

# assignment05

Jeff Nguyen

18/09/2020

**Regression Analysis on impact of COVID-19 on a 5-stock portfolio.**

**Ngoc Son (Jeff) Nguyen**

**University of Southern California**

**Marshall School of Business**

**FBE 506 Quantitative Method in Finance**

**Fall 2020**

**Directed by Professor Mohammad Safarzadeh**

## Table of Contents

### Abstract

### Introduction

We selected the following 5 securities to base our analysis of impact of COVID-19 on a CAPM model of 5 stocks upon.

Ticker	Security	Sector	Industry	Founded	Full Time Employees
MSFT	Microsoft Corporation	Technology	Software-Infrastructure	1975	163,000
GWPH	GW Pharmaceuticals PLC	Healthcare	Drug Manufacturers-General	1998	901
DIS	The Walt Disney Company	Communication Services	Entertainment	1923	223,000
CAT	Caterpillar INC	Industrials	Farm & Heavy Construction Machinery	1925	102,300
AMZN	Twitter INC	Consumer Cyclical	Internet Retail	1994	1,125,300

All information and data related to the securities are obtained from Yahoo Finance: MSFT, GWPH, DIS, CAT, and AMZN.

The objective of the study of the study is using the Modern Portfolio Theory to model a portfolio of five securities from different industries using adjusted closing price data from January 02, 2014 to December 31,

2018 to achieve the following:

- 1) Estimated the CAPM for the five securities portfolio.
- 2) Understand the impact of COVID-19 on the alpha and market risk of the CAPM model.
- 3) Compare the MPT portfolio to a similarly diversified portfolio (State Street's SPDR S&P 500 Trust ETF).
- 4) Use the CAPM model to forecast the returns on the portfolio.

## Data Analysis

### 1) Estimated the CAPM for the five securities portfolio

1. Select at least five stocks from different industries (for the list of the firms in different industries see, [https://biz.yahoo.com/p/sum\\_conameu.html](https://biz.yahoo.com/p/sum_conameu.html)).

```
library(quantmod)

## Loading required package: xts

## Loading required package: zoo

##
## Attaching package: 'zoo'

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

## Loading required package: TTR

## Registered S3 method overwritten by 'quantmod':
##   method             from
##   as.zoo.data.frame zoo

## Version 0.4-0 included new data defaults. See ?getSymbols.

# Set start date and end date of data
start_date <- "2014-01-01"
end_date <- "2018-12-31"

# Get data for JPM, FB and the 10 year T-bill (TNX)
getSymbols("MSFT", src = "yahoo", from = start_date, to = end_date)

## 'getSymbols' currently uses auto.assign=TRUE by default, but will
## use auto.assign=FALSE in 0.5-0. You will still be able to use
## 'loadSymbols' to automatically load data. getOption("getSymbols.env")
## and getOption("getSymbols.auto.assign") will still be checked for
## alternate defaults.
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.warning4.0"=FALSE). See ?getSymbols for details.
```

```

## [1] "MSFT"

getSymbols("GWPB", src = "yahoo", , from = start_date, to = end_date)

## [1] "GWPB"

getSymbols("DIS", src = "yahoo", , from = start_date, to = end_date)

## [1] "DIS"

getSymbols("CAT", src = "yahoo", , from = start_date, to = end_date)

## [1] "CAT"

getSymbols("AMZN", src = "yahoo", , from = start_date, to = end_date)

## [1] "AMZN"

getSymbols("^GSPC", src = "yahoo", , from = start_date, to = end_date) # S&P 500

## [1] "^GSPC"

getSymbols("^TNX", src = "yahoo", from = start_date, to = end_date) # TNX (10-year T-bill)

## Warning: ^TNX contains missing values. Some functions will not work if objects
## contain missing values in the middle of the series. Consider using na.omit(),
## na.approx(), na.fill(), etc to remove or replace them.

## [1] "^TNX"

# Adjusted Prices
adjMSFT <- MSFT$MSFT.Adjusted
adjGWPB <- GWPB$GWPB.Adjusted
adjDIS <- DIS$DIS.Adjusted
adjCAT <- CAT$CAT.Adjusted
adjAMZN <- AMZN$AMZN.Adjusted

# Get adjusted returns data
rMSFT <- diff(log(to.monthly(MSFT)$MSFT.Adjusted))
rGWPB <- diff(log(to.monthly(GWPB)$GWPB.Adjusted))
rDIS <- diff(log(to.monthly(DIS)$DIS.Adjusted))
rCAT <- diff(log(to.monthly(CAT)$CAT.Adjusted))
rAMZN <- diff(log(to.monthly(AMZN)$AMZN.Adjusted))
rGSPC <- diff(log(to.monthly(GSPC)$GSPC.Adjusted))
rTNX <- (to.monthly(TNX)$TNX.Adjusted) / 1200 # Using monthly rate

## Warning in to.period(x, "months", indexAt = indexAt, name = name, ...): missing
## values removed from data

```

```

# Calculate statistics
MSFT_return_mean <- mean(rMSFT, na.rm = TRUE)
GWPH_return_mean <- mean(rGWPH, na.rm = TRUE)
DIS_return_mean <- mean(rDIS, na.rm = TRUE)
CAT_return_mean <- mean(rCAT, na.rm = TRUE)
AMZN_return_mean <- mean(rAMZN, na.rm = TRUE)
GSPC_return_mean <- mean(rGSPC, na.rm = TRUE)
TNX_return_mean <- mean(rTNX, na.rm = TRUE)

MSFT_return_var <- var(rMSFT, na.rm = TRUE)
GWPH_return_var <- var(rGWPH, na.rm = TRUE)
DIS_return_var <- var(rDIS, na.rm = TRUE)
CAT_return_var <- var(rCAT, na.rm = TRUE)
AMZN_return_var <- var(rAMZN, na.rm = TRUE)
GSPC_return_var <- var(rGSPC, na.rm = TRUE)

# Excess Returns
reMSFT <- rMSFT - rTNX
reGWPH <- rGWPH - rTNX
reDIS <- rDIS - rTNX
reCAT <- rCAT - rTNX
reAMZN <- rAMZN - rTNX

# Information Tables:
pricTab1 <- data.frame(MSFT, GWPH, DIS, CAT, AMZN)

# Creates data frame of asset prices
retTab1 <- data.frame(rMSFT, rGWPH, rDIS, rCAT, rAMZN)

# Creates data frame of returns
EretTab1 <- data.frame(reMSFT, reGWPH, reDIS, reCAT, reAMZN)

# Excess return data frame
retTab1 <- retTab1[-1,] # remove missing data due to lagging
EretTab1 <- EretTab1[-1,] # remove missing data due to lagging
priceMat1 <- matrix(c(MSFT, GWPH, DIS, CAT, AMZN), nrow=length(MSFT), ncol=5, byrow=TRUE) # creates a

# Variance/Covariance Matrix
asset.names <- c("MSFT", "GWPH", "DIS", "CAT", "AMZN")

# Create a list of row and col names for the var/cov matrix
VCV <- matrix(c(cov(retTab1)), nrow=5, ncol = 5, byrow=TRUE) # create a var/cov matrix by finding cov o
dimnames(VCV) <- list(asset.names, asset.names) # assigns asset.names to the VCV matrix

#Calculate Returns
rm <- matrix(colMeans(retTab1, na.rm=TRUE)) # creates an average return matrix, omitting missing value
erm <- matrix(colMeans(EretTab1, na.rm=TRUE)) # creates an average excess return matrix, omitting miss
tnxy = mean((rTNX)[-1,]) # calculates the average bond yield excluding Jan (risk free rate)

#Create Return Table
retmat <- matrix(c(rm, erm), ncol=2)
dimnames(retmat) = list(asset.names, c("Return ", "Excess Return"))

