Project 3

Yaman Yucel

2023-04-17

Project 3

Part a: Convert the prices into returns for all the 5 stocks. Important note: In this data set the most recent data are at the beginning. You will need to consider this when converting the prices into returns.

```
a <- read.table("http://www.stat.ucla.edu/~nchristo/statistics_c183_c283/statc183c283_5stocks.txt", hea

#Exxon-mobil -> P1
#General Motors -> P2
#Hewlett Packard -> P3
#McDonalds -> P4
#Boeing -> P5


#Reverse the list such that older values appear on top
b<- a[dim(a)[1]:1,]

#Convert adjusted close prices into returns:
r <- (b[-1,2:ncol(b)]-b[-nrow(b),2:ncol(b)])/b[-nrow(b),2:ncol(b)]
```

Part b: Compute the mean return for each stock and the variance-covariance matrix.

```
#Compute mean vector:
means <- colMeans(r)
print("Means:")
```

```
## [1] "Means:"
```

```
print(means)
```

```
##           P1           P2           P3           P4           P5
## 0.0027625075 0.0035831363 0.0066229478 0.0004543727 0.0045679106
```

```
#Compute variance covariance matrix
covmat <- cov(r)
print("Var-Covar:")
```

```
## [1] "Var-Covar:"
```

```
print(covmat)
```

```
##             P1          P2          P3          P4          P5
## P1 0.005803160 0.001389264 0.001666854 0.000789581 0.001351044
## P2 0.001389264 0.009458804 0.003944643 0.002281200 0.002578939
## P3 0.001666854 0.003944643 0.016293581 0.002863584 0.001469964
## P4 0.000789581 0.002281200 0.002863584 0.009595202 0.003210827
## P5 0.001351044 0.002578939 0.001469964 0.003210827 0.009242440
```

```r
#Compute correlation matrix:
cormat <- cor(r)
print("Correlation:")
```

```
## [1] "Correlation:"
```

```r
print(cormat)
```

```
##           P1        P2        P3        P4        P5
## P1 1.0000000 0.1875142 0.1714182 0.1058126 0.1844777
## P2 0.1875142 1.0000000 0.3177469 0.2394518 0.2758225
## P3 0.1714182 0.3177469 1.0000000 0.2290206 0.1197857
## P4 0.1058126 0.2394518 0.2290206 1.0000000 0.3409545
## P5 0.1844777 0.2758225 0.1197857 0.3409545 1.0000000
```

```r
#Compute the vector of variances:
variances <- diag(covmat)
print("Variances:")
```

```
## [1] "Variances:"
```

```r
print(variances)
```

```
##          P1          P2          P3          P4          P5
## 0.005803160 0.009458804 0.016293581 0.009595202 0.009242440
```

```r
#Compute the vector of standard deviations:
stdev <- diag(covmat)^.5
print("STD")
```

```
## [1] "STD"
```

```r
print(stdev)
```

```
##         P1         P2         P3         P4         P5
## 0.07617847 0.09725638 0.12764631 0.09795510 0.09613761
```

```r
#Compute inverse of variance covariance matrix
inv_covmat <- solve(covmat)
```

Part c: Use only Exxon-Mobil and Boeing stocks: For these 2 stocks find the composition, expected return, and standard deviation of the minimum risk portfolio

```r
ones_2 = rep(1,2)

r_2 = r[c("P1","P5")]

means_2 <- colMeans(r_2)
covmat_2 <- cov(r_2)
stdev_2 <- diag(covmat_2)^.5
inv_covmat_2 <- solve(covmat_2)

min_risk_weight_vector_2 <- inv_covmat_2  %*% ones_2 /as.numeric(t(ones_2)  %*%  inv_covmat_2 %*% ones_
min_risk_varp_2 <- t(min_risk_weight_vector_2) %*% covmat_2 %*% min_risk_weight_vector_2
min_risk_Rp_2 <- t(min_risk_weight_vector_2) %*% means_2
min_risk_sigmap_2 <- sqrt(min_risk_varp_2)
print("Composition min-risk portfolio with 2 stocks")
```

```
## [1] "Composition min-risk portfolio with 2 stocks"
```

```
print(min_risk_weight_vector_2)
```

```
##         [,1]
## P1 0.6393153
## P5 0.3606847
```

```
print("Mean of min-risk portolio with 2 stocks")
```

```
## [1] "Mean of min-risk portolio with 2 stocks"
```

```
print(min_risk_Rp_2)
```

```
##             [,1]
## [1,] 0.003413689
```

```
print("Std of min-risk portolio with 2 stocks")
```

