Stats C183 Project 1

Charles Liu (304804942)

4/3/2020

Load Necessary Packages:

```
library(readr)
```

A)

```
# Read your csv file:
a <- read.csv("C:/Users/cliuk/Documents/UCLA Works/UCLA Spring 2020/Stats C183/Homeworks/HW 1/stockData
```

B)

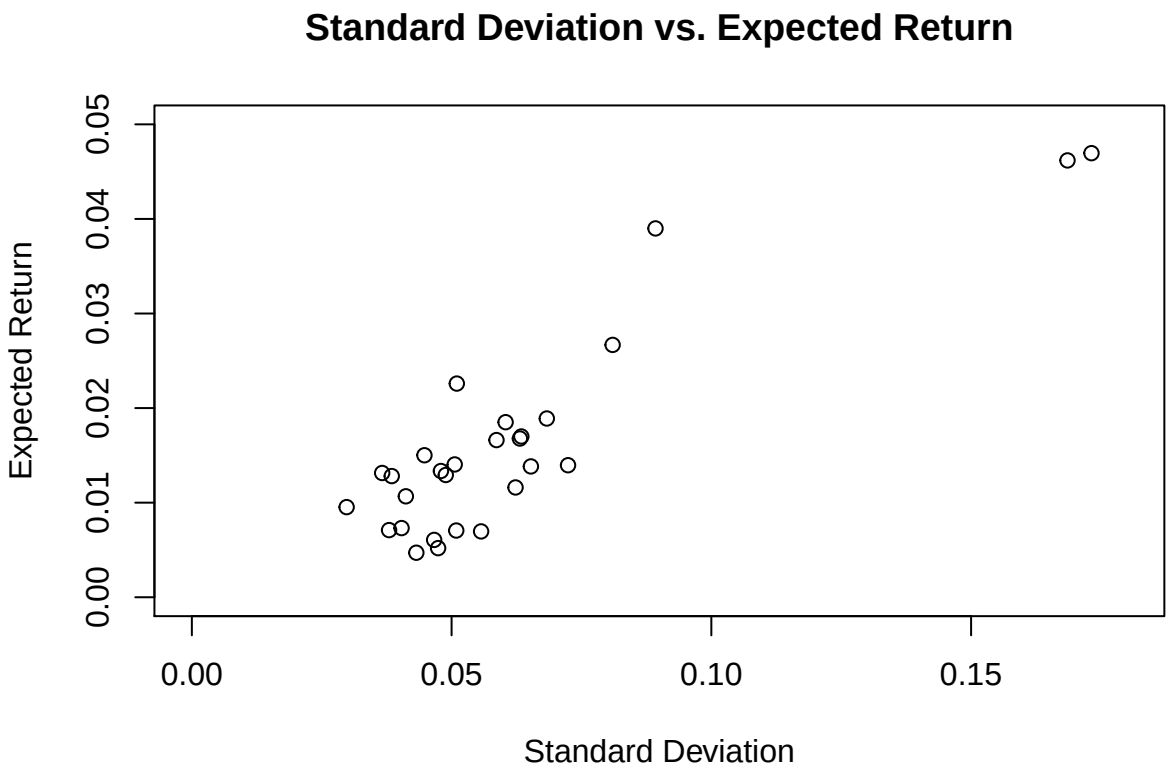
```
# Convert adjusted close prices into returns:
r <- (a[-1,3:ncol(a)]-a[-nrow(a),3:ncol(a)])/a[-nrow(a),3:ncol(a)]
```

C)

```
# Compute mean vector:
means_31 <- colMeans(r) # With ^GSPC

# Compute variance covariance matrix:
covmat_31 <- cov(r) # With ^GSPC

# Compute correlation matrix:
cormat_31 <- cor(r) # With ^GSPC

# Compute the vector of variances:
variances_31 <- diag(covmat_31)

# Compute the vector of standard deviations:
stdev_31 <- diag(covmat_31)^.5
```

D)

```
plot(stdev_31, means_31, xlim = c(0, 0.18), ylim = c(0, 0.05),
     main = "Standard Deviation vs. Expected Return",
     xlab = "Standard Deviation", ylab = "Expected Return")
```



E)