```

First we want to look at the data statistics

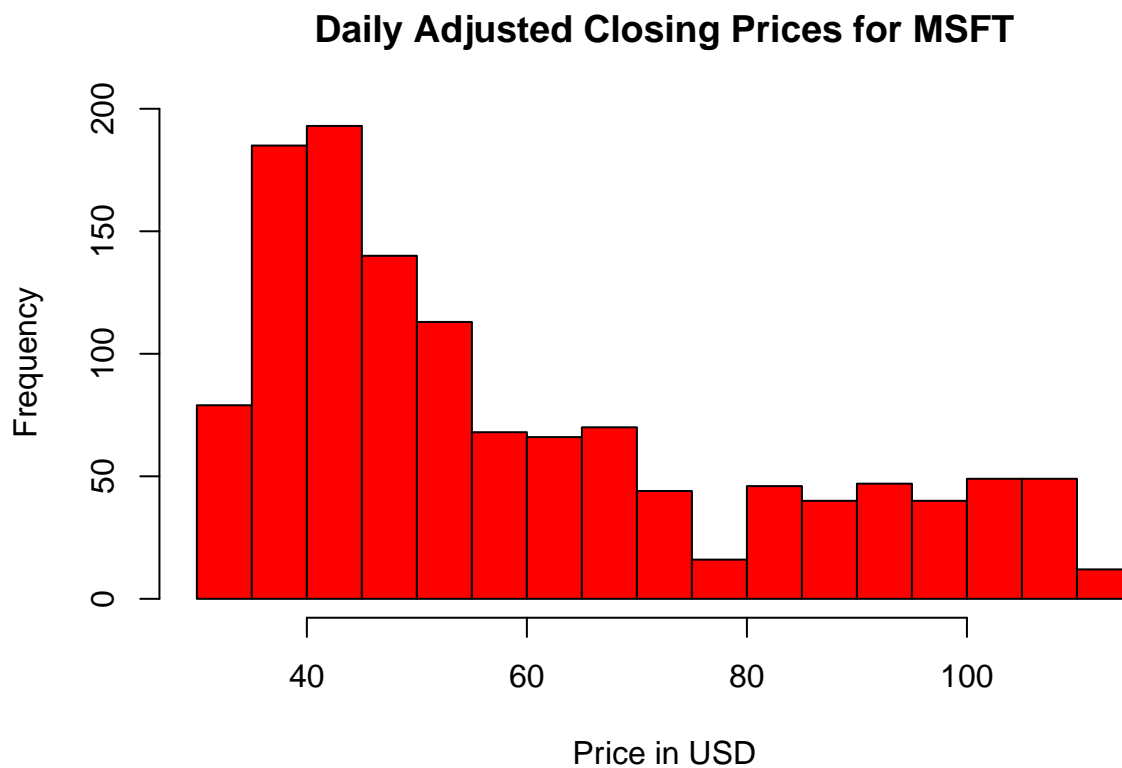
Instruments	Mean Returns	Variance of Returns	Beta (5Y Monthly)
MSFT	0.0185744	0.0035553	.87
GWPH	0.0090021	0.0293563	1.96
DIS	0.0078206	0.0026173	1.08
CAT	0.0076711	0.0057854	.98
AMZN	0.024	0.006961	1.3

Parameters of indices:

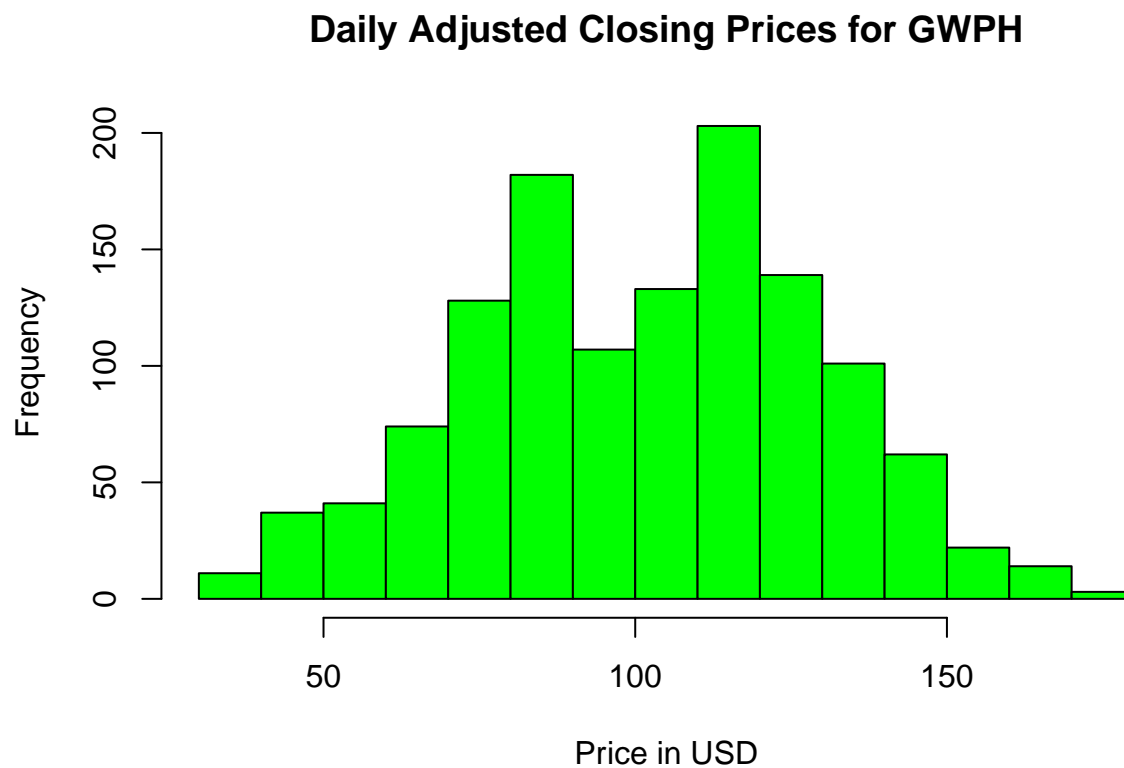
Instruments	Mean Returns	Variance of Returns	Beta
S&P 500	0.0056356	0.0010169	N/A
10-Year T-bill	0.0019378	0	N/A

We look at distribution of adjusted closing prices for each security:

```
hist(adjMSFT,
     main='Daily Adjusted Closing Prices for MSFT',
     xlab='Price in USD',
     col='red',
     )
```