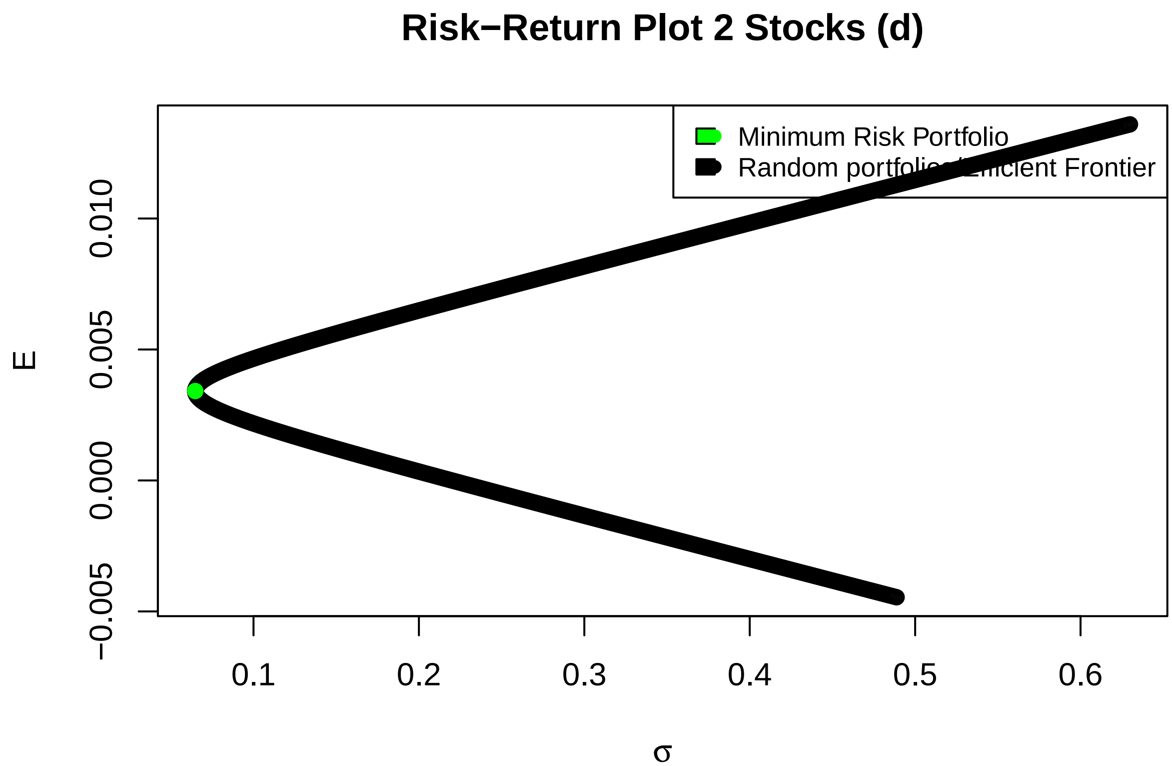
```
## [1] "Std of min-risk portolio with 2 stocks"
```

```
print(min_risk_sigmap_2)
```

```
##            [,1]
## [1,] 0.06478695
```

Part d : Plot the portfolio possibilities curve and identify the efficient frontier on it

