FINM3422_Final Individual Assignment

Regan Smith - Student Number: 44853189

12/10/2020

Before starting the assignment, the packages for various functions must be installed. Here, we have a # sign before the calls as the packages are already installed.

```
#install.packages('quantmod')
#install.packages('xts')
#install.packages("ggplot2")
#install.packages("data.table")
#install.packages("gridExtra")
#install.packages("knitr")
#install.packages("zoo")
#install.packages("psych")
#install.packages("fBasics") ## online download, needed to find summary
statistics
#install.packages("kableExtra") ## online download, needed for Kable function
#install.packages("tidyverse") ## online download, needed for pull() function
#install.packages("lubridate")
#install.packages("car")
#install.packages("dplyr")
#install.packages("TimeWarp")
#install.packages("fPortfolio")
#install.packages("tseries")
#install.packages("boot")
#install.packages("Matrix")
#install.packages("matrixcalc")
#install.packages("pracma")
```

Next, we must load the installed packages using the 'library' function.

```
library(xts)
library(quantmod)
library(data.table)
library(ggplot2)
library(gridExtra)
library(fBasics)
library(knitr)
library(zoo)
library(psych)
library(fBasics)
library(kableExtra)
library(tidyverse)
library(lubridate)
library(car)
```

```
library(dplyr)
library(TimeWarp)
library(fPortfolio)
library(tseries)
library(boot)
library(Matrix)
library(matrixcalc)
library(pracma)
```

Question 1

a.

```
nfunds <- 4
ID <- rep("Fund no.",nfunds)
Styles <- c("Large growth fund","Large value fund","Small value fund","Small
growth fund")

table <- data.table(ID,Mysterious_Fund = c(1:4),Styles)
print(table)

##          ID Mysterious_Fund            Styles
## 1: Fund no.               1 Large growth fund
## 2: Fund no.               2  Large value fund
## 3: Fund no.               3  Small value fund
## 4: Fund no.               4 Small growth fund
```

b. *Briefly explain the reasoning behind your answers above.*

Answer. We know from the Fama French Model that SMB measures the difference between the returns of small stocks compared to that of larger stocks (size determined by market capitalization). Therefore over the course of the regression sample period the data for SMB measures this. The reason this measure was included in the Fama French Model is because historically, small stocks tend to outperform the larger stocks.

Once, the regression is complete we have the estimates for the factor loadings of SMB (γ) for each fund. If the factor loading is positive then we expect the recorded stock to be smaller as the stock's returns are correlated with the returns of small stocks relative to big stocks. Oppositely, if the factor loading is negative than we expect this stock to be big as the stock is inversely correlated to the return of small stocks relative to big stocks.

Now in terms of HML, we know this term represents the difference in the returns of stocks with high book-to-market value compared to those with low book-to-market value. Note that stocks with high book-to-market are value stocks as the intrinsic value of company is higher than the value determined by the stock market. Again, this term is included in the Fama French model because stocks with high book-to-market value tend to outperform the market over the long term. Now on the other side, are companies with low book-to-market values. These are often companies such as tech stocks whose share price is much higher than the current intrinsic value of the company. As a result, investors choose these stocks with the hopes that the stocks will grow quickly in value, hence why these stocks are called growth stocks. Similar to the SMB factor loading, the HML factor loading (λ) will return positive for a given fund if the fund is a high book-to-market or 'value' fund. This is because the returns of the fund are correlated with the return of the broader array of high book-to-market 'value' funds. Alternately, if a given fund returns a negative value for the HML factor loading (λ), then this fund is a low book-to-market or growth stock as its returns are inversely correlated with the broader returns of value funds compared to growth funds.