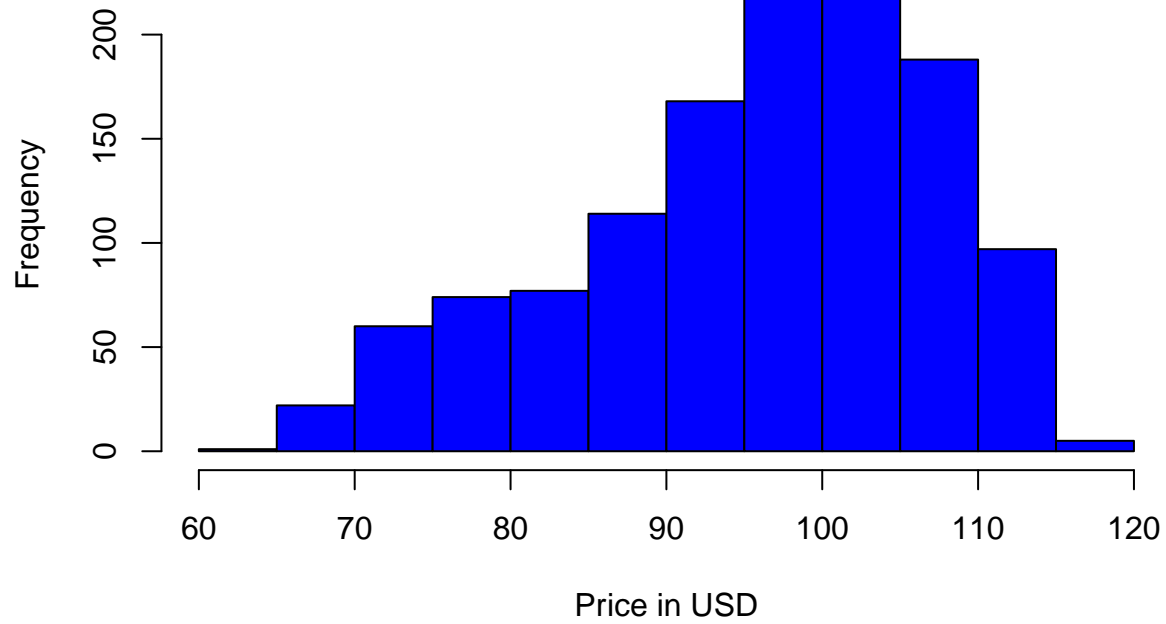


```
hist(adjGWPB,
     main='Daily Adjusted Closing Prices for GWPB',
     xlab='Price in USD',
     col='green',
)
```

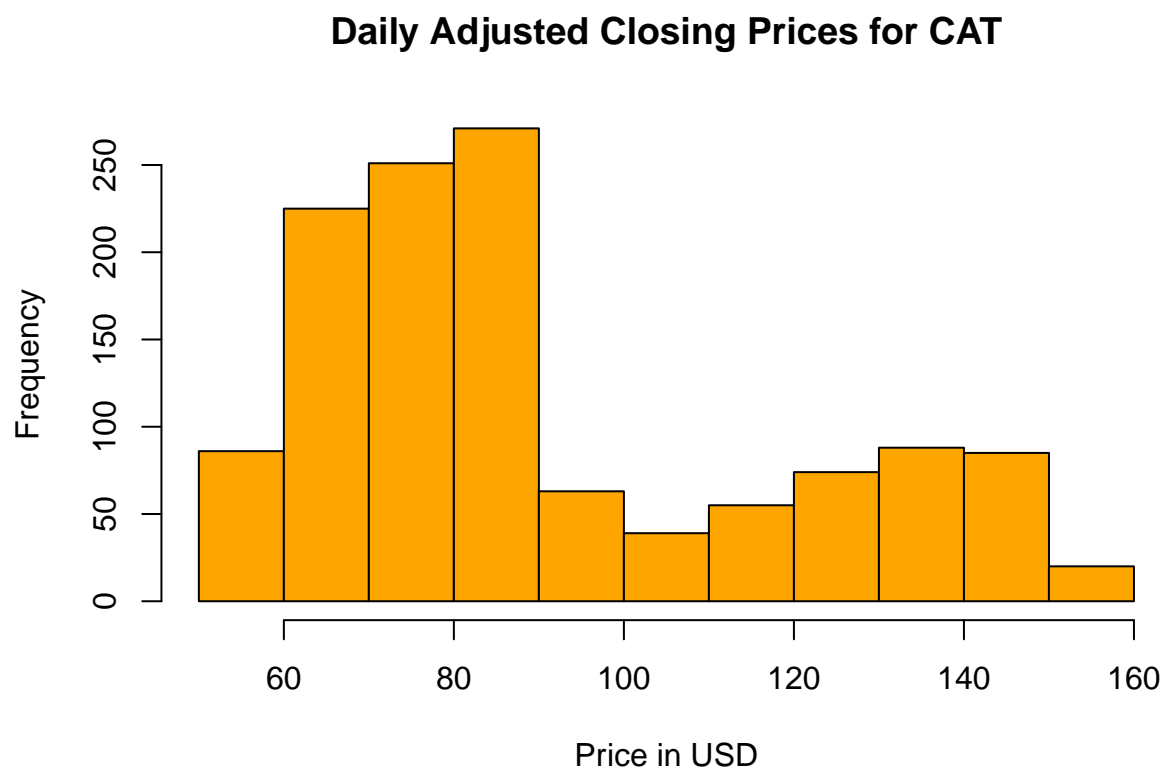


```
hist(adjDIS,
     main='Daily Adjusted Closing Prices for DIS',
     xlab='Price in USD',
     col='blue',
)
```

## Daily Adjusted Closing Prices for DIS



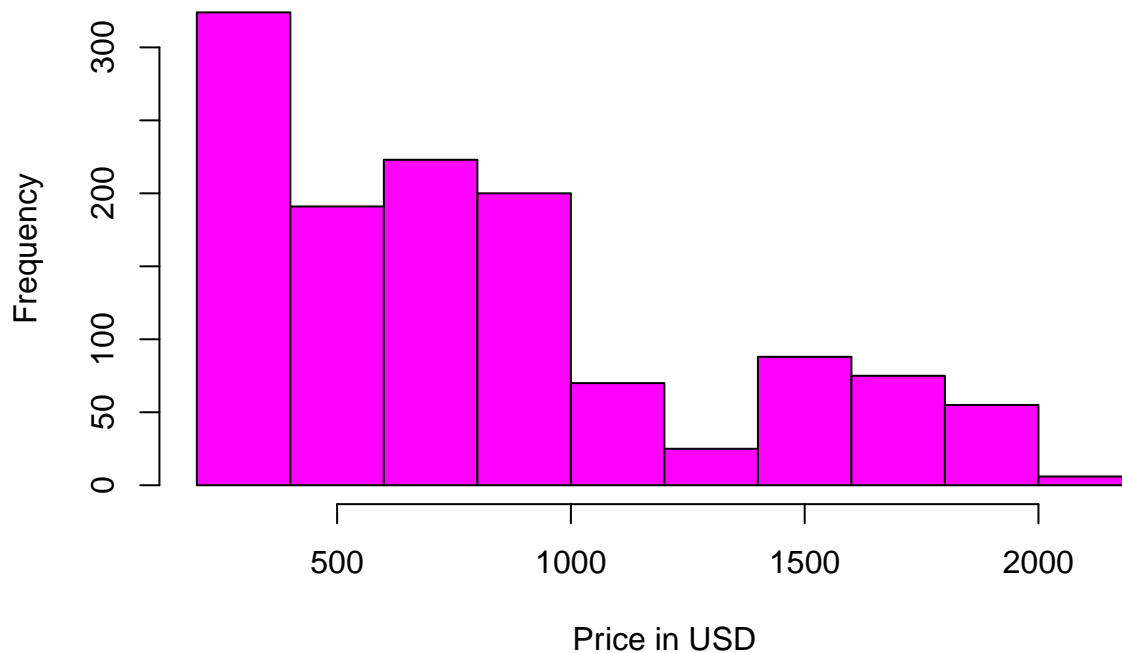
```
hist(adjCAT,  
      main='Daily Adjusted Closing Prices for CAT',  
      xlab='Price in USD',  
      col='orange',  
      )
```



```
hist(adjAMZN,  
      main='Daily Adjusted Closing Prices for AMZN',  
      xlab='Price in USD',  
      col='magenta',  
      )
```



