Project 2

Group 2 - Jessica Chung, Ishita Ishita, Chloe Kwon, Brett Scroggins

Due 2/28/17

Calculate Daily Returns for Each Stock

```
### Calculate Daily Returns
returnsMat <- diff(priceMat)/priceMat[-nrow(priceMat),]
```

Calculate Correlation Matrix

```
### Calculate the Correlation Matrix
rho <- cor(returnsMat, use = "complete.obs")
```

Code the Integer Program

```
constructFunds <- function(rh,q,pMat,sMat,ut,ud){
  library("lpSolve")
  #set n to be the number of stocks we're looking at
  n <- length(ut)
  Q <- q
  #Create the y constraint portion of the A matrix
  y_constraint <- c(rep(0,n*n),rep(1,n))
  y_constraint <- matrix(y_constraint, 1, n*n+n)
  #Create the x constraint portion of the A matrix
  x_constraint <- c()
  for (i in 1:n) {
    x_constraint <- append(x_constraint,rep(0, n*(i-1)))
    x_constraint<- append(x_constraint,rep(1, n))
    x_constraint <- append(x_constraint,rep(0, n*(n-i)+n))

  }
  x_constraint <- matrix(x_constraint, byrow = TRUE,n, n*n+n)
  #Create the xy constraint portion of the A matrix
  xy_x <- diag(n*n)
  xy_y <- rep(c(diag(x=-1,n)),n)
  xy_constraint <- cbind(xy_x, matrix(xy_y, byrow=TRUE,n*n,n))


  A <- rbind(y_constraint,x_constraint,xy_constraint)

  b <- c(Q,rep(1,n),rep(0,n*n))

  dir <- c(rep("=",n+1),rep("<=",n*n))

  c <- c(rh,rep(0,n))
```

```r
  sol <- lp("max",c,A,dir,b,binary.vec = 1:((n^2)+n))

  # Calculate the weights
  xij_solution <- matrix(sol$solution[1:(n*n)],byrow = TRUE,n,n)
  valueMat <- sMat*pMat
  lastValueDate <- valueMat[nrow(valueMat),]
  wj <- colSums(xij_solution*lastValueDate)
  total <- sum(wj)
  wt <- wj/total
  return(wt)
}
```