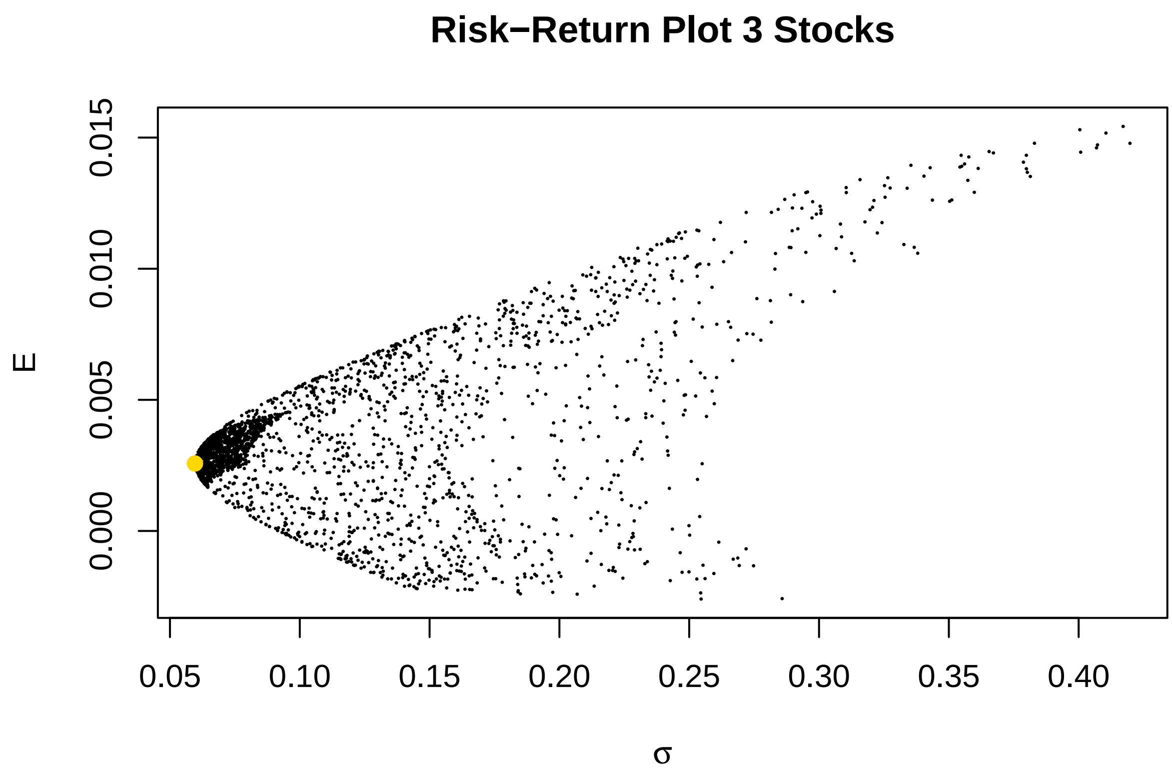
```
x1 = seq(from = -5,to = 5, by = 0.01)
x2 = 1 - x1
means_plot= rep(0,length(x1))
vars_plot = rep(0,length(x1))
for (i in 1:length(x1)){
  coef_temp = c(x1[i],x2[i])
  means_plot[i] = t(coef_temp) %*% means_2
  vars_plot[i] = t(coef_temp) %*% covmat_2 %*% coef_temp
}
plot(sqrt(vars_plot), means_plot, ylab = 'E', xlab = expression(sigma), main="Risk-Return Plot 2 Stocks
points(sqrt(min_risk_varp_2), min_risk_Rp_2, pch=19,lwd=1,col="green")
legend("topright",
       legend=c("Minimum Risk Portfolio","Random portfolios/Efficient Frontier"),
       col=c("green","black"),
       pch = 19,
       fill =c("green","black"),
       cex=0.8)
```



Part e: Use only Exxon-Mobil, McDonalds and Boeing stocks and assume short sales are allowed to answer the following question: For these 3 stocks compute the expected return and standard deviation for many combinations of xa, xb, xc with xa + xb + xc = 1 and plot the cloud of points.

```
coef_3 <- read.table("http://www.stat.ucla.edu/~nchristo/datac183c283/statc183c283_abc.txt", header=T)

n_samples_3 = dim(coef_3)[1]

ones_3 = rep(1,3)

r_3 = r[c("P1","P4","P5")]

means_3 <- colMeans(r_3)
covmat_3 <- cov(r_3)
stdev_3 <- diag(covmat_3)^.5
inv_covmat_3 <- solve(covmat_3)

min_risk_weight_vector_3 <- inv_covmat_3  %*% ones_3 /as.numeric(t(ones_3)  %*%  inv_covmat_3 %*% ones_3
min_risk_varp_3 <- t(min_risk_weight_vector_3) %*% covmat_3 %*% min_risk_weight_vector_3
min_risk_Rp_3 <- t(min_risk_weight_vector_3) %*% means_3
min_risk_sigmap_3 <- sqrt(min_risk_varp_3)

means_plot_3 = rep(0,n_samples_3)
vars_plot_3 = rep(0,n_samples_3)
for (i in 1:n_samples_3){
  coef_temp = c(coef_3$a[i],coef_3$b[i],coef_3$c[i])
  means_plot_3[i] = t(coef_temp) %*% means_3
```

```
  vars_plot_3[i] = t(coef_temp) %*% covmat_3 %*% coef_temp
}
plot(sqrt(vars_plot_3), means_plot_3,pch=19,cex = 0.2,lwd=0.1,ylab = 'E', xlab = expression(sigma), mai
points(min_risk_sigmap_3,min_risk_Rp_3, pch=19,lwd=1,col="gold")
```



