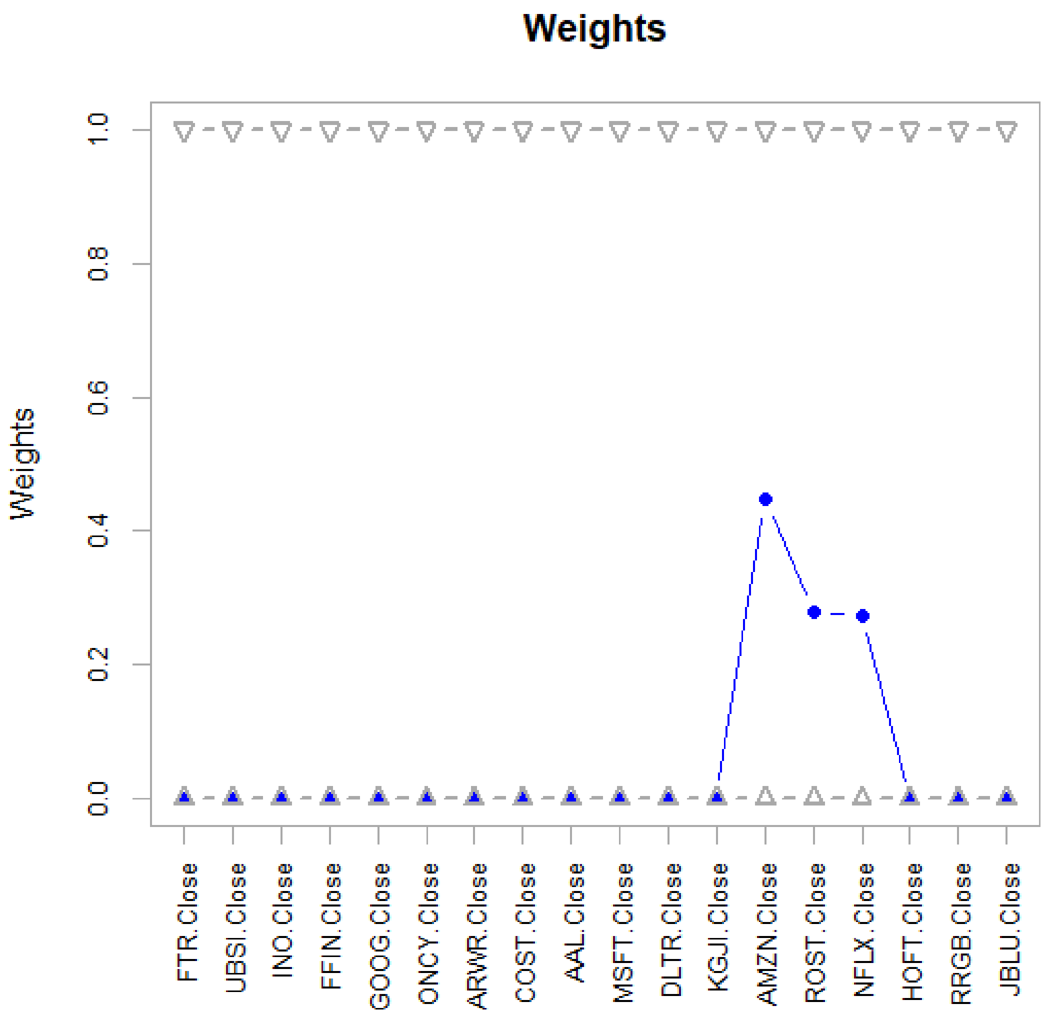
```
rebal_weights <-extractWeights(opt_rebal)

rebal_returns <- Return.portfolio(NYSE_2007_portfolioReturns,
                                  weights=rebal_weights)

rets_df <- cbind(rebal_returns, benchmark, sp500Rets)

charts.PerformanceSummary(rets_df, main="NYSE* Profit & Loss Over Time")
```