## Daily Adjusted Closing Prices for AMZN



## CAPM Portfolio Construction

Question covered: 2,3,4,5

Methodology:

- 2a) Find the optimum weights using MPT.
- 2b) Allocate \$100.00 among the selected stocks using adjusted closing prices at 2018M12. 2019M1 will have a value of 100 as an index.
- 2c) Using the adjusted closing prices from 2018M12 to 2020M8 calculate the holding values of the portfolio (assume fixed holdings with no re-balancing taking place over time).
- 3) Find the tangency point of the Capital Allocation Line (CAL) and the efficient frontier.
- 4) Calculate the CAL equation and graph CAL and the efficient frontier.
- 5) Estimate CAPM for your portfolio and graph the estimated  $\beta$  of the CAPM and the average return of your portfolio as a point relative to SML.

### 2a) Find the optimum weights using MPT

Since the investor's objective is to minimize risk subjected to a minimum return of the risk free asset—US Treasury Bill, in this case—we solve the constrained optimization problem.

Let  $x_i$  denotes the weight of the investment in asset  $i$  ( $i = 1, 2, 3, 4, 5$ ), and assume all money is invested in  $i$ , meaning  $\sum x_i = x_1 + x_2 + x_3 + x_4 + x_5 = 1$ .

The returns of the portfolio is:

$$R_{p,x} = x_1 * r_1 + x_2 * r_2 + x_3 * r_3 + x_4 * r_4 + x_5 * r_5$$

The expected returns on the portfolio is:

$$\begin{aligned}\mu_{p,x} &= E[R_{p,x}] \\ &= x_1 * \mu_1 + x_2 * \mu_2 + x_3 * \mu_3 + x_4 * \mu_4 + x_5 * \mu_5\end{aligned}\tag{1}$$

The variance of the portfolio returns is:

$$\sigma_{p,x}^2 = \text{var}(R_{p,x})$$

Formulating the Markowitz portfolio problem:

The investor's objective is:

$$\begin{aligned}\max \quad & \mu_p = w' * \mu \quad \text{s.t.} \\ & \sigma_p^2 = w' * (\sum) * w \quad \text{and} \quad w' * I = 1\end{aligned}$$

where:

$w$  = matrix of asset weights in the portfolio

$w'$  = transpose matrix of asset weights in the portfolio

$\mu$  = matrix of mean returns of asset in the portfolio

$\sum$  = Variance-covariance matrix of asset returns in the the portfolio

$$w' * I = \sum_{i=1}^n w_n \quad \text{or the sum weights of the asset in the portfolio, I is notation for identity matrix}\tag{2}$$

Let  $\mu_{p,0}$  denotes a target expected return level. Formulate the problem:

$$\begin{aligned}\min \quad & \sigma_{p,w}^2 = w' * (\sum) * w \quad \text{s.t.} \\ & \mu_p = w' * \mu = \mu_{p,0}, \quad \text{and} \quad w' * I = 1\end{aligned}\tag{3}$$

To solve this, form the Lagrangian function:

$$L(w, \lambda_1, \lambda_2) = w' * \sum * w + \lambda_1 * (w' * \mu - \mu_{p,0}) + \lambda_2 * (w' * I - 1)\tag{4}$$

Because there are two constraints ( $w' * \mu = \mu_{p,0}$  and  $w' * I = 1$ ) there are two Langrange multipliers  $\lambda_1$  and  $\lambda_2$ . The first order condition for a minimum are the linear equations:

$$\begin{aligned}\frac{\partial L(w, \lambda_1, \lambda_2)}{\partial w} &= \frac{\partial(\sum * w^2)}{\partial w} + \frac{\partial(\lambda_1 * (w' * \mu - \mu_{p,0}))}{\partial w} + \frac{\lambda_2 * (w' * I - 1)}{\partial w} = 0 \\ \frac{\partial L(w, \lambda_1, \lambda_2)}{\partial \lambda_1} &= 0 \\ \frac{\partial L(w, \lambda_1, \lambda_2)}{\partial \lambda_2} &= 0\end{aligned}\tag{5}$$

Simplify, we have:

$$\begin{aligned}
\frac{\partial L(w, \lambda_1, \lambda_2)}{\partial w} &= 2 * \sum * w + \lambda_1 * \mu + \lambda_2 * I = 0 \\
\frac{\partial L(w, \lambda_1, \lambda_2)}{\partial \lambda_1} &= w' * \mu - \mu_{p,0} = 0 \\
\frac{\partial L(w, \lambda_1, \lambda_2)}{\partial \lambda_2} &= w' * I - 1 = 0
\end{aligned} \tag{6}$$

Rewrite in matrix form:

$$\begin{pmatrix} 2 * \sum & \mu & I \\ \mu' & 0 & 0 \\ I' & 0 & 0 \end{pmatrix} * \begin{pmatrix} w \\ \lambda_1 \\ \lambda_2 \end{pmatrix} = \begin{pmatrix} 0 \\ \mu_{p,0} \\ I \end{pmatrix} \tag{7}$$

or

$$\begin{aligned}
A * z_w &= b_0 \\
\text{where} \\
A &= \begin{pmatrix} 2 * \sum & \mu & I \\ \mu' & 0 & 0 \\ I' & 0 & 0 \end{pmatrix} \\
z_w &= \begin{pmatrix} w \\ \lambda_1 \\ \lambda_2 \end{pmatrix} \\
b_0 &= \begin{pmatrix} 0 \\ \mu_{p,0} \\ I \end{pmatrix}
\end{aligned} \tag{8}$$

The solution for  $z_w$  is:

$$z_w = A^{-1} * b_0 \tag{9}$$

The variance-covariance matrix is as follow:

VCV

```
##           MSFT           GWPB           DIS           CAT           AMZN
## MSFT 0.003555316 0.001599836 0.001128235 0.002042991 0.002538589
## GWPB 0.001599836 0.029356292 0.002528779 0.005980634 0.005396791
## DIS  0.001128235 0.002528779 0.002617304 0.001243413 0.001405934
## CAT  0.002042991 0.005980634 0.001243413 0.005785368 0.002040476
## AMZN 0.002538589 0.005396791 0.001405934 0.002040476 0.006961024
```

The monthly risk-free rate is: 0.001933

```
# Optimum Portfolio
ZOPT <- solve(VCV,erm) # multiply inverse of VCV to excess return to find z
WOPT <- ZOPT/sum(ZOPT) # calculates weights
dimnames(WOPT) <- list(asset.names, "Weights") #label the weight matrix

# Calculate stats
```

```

ROPT <- t(WOPT)%*%rm # calculate optimal portfolio's return
VOPT <- t(WOPT)%*%VCV%*%WOPT # calculate optimal portfolio's variance
SDOPT <- VOPT^0.5 # calculate optimal portfolio's std dev
SRatio <- (ROPT-tnxy)/(SDOPT) # calculate optimal portfolio's Sharpe ratio

# Create Optimal Stats Table
PTBL <- matrix(c(ROPT, VOPT, SDOPT, SRatio), nrow = 4) # create a matrix of return, variance, std dev,
optstat.names <- c("Return", "Variance", "Std Dev", "Sharpe") # labels for PTBL matrix

dimnames(PTBL) <- list(optstat.names, "Opt. Portfolio") # label the optimal portfolio matrix values