In terms of the four funds listed. Fund 1 has negative values for both γ and λ, thus the stock is a large growth fund. Fund 2 is also a large fund due to a negative γ term, however the fund is a value fund because λ is positive. Fund 3 is similar to Fund 2 in that they are both value funds (positive λ), however Fund 3 is different to Fund 2 because it is a small fund (positive γ). Fund 4 is also small (positive γ) however it is a growth fund because λ is negative.

c. *With reference to the FF3 model, describe the purpose of including each of the factors and how the FF3 model aims to improve over the Capital Asset Pricing Model (CAPM).*

Answer. The Fama French was initially made by Eugene Fama and Kenneth French to explain the price of stocks returns, which is similar to the Capital Asset Pricing Model (Model). Apart from the additional factors, the Fama French relates to the CAPM model because they both have a coefficient β_1, i in front of the $R_m, t - R_f, t$ (expected term market minus risk free rate). This term measures the movement of the stock relative to the market. These models are different firstly due to their intercept term. The intercept term for the CAPM model is the risk free rate whilst the intercept term for the Fama French Model - α, measures the average equity premium. The Fama French Model also improves apon the CAPM model through inclusions of additional factor loadings to explain certain market phenomena.

Fama and French found that smaller stocks and value stocks tended to outperform the market and wanted to included these factors in a model. The CAPM Model does not include these factors and hence assumes that the price of a stock is solely dependent on its systematic risk relative to the market (as well as risk free rate). This Model has been criticized by a number of influential investors including Warren Buffet and Charlie Munger of Berkshire Hathaway because it assumes that the volatility of a stock is the correct determinant of true business risk. Although the Fama French Model does not completely solve this issue, it seeks to improve upon the CAPM model by spreading the explanation of return variation to the other factors loadings - SMB and HML. For instance, if we have two identical stocks but one stock is a value stock and the other is a growth stock, then it is possible that the stock price volatility may be similar (growth stocks tend to be more volatile however the case is still possible). As a result, the CAPM model would calculate the beta the same for each stock and expected returns would be the same for each. We know though, that value stocks tend to outperform growth stocks therefore the CAPM model is not capturing this phenomena properly. Again, the same could be true of the SMB factor where a large (but very volatile) company such as Telsa, Inc. would have the same beta as a small company (assume similiar volatility to Tesla, Inc but due to smaller size). As a result, the CAPM model would price these stocks to have an equal expected return (as systematic beta is the same due to similiar volatility) even though we know empirically that small stocks tend to outperform larger ones.

Question 2

a. , b.

Setting up the Dummy variable RD.

```
sourcedata            <- 'yahoo'
tickers               <- c('^GSPC')
sampleperiod          <- c('2008-12-25', '2020-01-10')
getSymbols(tickers, src=sourcedata, from=sampleperiod[1], to=sampleperiod[2])

## [1] "^GSPC"

marketdata <- monthlyReturn(GSPC$GSPC.Close, type='log')
# log monthly returns of S&P500 daily index (closing price)

idx <- which(index(marketdata)>="2009-01-02"&index(marketdata)<="2011-12-
31"|index(marketdata)>="2017-01-01"&index(marketdata)<="2019-12-31")
# creating an index of the observations in the marketdata variable
corresponding to dates in regime 1 and regime 2 only

marketdata <- marketdata[idx]
# inputting the index to return monthly returns within regime 1 and regime 2
only
```

c.

```
for(i in 1:nrow(marketdata))
  {
if(index(marketdata)[i]>="2017-01-01")
{
marketdata$RD[i]=1
 }
    else
    {
     marketdata$RD[i]=0
    }
}
# creating the Dummy variable RD stored in marketdata$RD
```