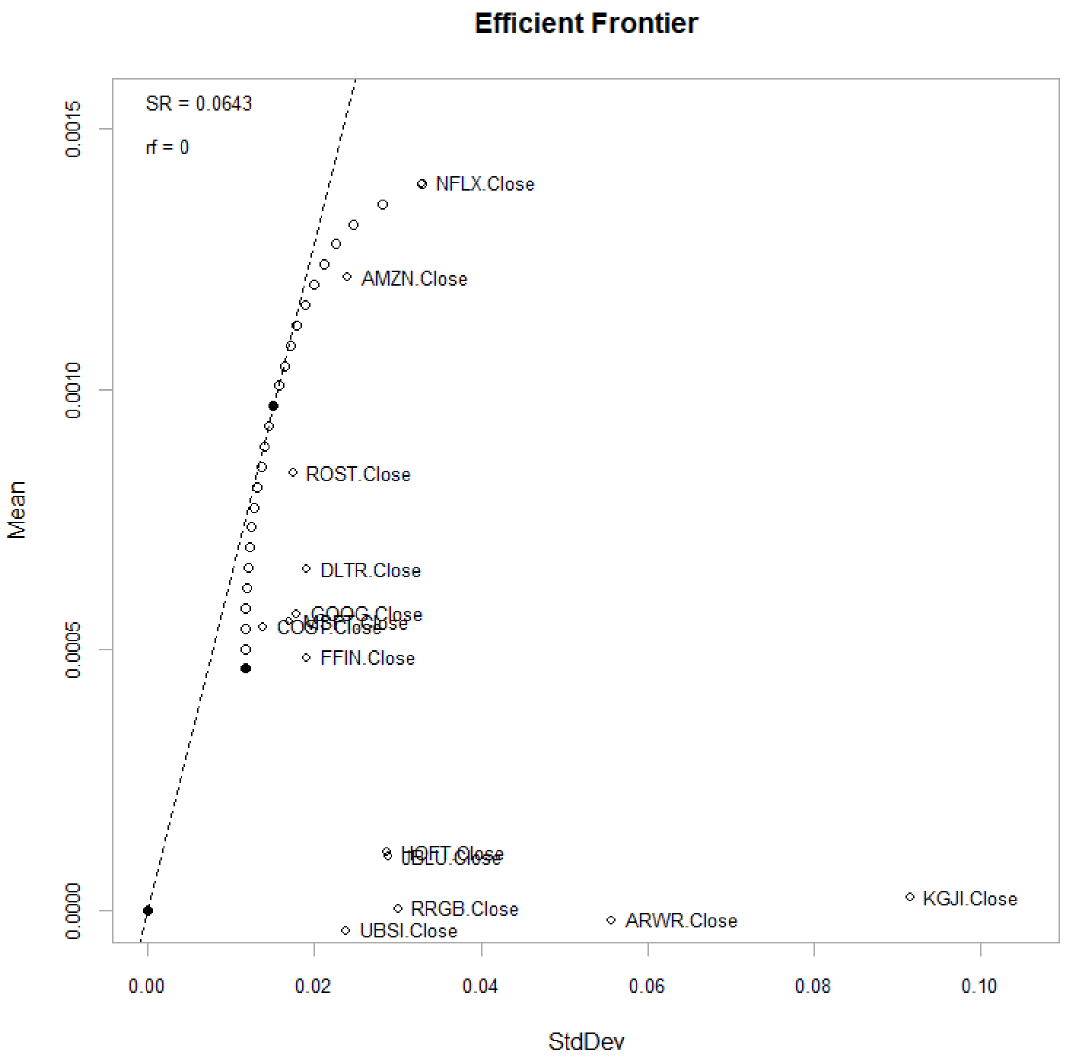
As you can see above for the NYSE hand selected stocks analyzed with this tutorial, that the benchmark portfolio was better than this portfolio but not as good as the S&P 500 stocks. We will see how the NASDAQ stock compare. Because these NYSE stocks were below zero for cumulative returns from 2007 until 2020 where they just broke even or had a slight positive cumulative return.

NASDAQ_2007_portfolioReturn

```
portNASDAQ_2007 <- portfolio.spec(colnames(NASDAQ_2007_portfolioReturns))

portNASDAQ_2007 <- add.constraint(portNASDAQ_2007, type="weight_sum",
min_sum=1, max_sum=1)
portNASDAQ_2007 <- add.constraint(portNASDAQ_2007, type="box") #, min=.10,
max=.40)
portNASDAQ_2007 <- add.objective(portNASDAQ_2007, type="return", name="mean")
portNASDAQ_2007 <- add.objective(portNASDAQ_2007, type="risk", name="StdDev")

optPort <- optimize.portfolio(NASDAQ_2007_portfolioReturns, portNASDAQ_2007,
                              optimize_method = "ROI", trace=TRUE)

chart.Weights(optPort)
```