```

The optimal portfolio weights are as follow:

WOPT

```

##           Weights
## MSFT  0.75172435
## GWPB -0.03902504
## DIS   0.01785490
## CAT  -0.18682397
## AMZN  0.45626977

```

The statistics of the optimal portfolio is:

PTBL

```

##           Opt. Portfolio
## Return      0.02326846
## Variance    0.00436794
## Std Dev     0.06609039
## Sharpe      0.32282226

```

2b) Allocate \$100.00 among the selected stocks using adjusted closing prices at 2018M12. 2019M1 will have a value of 100 as an index.

```

# Set start date and end date of data
start_date1 <- "2018-12-01"
end_date1 <- "2020-08-31"

# Get data for JPM, FB and the 10 year T-bill (TNX)
getSymbols("MSFT", src = "yahoo", from = start_date1, to = end_date1)

## [1] "MSFT"

getSymbols("GWPB", src = "yahoo", , from = start_date1, to = end_date1)

## [1] "GWPB"

```

```

getSymbols("DIS", src = "yahoo", , from = start_date1, to = end_date1)

## [1] "DIS"

getSymbols("CAT", src = "yahoo", , from = start_date1, to = end_date1)

## [1] "CAT"

getSymbols("AMZN", src = "yahoo", , from = start_date1, to = end_date1)

## [1] "AMZN"

getSymbols("^GSPC", src = "yahoo", , from = start_date1, to = end_date1) # S&P 500

## [1] "^GSPC"

getSymbols("^TNX", src = "yahoo", from=start_date1, to=end_date1) # TNX (10-year T-bill)

## Warning: ^TNX contains missing values. Some functions will not work if objects
## contain missing values in the middle of the series. Consider using na.omit(),
## na.approx(), na.fill(), etc to remove or replace them.

## [1] "^TNX"

rGSPC1 <- diff(log(to.monthly(GSPC)$GSPC.Adjusted))
rTNX1 <- to.monthly(TNX)$TNX.Adjusted /1200 # Using monthly rate

## Warning in to.period(x, "months", indexAt = indexAt, name = name, ...): missing
## values removed from data

rTNX1 <- rTNX1[-1,] # remove missing data due to lagging
mean_rTNX1 <- mean(rTNX1, na.rm=TRUE)

# Adjusted Prices
adjMSFT1 <- MSFT$MSFT.Adjusted
adjGWPH1 <- GWPH$GWPH.Adjusted
adjDIS1 <- DIS$DIS.Adjusted
adjCAT1 <- CAT$CAT.Adjusted
adjAMZN1 <- AMZN$AMZN.Adjusted

investedAmount <- 100

sharesMSFT <- as.numeric(investedAmount * WOPT[1] / adjMSFT1[1])
sharesGWPH <- as.numeric(investedAmount * WOPT[2] / adjGWPH1[1])
sharesDIS <- as.numeric(investedAmount * WOPT[3] / adjDIS1[1])
sharesCAT <- as.numeric(investedAmount * WOPT[4] / adjCAT1[1])
sharesAMZN <- as.numeric(investedAmount * WOPT[5] / adjAMZN1[1])

```

```

holdings <- data.frame("#MSFT"=sharesMSFT*adjMSFT1,
                      #"GWPH"=sharesGWPH*adjGWPH1,
                      #"DIS"=sharesDIS*adjDIS1,
                      #"CAT"=sharesCAT*adjCAT1,
                      #"AMZN"=sharesAMZN*adjAMZN1,
                      "Holding Value"=sharesMSFT*adjMSFT1 +
                                sharesGWPH*adjGWPH1 +
                                sharesDIS*adjDIS1 +
                                sharesCAT*adjCAT1 +
                                sharesAMZN*adjAMZN1)
names(holdings)[1] <- "Port. Holdings Val" # rename column

```

Based on the optimal weighting, to allocate \$100 to the portfolio, we would be purchase the following amount of each security:

Ticker	Weights	Stock to purchase
MSFT	0.7517244	0.6859734
GWPH	-0.039025	-0.0309109
DIS	0.0178549	0.0157364
CAT	-0.186824	-0.1424304
AMZN	0.4562698	0.0257436

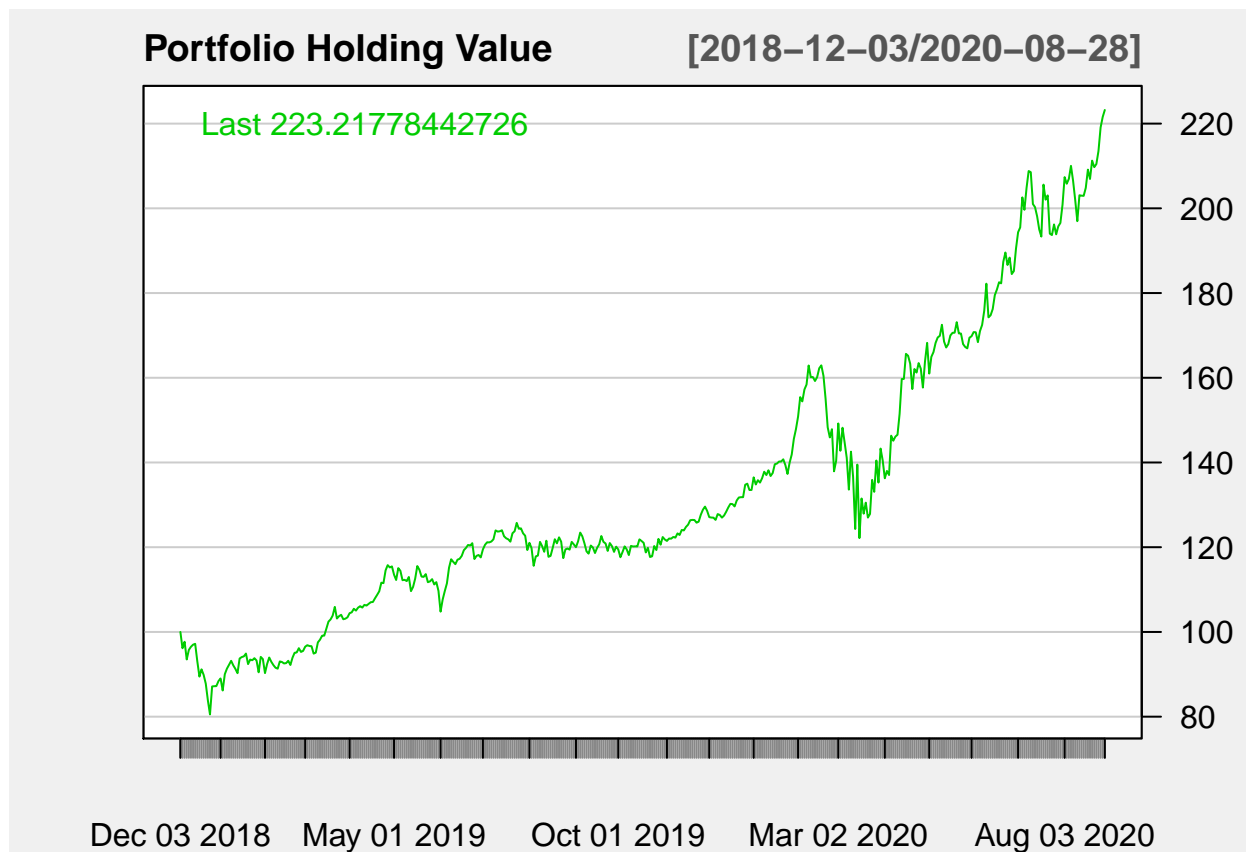
2c) Using the adjusted closing prices from 2018M12 to 2020M8 calculate the holding values of the portfolio (assume fixed holdings with no re-balancing taking place over time).