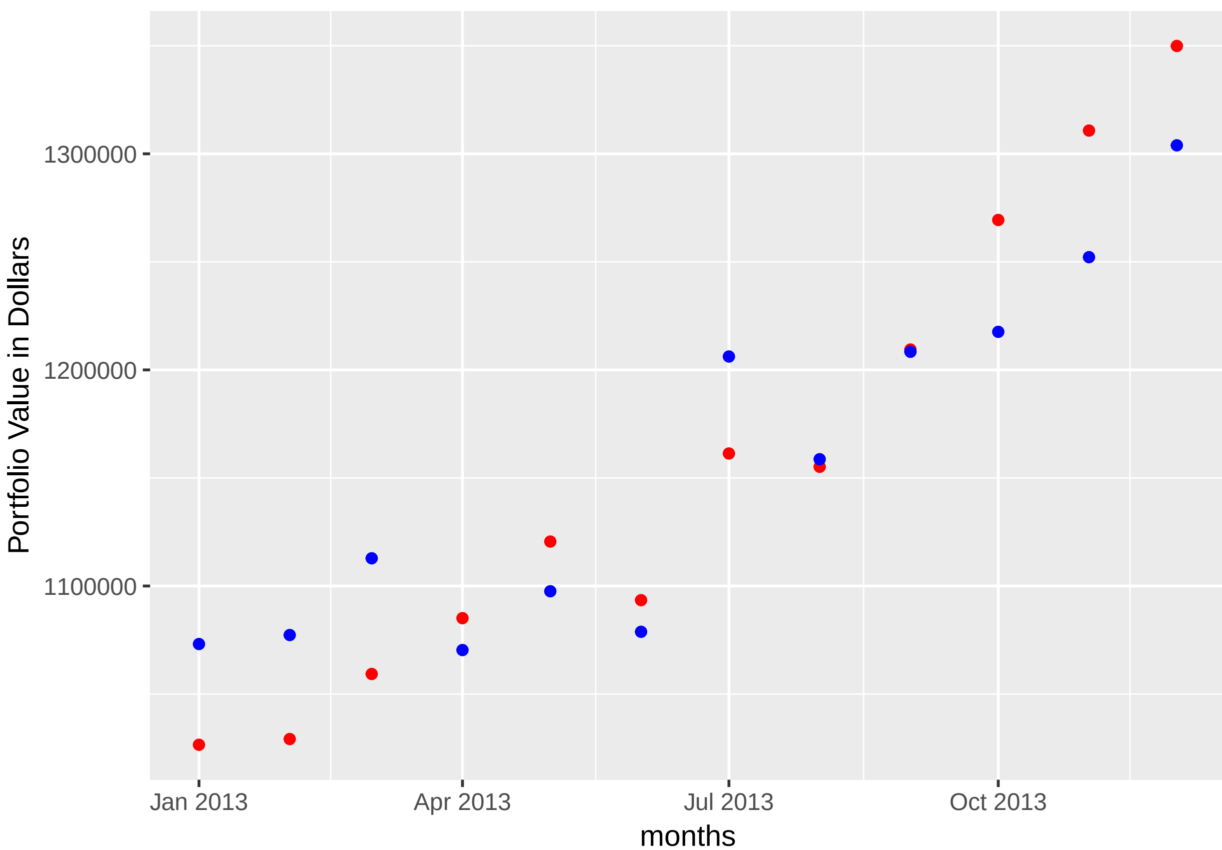
Use Weights to Construct Index Portfolio at the End of 2012

```r
## Create a function that returns the montly value of our index fund assuming a 1MM investment
getPerformance <- function(pMat,wts, mtpMat){
  lastPriceDate <- pMat[nrow(pMat),]
  portfolio <- wts*1000000
  portfolioShares <- portfolio/lastPriceDate
  portfolioReturn <- c()
  for (i in c(1:12)){
    portfolioReturn<-append(portfolioReturn,sum(portfolioShares*mtpMat[i,]))
  }
  return(portfolioReturn)
}

q = 25
weights <- constructFunds(rho, q, priceMat, sharesMat,unique_tickers, unique_dates)
#Stocks in our portfolio
subset(data.frame(unique_tickers,weights), weights!=0)
```

```
##    unique_tickers      weights
## 3             ADI 0.2964807551
## 4             ADP 0.1080383349
## 11           AMZN 0.1390916618
## 15           BIDU 0.0093990660
## 19           CELG 0.0488368463
## 27           CTRX 0.0033017861
## 29           CTXS 0.2053599706
## 36           ESRX 0.0150775179
## 39           FAST 0.0047217151
## 40             FB 0.0151813120
## 44           GMCR 0.0020976887
## 53           KRFT 0.0257411154
## 57           LMCA 0.0122518221
## 60           MNST 0.0029896074
## 63            MWW 0.0002246126
## 66           NFLX 0.0017565763
## 71           ORLY 0.0065926647
## 73           PCAR 0.0258156869
## 76           REGN 0.0055596826
## 77           ROST 0.0041338535
```

```
## 78           SBAC 0.0030749654
## 87           TRIP 0.0018607787
## 93            VOD 0.0084671149
## 96            WDC 0.0117989434
## 97            WFM 0.0421459214
```

```r
#Overal Return
pReturn <- getPerformance(priceMat, weights, monthlyPriceMat)

## Calculate the monthly value of the NASDAQ assuming a 1MM investment
nasdaq <- c(2731.53, 2738.58, 2818.69, 2887.44, 2981.76, 2909.60, 3090.19, 3073.81, 3218.20, 3377.73, 34
# calculate our index's return
total_investment = 1000000
shares <- total_investment/2660.93
nasdaqVal <- nasdaq*shares

## Plot NASDAQ vs Index Fund performance
months <- seq(as.Date("2013/1/1"), by = "month", length.out = 12)
library(ggplot2)
df <- data.frame(nasdaqVal,pReturn)
g <- ggplot(df,aes(months))
g <- g+geom_point(aes(y=nasdaqVal),color = "red")
g <- g+geom_point(aes(y=pReturn),color = "blue")
g <- g+ylab("Portfolio Value in Dollars")
g
```