d. Bootstrapping.

```
S <- nrow(marketdata)
# we sample from regime 1 and regime 2 only
set.seed(1)
idx = sample(S, replace=TRUE)

# define the function first
ols.fn <- function(data, idx){
```

```
  d <- data[idx, ] # allows boot to select sample
  fit <- lm(marketdata$monthly.returns ~ marketdata$RD , d)
  # regress log monthly market returns (S&P500 close) against RD Dummy
(stored in marketdata$RD)
  output <- summary(fit)$coefficients
  return(output)
}
set.seed(1)
olsresult <- boot(data=marketdata$monthly.returns, statistic=ols.fn, R=2000)

#p-value alpha
pvalue_alpha <- olsresult$t[,1] - mean(olsresult$t[,1])
pvalue_alpha <- mean(abs(pvalue_alpha) > abs(olsresult$t0[1,1]))
print(pvalue_alpha)

## [1] 0

#p-value beta
pvalue_beta <- olsresult$t[,2] - mean(olsresult$t[,2])
pvalue_beta <- mean(abs(pvalue_beta) > abs(olsresult$t0[2,1]))
pvalue_beta

## [1] 0
```

Question 3.

a.

```
sourcedata           <- 'yahoo'
tickers              <- c('MSFT','MMM','BA','XOM','JPM','PG')
sampleperiod         <- c('1998-01-01', '2019-12-30')
getSymbols(tickers, src=sourcedata, from=sampleperiod[1], to=sampleperiod[2])

## pausing 1 second between requests for more than 5 symbols
## pausing 1 second between requests for more than 5 symbols

## [1] "MSFT" "MMM"  "BA"   "XOM"  "JPM"  "PG"

which(annualReturn(MSFT)!=annualReturn(MSFT$MSFT.Close))

## [1] 1

# annualReturn function yields Close as default.

MSFT <- annualReturn(MSFT)
MMM <- annualReturn(MMM)
BA <- annualReturn(BA)
XOM <- annualReturn(XOM)
JPM <- annualReturn(JPM)
PG <- annualReturn(PG)
# default annualReturn = arithmetic (sample raw).
```

b.

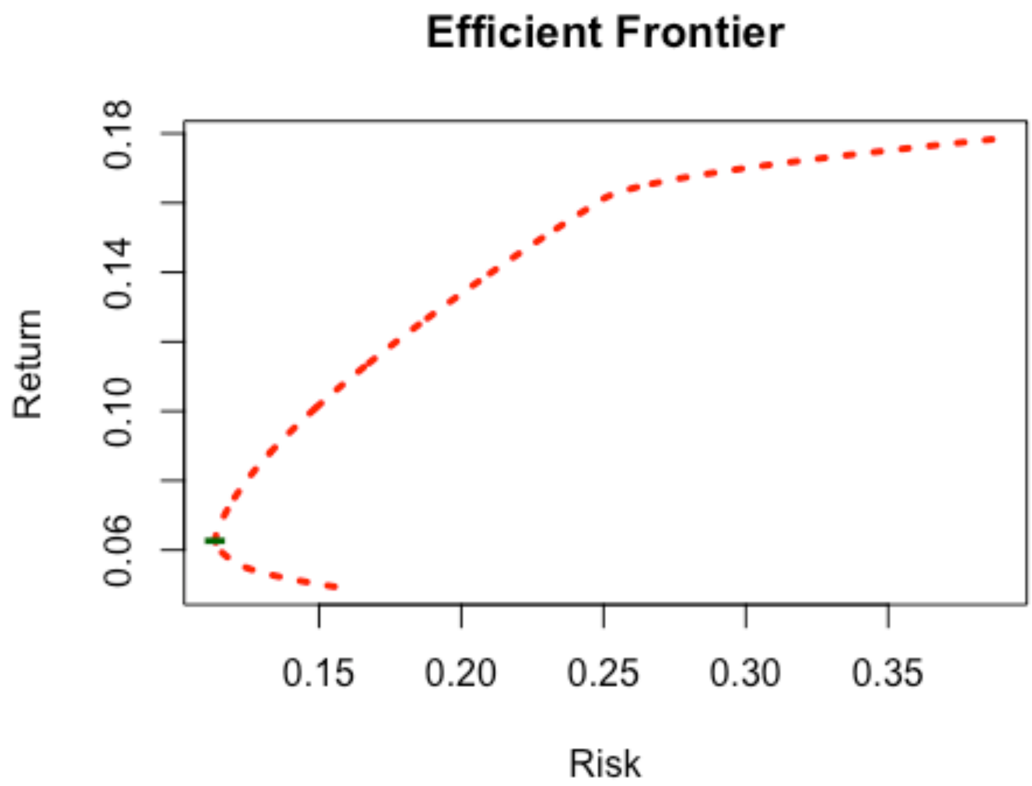
```
ret.mat = data.frame(MSFT,MMM,BA,XOM,JPM,PG)
colnames(ret.mat)=c("MSFT","MMM","BA","XOM","JPM","PG")
rownames(ret.mat)=NULL
ret = apply(ret.mat,2,mean, na.rm=T) #calculate expect returns based on
historical data

N = 1000 # create N target expected returns and optimize weight combination
to get minimum risk for each target return
ret_p = seq(min(ret), max(ret), length = N)
ret_p[seq(1,N,N/10)] # portfolio returns. 10 Obs.
```

```
##  [1] 0.0494320 0.0623423 0.0752526 0.0881628 0.1010731 0.1139834 0.1268937
##  [8] 0.1398040 0.1527143 0.1656246
```