We can then observe the fluctuations in the holding value of the portfolio from the period starting December 01 2018 to August 31, 2020 as follow.

```

chartSeries(holdings, name="Portfolio Holding Value", type="line", theme=chartTheme("white"))

```



By inspection we can see the portfolio experience a sharp sell off of almost 20% in December 2018, coincide with the broad U.S. market selloff due to a combination of the FED hiking the federal funds rate by 25 basis points to a targeted range of 2.25% to 2.5% (JeffCoxCNBCcom) and corporations followed suit by cutting profit forecasts and try temper expectations for earnings growth in 2019 after a big 2018 (Moyer).

The second visibly sharp sell off of the portfolio holding value also coincides with the broad market sell off in the mid March 2020 with investors raising cash in a risk-on environment when COVID-19 lockdowns started going into effects in the U.S.

### 3) Find the tangency point of the Capital Allocation Line (CAL) and the efficient frontier.

The tangency point of the Capital Allocation Line is the point where the weights of the portfolio is optimal, represented by the point  $(\sigma_p, r_p)$  which is (0.0660904, 0.0232685).

### 4) Calculate the CAL equation and graph CAL and the efficient frontier.

The efficient frontier is the portfolio possibility curve represented by the equation:  $CAL = 0.001933 + 0.3228223 * \sigma_p$

```
# Efficient Frontier and CAL
j <- 0 # set value for iterative loop variable t
return_p <- rep(0, 50000)
sd_p <- rep(0, 50000)

# create a matrix of 0 to fill later with sd of different weights
```

```

vect_0 <- rep(0, 50000)

# create a matrix of 0
fractions <- matrix(vect_0, 10000, 5)

# create a matrix of 0 to fill with weights
# iterate through weights for asset 1-5 from -20% to 100% by 10%
for (a in seq(-.2, 1, 0.1))
{
  for (b in seq(-.2, 1, 0.1))
  {
    for (c in seq(-.2, 1, 0.1))
    {
      for (d in seq(-.2, 1, 0.1))
      {
        for (e in seq(-.2, 1, 0.1))
        {
          #test that the weights are equal to 1
          if (a+b+c+d+e==1)
          {
            # increment j by 1 if a+b+c+d+e is equal to 1 (valid weights)
            j=j+1
            # load a,b,c,d,e values into row j of the matrix
            fractions[j,] <- c(a,b,c,d,e)
            # calculate the std dev of the portfolio at a given weight of assets
            sd_p[j] <- (t(fractions[j,])%*%VCV)%*%fractions[j,])^.5
            # calculate the return of the portfolio at a given weight of assets
            return_p[j] <- fractions[j,]%*%rm
          }
        }
      }
    }
  }
}

# assign filled vector spots in return_p to the R_p matrix to omit empty spots
Rport <- return_p[1:j]

# assign filled vector spots in sd_p to the sigma_p matrix to omit empty spots
StdDev_p <- sd_p[1:j]

# Create Capital Asset Line
# Create x-coordinates for CAL points
f <- seq(0, .24, .24)

# Calculate corresponding y-coordinates
CAL <- tnxy + SRatio * f

```

```

## Warning in SRatio * f: Recycling array of length 1 in array-vector arithmetic is deprecated.
## Use c() or as.vector() instead.

```

```

#Plot the portfolio possibilities curve:
plot(StdDev_p, Rport, col="green1", xlab="Portfolio Standard Deviation", ylab= "Portfolio Expected Return")

```



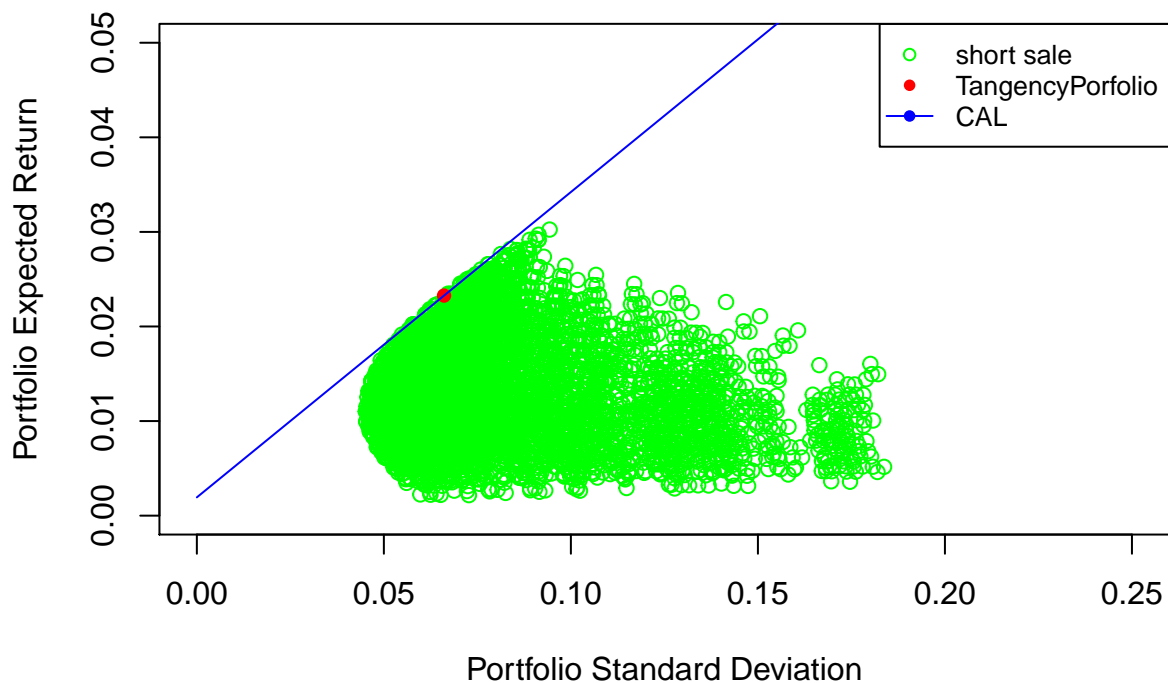
```

#Plot of tangency point in red
points(SDOPT, ROPT, col= "red", pch=16, bg="red")

#Plot of CAL in blue
points(f, CAL, col= "blue", type="l")

legend("topright",c("short sale", "TangencyPorfolio", "CAL"), cex=.8, col=c("green1", "red","blue"),
      lty =c(0,0,1),pch=c(1,16,16))

```



5) Estimate CAPM for your portfolio and graph the estimated  $\beta$  of the CAPM and the average return of your portfolio as a point relative to SML.