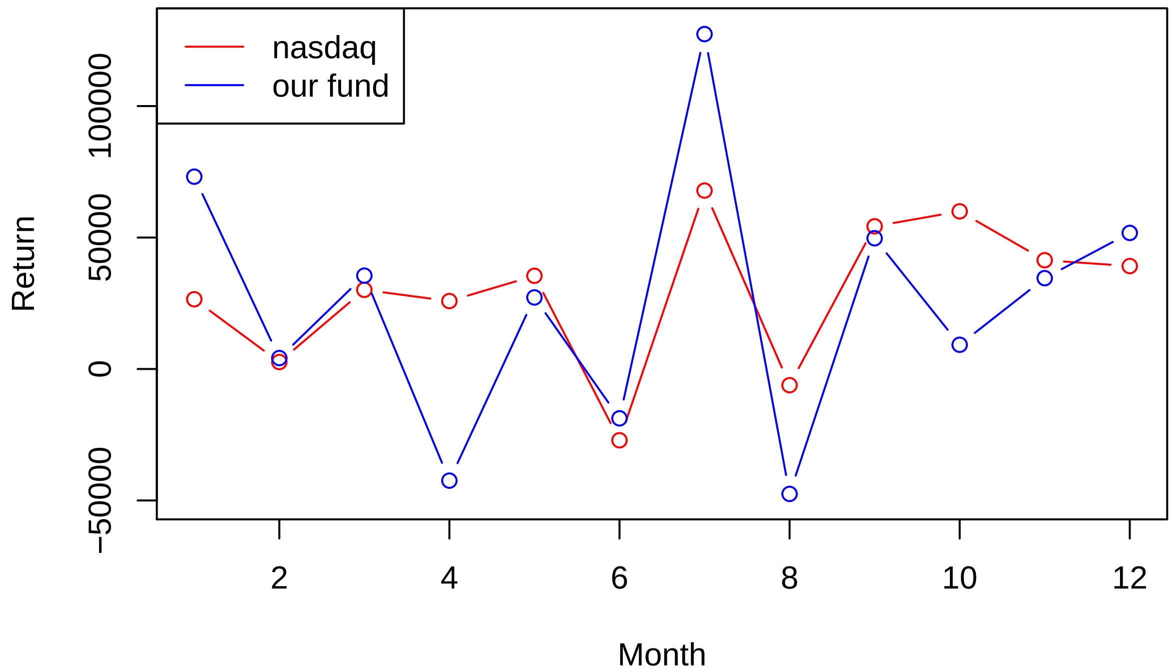



```r
##Plot real returns
monthlyPriceMat = matrix(NA, length(unique_mdates), length(unique_tickers))
```

```r
for (i in 1:length(unique_tickers)) {
  tic = unique_tickers[i]
  idx = is.element(unique_mdates, mdate[mticker == tic])
  monthlyPriceMat[idx, i] = mprice[mticker == tic]
}

monthlyPriceMat = rbind(priceMat[250,], monthlyPriceMat)


# calculate monthly return
returnMonthlyMat = matrix(NA, length(unique_mdates), length(unique_tickers)) #initialize matrix for dai

for (i in 1:length(unique_tickers)) {
  prices = monthlyPriceMat[, i]
  monthly_returns = diff(prices)
  returnMonthlyMat[, i] = monthly_returns
}


investment_vector = weights * total_investment
share_vector = investment_vector / priceMat[250,]
total_value = share_vector * monthlyPriceMat[1,]

real_return = NULL
for (i in 1:dim(returnMonthlyMat)[1]) {
  r = sum(share_vector * returnMonthlyMat[i,])
  real_return = c(real_return, r)
}
real_return
```