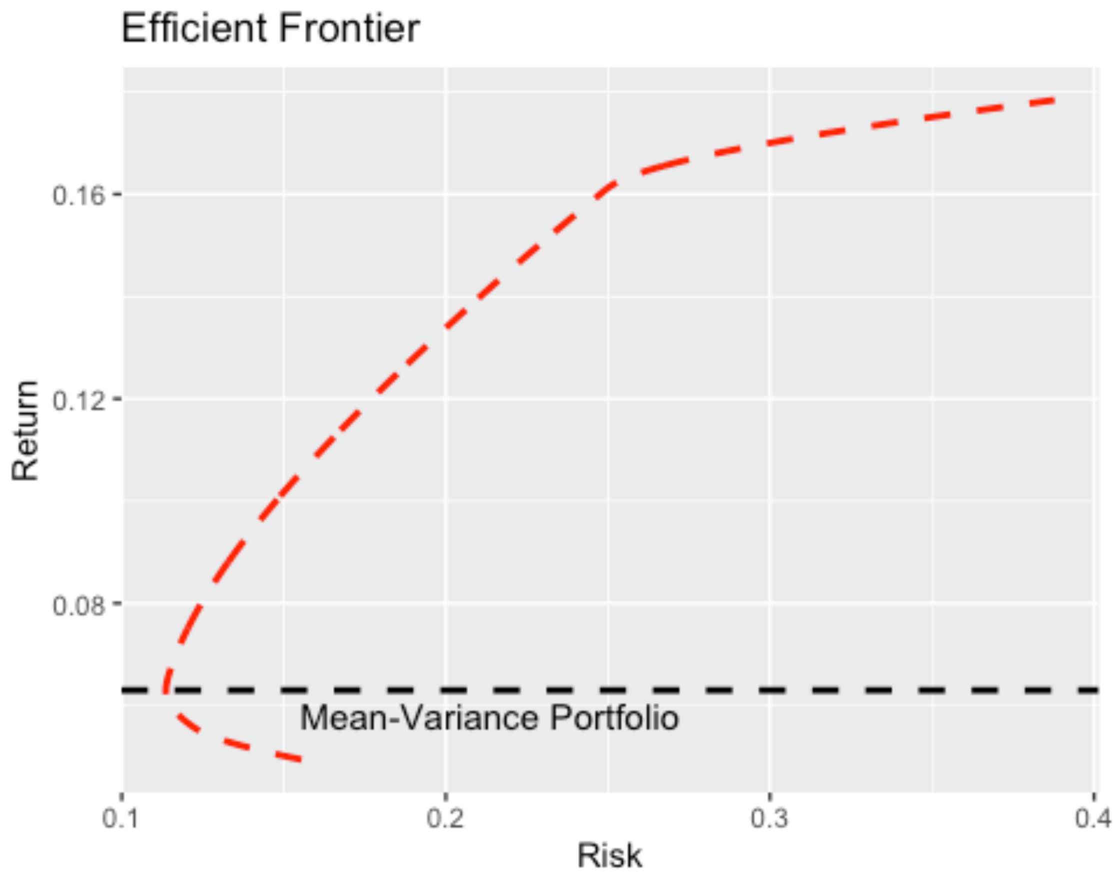
```
w_p = matrix(NA, nrow = N, ncol = ncol(ret.mat)) # define weight variables
var_p = rep(NA, length(N))  # define standard deviation variable
colnames(w_p) = colnames(ret.mat)
for (i in 1:N) {
temp_port = portfolio.optim(x = as.matrix(ret.mat), pm = ret_p[i])
#optimization
var_p[i] = temp_port$ps
w_p[i,] = temp_port$pw
}
```