The expected risk premium of the portfolio based on the CAPM model is given as:

$$\begin{aligned}
E(R_a - R_f) &= \beta * (R_m - R_f) \\
&\text{or} \\
R_a &= R_f + \beta * (R_m - R_f) \\
R_a - R_f &= \alpha_{Jensen} + \beta * (R_m - R_f) \\
&\text{or} \\
Y &= \alpha_{Jensen} + \beta * X + \epsilon
\end{aligned} \tag{10}$$

with

$$\begin{aligned}
Y &= R_a - R_f \\
X &= R_m - R_f \\
\beta &= \text{Market risk or systematic risk} \\
\epsilon &= \text{stochastic error term}
\end{aligned}$$

Here, the risk premium of the S&P 500 is the independent variable and the expected risk premium of the portfolio is the dependent variable.

Hypothesis for regression:

$$\begin{aligned}
H_0 : \alpha &= 0 \\
H_a : \alpha &\neq 0 \\
&\text{and} \\
H_0 : \beta &= 0 \\
H_a : \beta &\neq 0
\end{aligned} \tag{11}$$

```

# Calculate and normalized the CAPM holdings
ra <- diff(log(to.monthly(holdings)[,1]))

Y <- na.omit(ra - rTNX1)
names(Y)[1] <- "Portfolio Risk Premium" # Rename column
Y_bar <- mean(Y)
Y_bar

## [1] 0.03515135

X <- na.omit(rGSPC1 - rTNX1)
mean_X <- mean(X)

names(X)[1] <- "Market Risk Premium" # Rename column
data1 <- data.frame(X, Y)

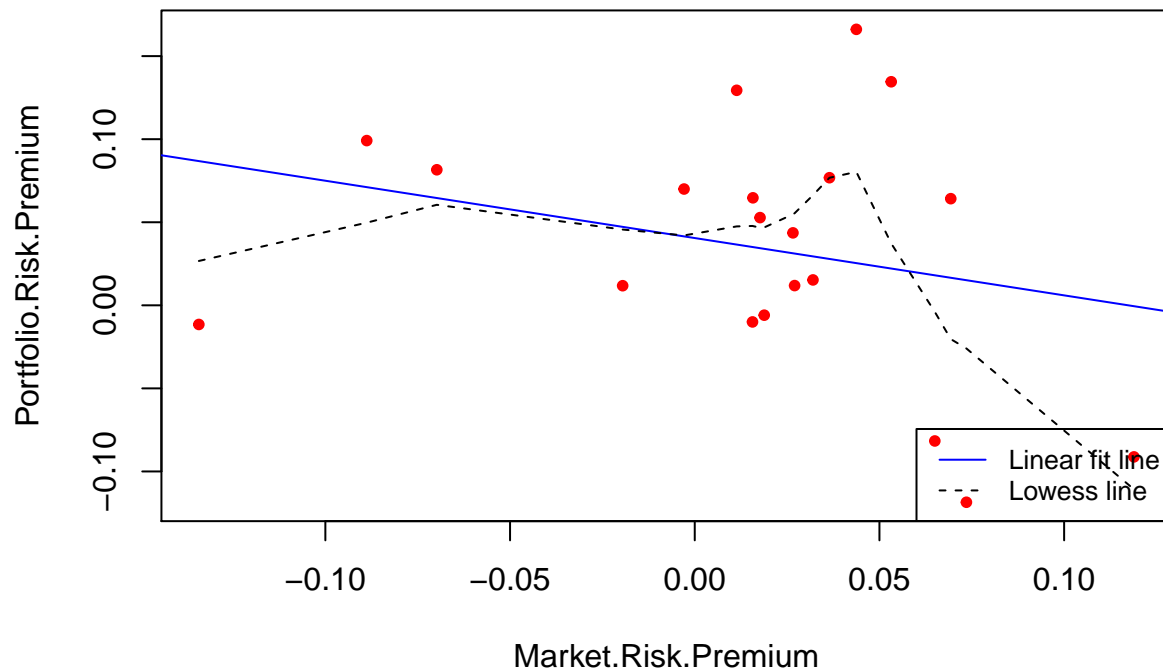
plot(data1, col='red', main="Relationship Between Market & Portfolio Risk Premium", pch=20, cex=1)

# Add fit lines
abline(lm(Y~X), col="blue") # Regression line Y ~ X
lines(lowess(X,Y), col="black", lty=2) # Lowess line (X,Y)

legend("bottomright",c("Linear fit line", "Lowess line"), cex=.8, col=c("blue", "black"), lty=1:2)

```

## Relationship Between Market & Portfolio Risk Premium



Through inspection, we observe a non linear relationship between the Market Portfolio Risk Premium (the independent  $X$  variable on the x-axis) and the CAPM Portfolio Risk Premium (the dependent  $Y$  variable on the y-axis).

Therefore, we fit an equation of a line  $Y = \alpha_{Jensen} + \beta * X$

```
fit1 <- lm(Y~X, data=data1)
summary(fit1)
```

```
##
## Call:
## lm(formula = Y ~ X, data = data1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.13365 -0.04123  0.01465  0.03414  0.14071
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.04050    0.01728   2.343  0.0308 *
## X           -0.34504    0.29305  -1.177  0.2544
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07458 on 18 degrees of freedom
## Multiple R-squared:  0.07151,    Adjusted R-squared:  0.01992
## F-statistic: 1.386 on 1 and 18 DF,  p-value: 0.2544
```

The estimated equation is  $Y = 0.04050 - 0.34504 * X$ , where the  $p_{value}$  for the intercept  $.0308 < .05$ . Therefore, we reject the null hypothesis at 95% confidence level that the intercept  $\alpha_{Jensen}$  statistically is not different from zero.

The coefficient  $\beta = 1.07468$  represents the increase in portfolio risk premium relative to increase in the market portfolio risk premium. The  $p_{value}$  for  $\beta$  is  $0.2544 > .05$ , implying that the coefficient  $\beta$  statistically is not significant at 95% or more, and we accept the null hypothesis.

Goodness of Fit:

Through inspection, we observe the  $R^2 = .07151$  value to not be close to 1 at all.  $R^2 = .07151$  implies that 7.15% of the variations in the portfolio risk premium is explained by the market risk premium.

Standard Error of Regression:

We can see that the Standard Error of Regression is  $S.E. = .07458$ .

From this, we can calculate the forecasting efficiency statistic to be:

$$\begin{aligned} \frac{S.E.}{\bar{Y}} &= \frac{.05618}{0.0351514} \\ &= 212.17\% > 10\% \end{aligned} \quad (12)$$

This statistic implies that this is not a good forecasting model.