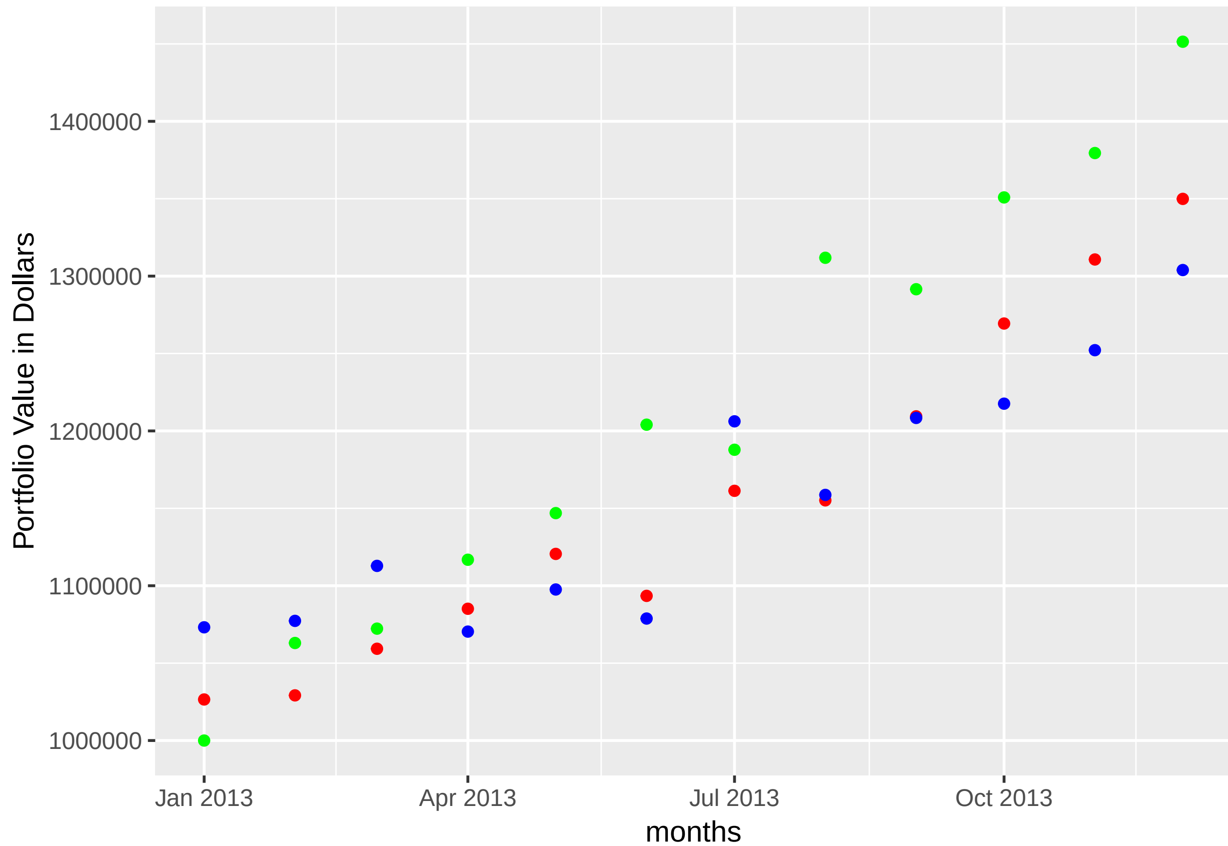
```
##  [1]  73152.376   4159.371  35517.915 -42461.182  27212.774 -18771.812
##  [7] 127382.091 -47517.114  49721.466   9218.980  34551.767  51753.977
```

```r
# calculate nasdaq return
nasdaq_2013 = c(2660.93, 2731.53, 2738.58, 2818.69, 2887.44, 2981.76, 2909.60, 3090.19, 3073.81, 3218.2
num_share_nasdaq = total_investment / nasdaq_2013[1]
nasdaq_monthly_return = NULL
for (i in 1:length(nasdaq_2013)) {
  monthly_return = nasdaq_2013[i+1] - nasdaq_2013[i]
  nasdaq_monthly_return = c(nasdaq_monthly_return, monthly_return)
}

nasdaq_return = nasdaq_monthly_return[1:12] * num_share_nasdaq

par(mfrow=c(1, 1))
plot(nasdaq_return, type='b', xlab='Month', ylab='Return', col='red', ylim=c(-50000,130000))
lines(real_return, type='b', col='blue')
legend('topleft',legend=c('nasdaq', 'our fund'), col=c('red', 'blue'), lty=c(1, 1))
```



Calculate New Similarity Matrix Using Correlation Between Stocks Using 20 Day Moving Averages