Plot the efficient frontier with standard plot and ggplot

```
port = data.frame(cbind(w_p, var_p, ret_p))
minvar_p = port[port$var_p == min(port$var_p), ]
plot(var_p, ret_p, type = "l", lty = 3, lwd = 3, col = "red", xlab = "Risk",
ylab = "Return",
main = "Efficient Frontier")
points(minvar_p$var_p, minvar_p$ret_p, cex = 2, pch = "-", col = "dark
green")
```

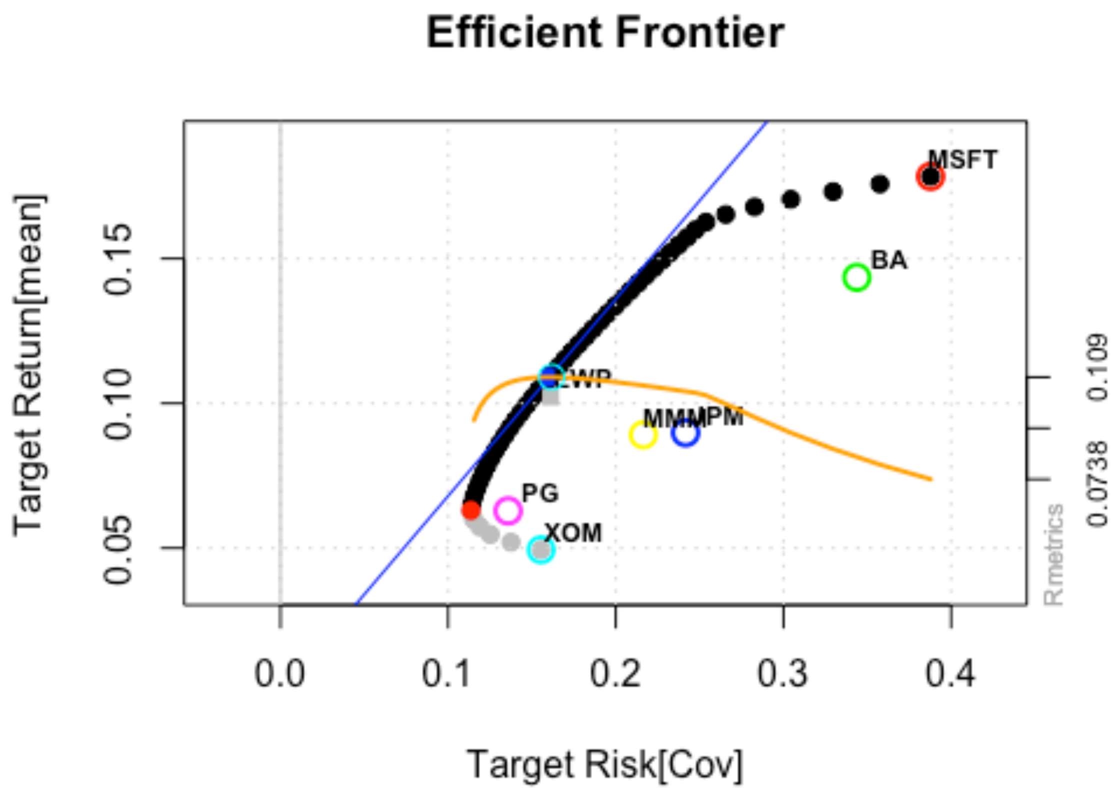



```r
ggplot(port, aes(x = var_p, y = ret_p)) + geom_path(colour = "red", size = 1,
lty = 2) + labs(x = "Risk", y = "Return", title = "Efficient Frontier") +
annotate("text", label = "Mean-Variance Portfolio", x = minvar_p$var_p + 0.1,
y = minvar_p$ret_p - 5e-03, size = 4, colour = "black") +
geom_hline(yintercept = minvar_p$ret_p, col = "black", size = 1, linetype =
"dashed")
```