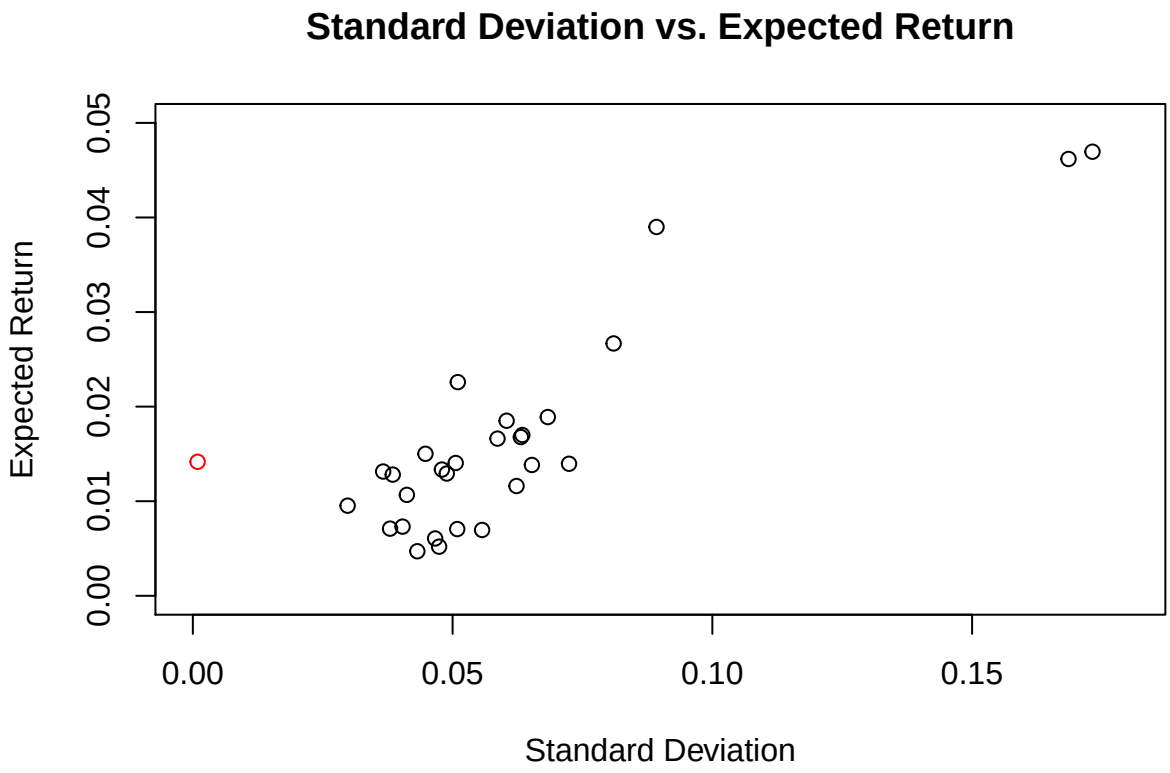
```r
# Compute mean vector:
means <- colMeans(r[-1]) # Without ^GSPC

# Compute variance covariance matrix:
covmat <- cov(r[-1]) # Without ^GSPC

# Compute correlation matrix:
cormat <- cor(r[-1]) # Without ^GSPC

# Compute the vector of variances:
variances <- diag(covmat)

# Compute the vector of standard deviations:
stdev <- diag(covmat)^.5

# Equal Allocation Formulas:
x <- rep(1/30, 30)
R_equal <- t(x) %*% means
sigma_equal <- t(x) %*% covmat %*% x

# Equal Allocation Numbers:
R_equal
```

```
##             [,1]
## [1,] 0.01416942
```

```
sigma_equal
```

```
##              [,1]
## [1,] 0.0009098234
```

```
# Plot Equal Allocation point to part C:
par(mfrow = c(1,1))
plot(stdev_31, means_31, xlim = c(0, 0.18), ylim = c(0, 0.05),
     main = "Standard Deviation vs. Expected Return",
     xlab = "Standard Deviation", ylab = "Expected Return")
points(sigma_equal, R_equal, col = "red")
```



F)

```
# Min Risk Formulas:
ones <- rep(1, 30)
R_min <- (t(ones) %*% solve(covmat) %*% means)/(t(ones) %*% solve(covmat) %*% ones)
sigma_min <- (1)/((t(ones) %*% solve(covmat) %*% ones)^1/2)

# Min Risk Numbers
R_min
```

```
##             [,1]
## [1,] 0.007260777
```

```
sigma_min
```

```
##                [,1]
## [1,] 0.0007039741
```

```
# Plot Minimum Risk point to part C:
par(mfrow = c(1,1))
plot(stdev_31, means_31, xlim = c(0, 0.18), ylim = c(0, 0.05),
     main = "Standard Deviation vs. Expected Return",
     xlab = "Standard Deviation", ylab = "Expected Return")
points(sigma_equal, R_equal, col = "red")
points(sigma_min, R_min, col = "blue")
```