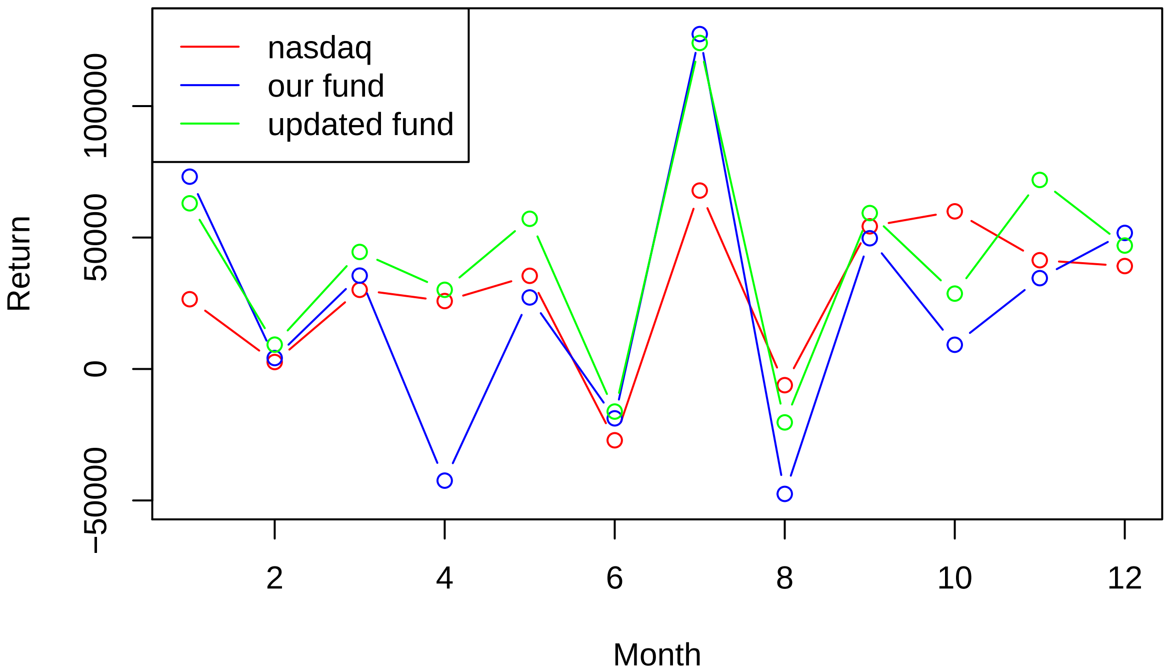
```
i <- 20
similarityMat <- function(pMat, sMat, ut,ud){

### Create New Similarity Matrix
csCol <- function(stock,i){
    stock[which(is.na(stock))] <- 0
    cs <- cumsum(stock)
    rsum <- (cs[(i+1):length(stock)] - cs[1:(length(stock) - i)]) / i
  }

  maStocks <- apply(pMat,2, csCol,20)
  corVarMat <- cor(maStocks,maStocks)
  return(corVarMat)
}

rho_new <- similarityMat(priceMat,sharesMat,unique_tickers,unique_dates)
weights_new <- constructFunds(rho_new,q,priceMat,sharesMat,unique_tickers,unique_dates)
newpReturn <- getPerformance(priceMat,weights_new, monthlyPriceMat)
df <- data.frame(nasdaqVal,pReturn, newpReturn)
g <- ggplot(df,aes(months))
g <- g+geom_point(aes(y=nasdaqVal),color = "red")
g <- g+geom_point(aes(y=pReturn),color = "blue")
g <- g+geom_point(aes(y=newpReturn),color = "green")
g <- g+ylab("Portfolio Value in Dollars")
g
```




```r
investment_vector_new = weights_new * total_investment
share_vector_new = investment_vector_new / priceMat[250,]
total_value_new = share_vector_new * monthlyPriceMat[1,]

real_return_new = NULL
for (i in 1:dim(returnMonthlyMat)[1]) {
  r_new = sum(share_vector_new * returnMonthlyMat[i,])
  real_return_new = c(real_return_new, r_new)
}
real_return_new
```

```
##  [1]  63015.253   9269.211  44512.520  30117.911  57121.185 -16196.694
##  [7] 124010.944 -20308.488  59303.685  28669.435  71903.871  46949.189
```

```r
par(mfrow=c(1, 1))
plot(nasdaq_return, type='b', xlab='Month', ylab='Return', col='red', ylim=c(-50000,130000))
lines(real_return, type='b', col='blue')
lines(real_return_new, type='b', col='green')
legend('topleft',legend=c('nasdaq', 'our fund', 'updated fund'), col=c('red', 'blue','green'), lty=c(1,
```



Comparing Models

We decided to use a rolling 20 day average stock price to calculate a new correlation matrix. This metric was selected as it smooths volitility to give a clearer picture when determining a more optimal allocation of stocks to purchase based on their potential similarity. What can be seen by the output is that this returned the highest amount of profit of any of the three choices because the new similarity matrix took into account the stocks' relevant histories.