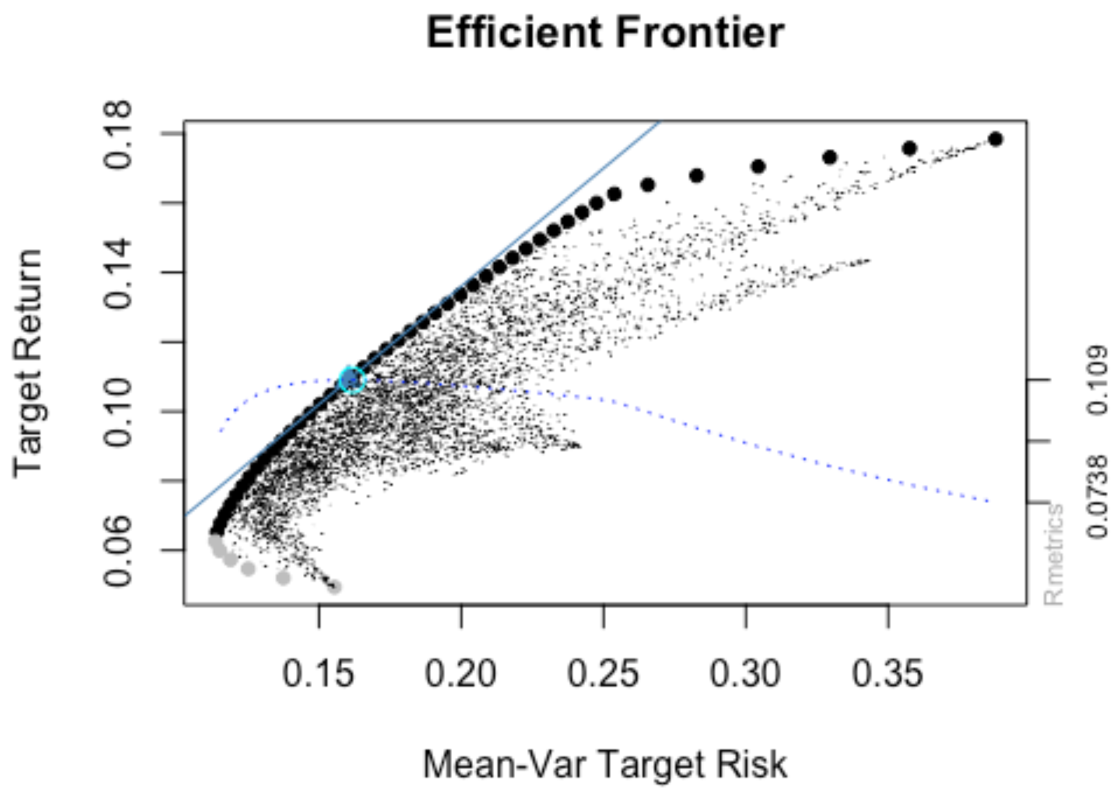
TailoredFrontierPlot

```
ts_p = as.timeSeries(ret.mat)
ef = portfolioFrontier(ts_p, constraints = "LongOnly")
tailoredFrontierPlot(object = ef)
```