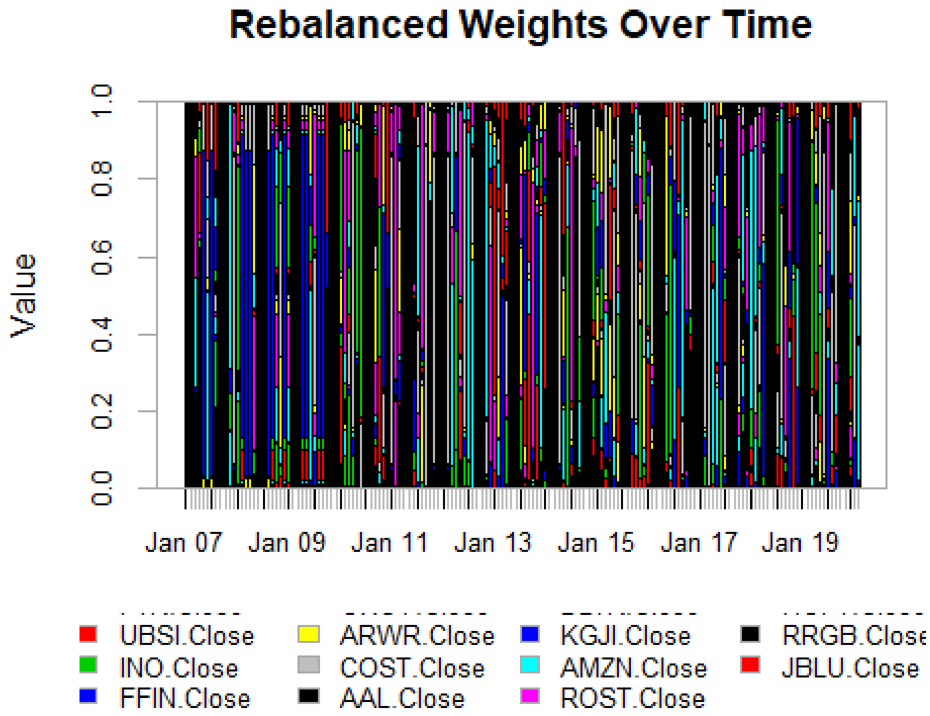


```
ef <- extractEfficientFrontier(optPort, match.col = "StdDev", n.portfolios =
25,
                         risk_aversion = NULL)

chart.EfficientFrontier(ef,
                        match.col = "StdDev", n.portfolios = 25, xlim = NULL,
ylim = NULL,
                        cex.axis = 0.8, element.color = "darkgray", main =
"Efficient Frontier",
                        RAR.text = "SR", rf = 0, tangent.line = TRUE,
cex.legend = 0.8,
                        chart.assets = TRUE, labels.assets = TRUE, pch.assets
= 21,
                        cex.assets = 0.8)
```

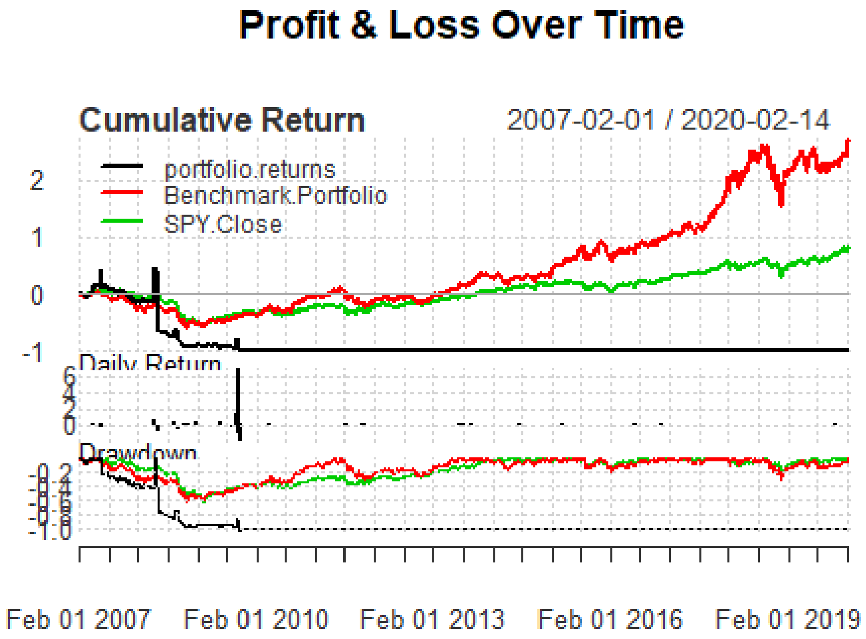


```
rp <- random_portfolios(portNASDAQ_2007, 10000, "sample")

opt_rebal <- optimize.portfolio.rebalancing(NASDAQ_2007_portfolioReturns,
                                            portNASDAQ_2007,
                                            optimize_method="random",
                                            rp=rp,
                                            rebalance_on="months",
                                            training_period=1,
                                            rolling_window=10)

equal_weight <- rep(1 / ncol(NASDAQ_2007_portfolioReturns),
                    ncol(NASDAQ_2007_portfolioReturns))

benchmark <- Return.portfolio(NASDAQ_2007_portfolioReturns,
                              weights = equal_weight)

colnames(benchmark) <- "Benchmark Portfolio"
```

```
sp500prices <- getSymbols.yahoo("SPY", from='2007-01-03', periodicity =
'daily', auto.assign=FALSE)[,4]
sp500Rets <- na.omit(ROC(sp500prices))
sp500Rets <- as.xts(sp500Rets)

chart.Weights(opt_rebal, main="Rebalanced Weights Over Time")
```



```
rebal_weights <-extractWeights(opt_rebal)

rebal_returns <- Return.portfolio(NASDAQ_2007_portfolioReturns,
                                  weights=rebal_weights)

rets_df <- cbind(rebal_returns, benchmark, sp500Rets)

charts.PerformanceSummary(rets_df, main="Profit & Loss Over Time")
```



Looking at the above, the portfolio returns for the NASDAQ hand selected stock were far below the benchmark portfolio and the S&P 500 close as well as below zero cumulative return from 2009 to 2019. This makes it a terrible portfolio of stocks, but could be modified later by analyzing individual stocks, and points in time, and returns on investment (ROI) for different historical time frames of importance.

The NYSE portfolio was better than the NASDAQ stocks by just breaking even since the start at 2007 to 2020.