PART F AND G SOLVED TOGETHER Part f:Assume Rf = 0.001 and that short sales are allowed. Find the composition, expected return and standard deviation of the portfolio of the point of tangency G and draw the tangent to the efficient frontier of question (e). Part g: Find the expected return and standard deviation of the portfolio that consists of 60% and G 40% risk free asset. Show this position on the capital allocation line (CAL)

```
plot(sqrt(vars_plot_3), means_plot_3,pch=19,cex = 0.2,lwd=0.1,ylab = 'E', xlab = expression(sigma), mai
points(min_risk_sigmap_3,min_risk_Rp_3, pch=19,lwd=1,col="gold")
R_f = 0.001
R_new = matrix(means_3 - R_f)
z = inv_covmat_3 %*% R_new
lambda_g = ones_3 %*% z
x_G = z / as.numeric(lambda_g) # composition
print("Composition of tangent")
```

```
## [1] "Composition of tangent"
```

```
print(x_G)
```

```
##          [,1]
## P1  0.5284782
## P4 -0.4955882
## P5  0.9671100
```

```r
varg <- t(x_G) %*% covmat_3 %*% x_G
Rg <- t(x_G) %*% means_3
print("Expected Return of tangent")
```

```
## [1] "Expected Return of tangent"
```

```r
print(Rg)
```

```
##             [,1]
## [1,] 0.005652415
```

```r
sigmag <- sqrt(varg)
print("Std of tangent")
```

```
## [1] "Std of tangent"
```

```r
print(sigmag)
```

```
##           [,1]
## [1,] 0.1025256
```

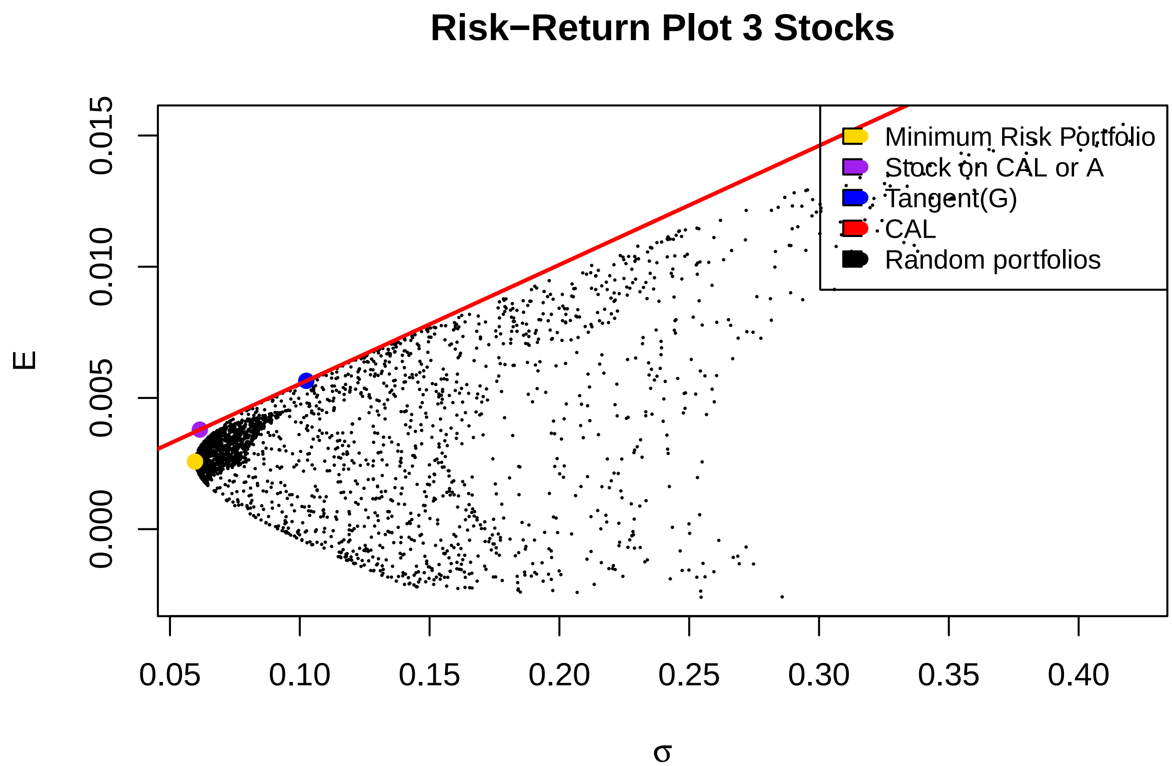
```r
points(sigmag,Rg, pch=19,lwd=1,col="blue")
part_g_pf_R = R_f * 0.4 + 0.6 * Rg
part_g_pf_sigma = (part_g_pf_R - R_f) / ((Rg - R_f)/ sigmag)
points(part_g_pf_sigma,part_g_pf_R, pch=19,lwd=1,col="purple")
abline(a = R_f, b = (Rg - R_f)/sigmag , lwd = 2, col = "red")
print(paste("Expected E of purple(%60-%40): ",part_g_pf_R))
```

```
## [1] "Expected E of purple(%60-%40):  0.00379144914560649"
```

```r
print(paste("STD of purple(%60-%40): ",part_g_pf_sigma))
```

```
## [1] "STD of purple(%60-%40):  0.0615153481280395"
```

```r
legend("topright",
       legend=c("Minimum Risk Portfolio", "Stock on CAL or A", "Tangent(G)","CAL","Random portfolios"),
       col=c("gold","purple", "blue","red","black"),
       pch = 19,
       fill =c("gold","purple", "blue","red","black"),
       cex=0.8)
```