The Security Market Line:

```
# Generate the SML equation
slope_SML <- (mean_X - mean_rTNX1) / (1-0)
#slope_SML
SML <- function(beta) mean_rTNX1 + slope_SML * beta
SML

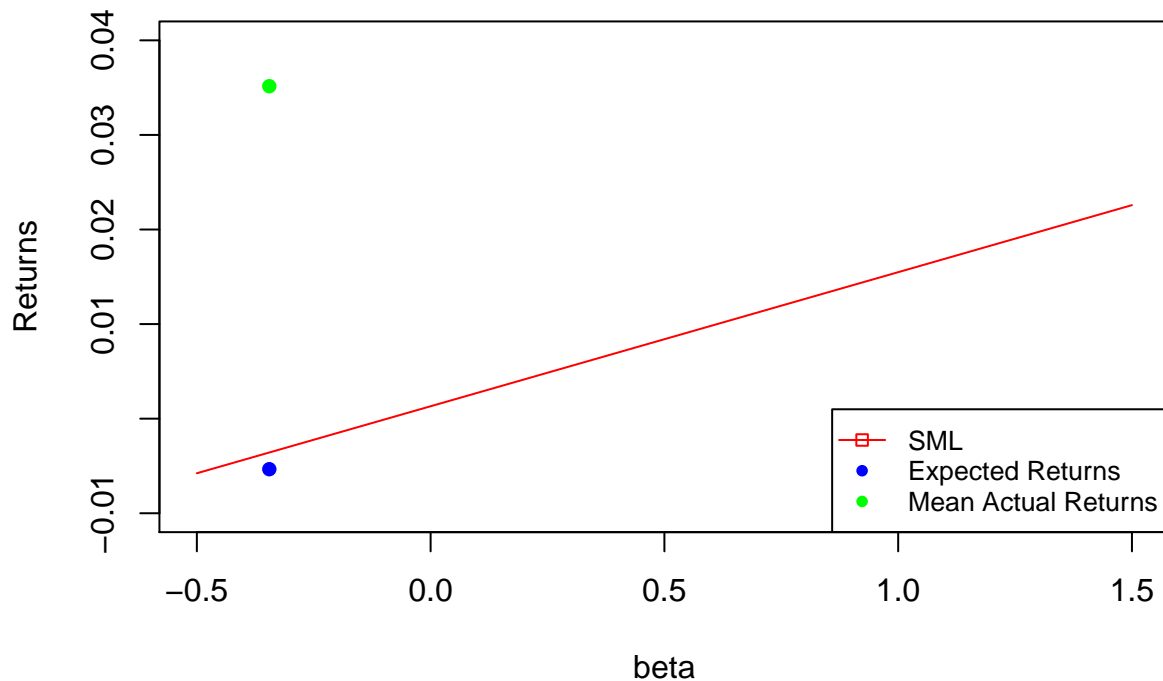
## function(beta) mean_rTNX1 + slope_SML * beta

# Plot the SML
beta <- seq(-.5, 1.5)
plot(beta, mean_rTNX1 + slope_SML * beta, col="red", type="l", main="CAPM portfolio beta relative to the market",
      # Plot the expected returns
      points(-.34504, -.34504*mean_X, col="blue", pch=16)

# Plot the average returns
points(-.34504, Y_bar, col="green", pch=16)

legend("bottomright",c("SML", "Expected Returns", "Mean Actual Returns"), cex=.8,
      col=c("red", "blue", "green"), lty=c(1,0,0), pch=c(0,16,16))
```

## CAPM portfolio beta relative to the Security Market Line



The Security Market Line pass through the point  $(0, \overline{R_f})$  and  $(1, \overline{X})$ , which are  $(0, 0.0013135)$  and  $(1, 0.0154876)$ .

Relative to its market risk of  $\beta = -0.34504$ , the expected return is  $-0.0053439$  and the average return is  $0.0351514$ . We can observe that at this estimated  $\beta$ , the expected return is below the security market line and the actual average return is above the security market line.

## 2) Understand the impact of COVID-19 on the alpha and market risk of the CAPM model.

Question covered: 6).

Methodology: Test whether the closing of the economy due to COVID-19 had any effect on Jensen alpha and the market risk of the CAPM model. (6)

## 3) Compare the CAPM portfolio to a diversified State Street's SPDR S&P 500 Trust ETF portfolio (7,8,9)

Methodology:

7) Calculate CV, Sharpe, Treynor, and Sortino ratios for your portfolio and compare them to a similarly diversified portfolio of Vanguard, Fidelity, or any other similar portfolio.

8) Calculate 2% and 3% VaRs as a percentage of the mean return of your portfolio when the risk horizon is one year, six months, and one month. Calculate the same VaRs for the selected portfolio in item 6. Compare the VaRs of your portfolio to the ones of the market portfolios.

9) Graph the scatter diagram of your portfolio and comment on trends, outliers, structural breaks and

any other special features. Graph the scatter diagram of your portfolio and S&P 500 on the same coordinate system and compare the trends.

#### 4) Forecast the returns of the portfolio CATsed on the CAPM model (10,11,12)

10) Using the CAPM equation of your portfolio do two periods ex-post forecasting of the returns to your portfolio and compare your forecast to the actual returns. Find the accuracy statistics of your forecast and report them.

11) Do two-periods ex-ante forecasting of returns to your portfolio assuming that the monthly risk premiums to S&P 500 will be 1.10% in 2020M9 and 1.25 in 2020M10.

12) Do forecasting of the returns to your portfolio for the period 2014M1-2020M8. Find the accuracy statistics of your forecast. Do a naïve forecasting of the returns to your portfolio for the period 2014M1-2020M8. Find the accuracy statistics of your forecast. Compare the forecasting accuracy criterion for the two forecasting methods. Which one results in a better forecasting outcome?

\*\*\*\*

## Citations

“Amazon.com, Inc. (AMZN) Stock Price, News, Quote & History.” Yahoo! Finance, Yahoo!, 14 Nov. 2020, [ca.finance.yahoo.com/quote/amzn/?p=amzn](https://ca.finance.yahoo.com/quote/amzn/?p=amzn).

“Caterpillar, Inc. (CAT) Stock Price, News, Quote & History.” Yahoo! Finance, Yahoo!, 13 Nov. 2020, [ca.finance.yahoo.com/quote/CAT/?p=CAT](https://ca.finance.yahoo.com/quote/CAT/?p=CAT).

JeffCoxCNBCcom. “Fed Hikes Rate, Lowers 2019 Projection to 2 Increases.” CNBC, CNBC, 19 Dec. 2018, [www.cnbc.com/2018/12/19/fed-hikes-rates-by-a-quarter-point-.html](https://www.cnbc.com/2018/12/19/fed-hikes-rates-by-a-quarter-point-.html).

“GW Pharmaceuticals Plc (GWPH) Stock Price, News, Quote & History.” Yahoo! Finance, Yahoo!, 13 Nov. 2020, [ca.finance.yahoo.com/quote/GWPH/?p=GWPH](https://ca.finance.yahoo.com/quote/GWPH/?p=GWPH).

Moyer, Liz. “We’re Finding out Now Why the Stock Market Tanked in December.” CNBC, CNBC, 9 Jan. 2019, [www.cnbc.com/2019/01/09/markets-december-tumble-may-have-hinted-at-profit-revisions-to-come.html](https://www.cnbc.com/2019/01/09/markets-december-tumble-may-have-hinted-at-profit-revisions-to-come.html).

“Microsoft Corporation (MSFT) Stock Price, News, Quote & History.” Yahoo! Finance, Yahoo!, 14 Nov. 2020, [ca.finance.yahoo.com/quote/msft/?p=msft](https://ca.finance.yahoo.com/quote/msft/?p=msft).

“Walt Disney Company (The) (DIS) Stock Price, News, Quote & History.” Yahoo! Finance, Yahoo!, 14 Nov. 2020, [ca.finance.yahoo.com/quote/dis/?p=dis](https://ca.finance.yahoo.com/quote/dis/?p=dis).