c. Now assume risk free rate of 3% Plot with fportfolio package

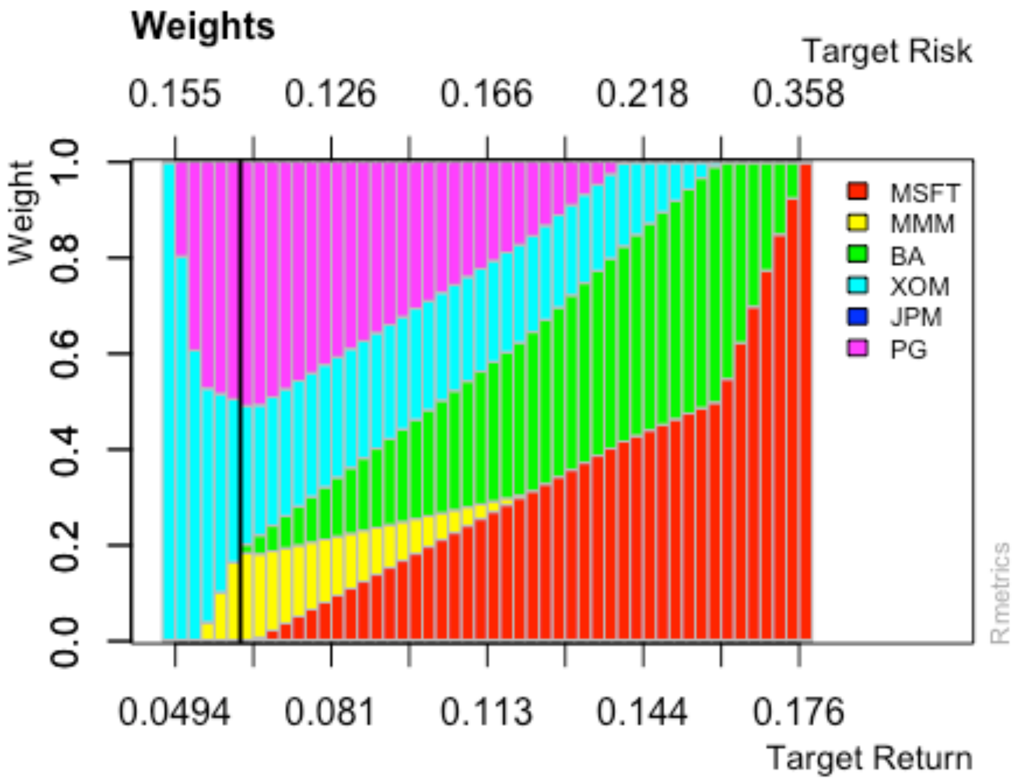
```
pspec = portfolioSpec(portfolio=list(riskFreeRate = 0.03,nFrontierPoints =
1000)) #set rf to 3%
plot(ef, c(1,3,7,8))
```



d.

Variant Weights dependant on Target Risk and Return

```
weightsPlot(ef,labels = TRUE, col=rainbow(ncol(ret.mat)))
```




```
#optimising based on rf and target expected return
optimalweights <- portfolio.optim(x=as.matrix(ret.mat),rf=0.03, pm=0.15,
riskless=TRUE, shorts=FALSE)
W <- round(optimalweights$pw, digits = 5)
dt<-data.table(colnames(ret.mat),W,100*round(W, digits = 4))
colnames(dt)<-c("Ticker","Weight","Weight (%)")
print(dt)
```

```
##    Ticker  Weight Weight (%)
## 1:   MSFT 0.45777      45.78
## 2:    MMM 0.00000       0.00
## 3:     BA 0.45111      45.11
## 4:    XOM 0.04412       4.41
## 5:    JPM 0.00000       0.00
## 6:     PG 0.00000       0.00
```