Part h: Refer to question (g). Use the expected value (E) you found in (g) to compute

$$x = \frac{(E - R_f)\,\Sigma^{-1}\left(\overline{R} - R_f 1\right)}{\left(\overline{R} - R_f 1\right)'\,\Sigma^{-1}\left(\overline{R} - R_f 1\right)} \tag{1}$$

What does this x represent? ANSWER: x is the composition of any stock on the CAL which was drawn above. x varies according to E, expected return from stocks. For g, it is the composition when portfolio consists of 60% G and 40% risk free asset. Expected return and risk is written above for that portolfolio.Also, plotted as the the purple point.

```
print(paste("Expected E of purple(%60-%40): ",part_g_pf_R))
```

```
## [1] "Expected E of purple(%60-%40):  0.00379144914560649"
```

```
print(paste("STD of purple(%60-%40): ",part_g_pf_sigma))
```

```
## [1] "STD of purple(%60-%40):  0.0615153481280395"
```

```
E = as.numeric(Rg)
composition_portfolio_g = (E - R_f) *( inv_covmat_3 %*% R_new / as.numeric( t(R_new) %*% inv_covmat_3 %
#This is the composition of portfolio in plotted with purple( 60% risk, 40% free)
print(composition_portfolio_g)
```

```
##          [,1]
## P1  0.5284782
## P4 -0.4955882
## P5  0.9671100
```

Part i: Now assume that short sales are allowed but risk free asset does not exist. Part 1: Using Rf 1 = 0.001 and Rf 2 = 0.002 find the composition of two portfolios A and B (tangent to the efficient frontier - you found the one with Rf 1 = 0.001 in question (f)).

```
R_f1 = 0.001
R_newA = matrix(means_3 - R_f1)
z = inv_covmat_3 %*% R_newA
lambda_g = ones_3 %*% z
x_A = z / as.numeric(lambda_g) # composition
print("Composition when Rf = 0.001")
```

```
## [1] "Composition when Rf = 0.001"
```

```
print(x_A)
```

```
##          [,1]
## P1  0.5284782
## P4 -0.4955882
## P5  0.9671100
```

```
R_f2 = 0.002
R_newB = matrix(means_3 - R_f2)
z = inv_covmat_3 %*% R_newB
lambda_g = ones_3 %*% z
x_B = z / as.numeric(lambda_g) # composition
print("Composition when Rf = 0.002")
```

```
## [1] "Composition when Rf = 0.002"
```

```
print(x_B)
```

```
##          [,1]
## P1  0.5312205
## P4 -1.8026632
## P5  2.2714427
```

Part 2: Compute the covariance between portfolios A and B?

```
cov_AB = t(x_A) %*% covmat_3 %*% x_B
var_A = t(x_A) %*% covmat_3 %*% x_A
var_B = t(x_B) %*% covmat_3 %*% x_B
mean_A = t(x_A) %*% means_3
mean_B = t(x_B) %*% means_3
mean_AB = c(mean_A,mean_B)
print("Covariance between portfolios A and B")
```

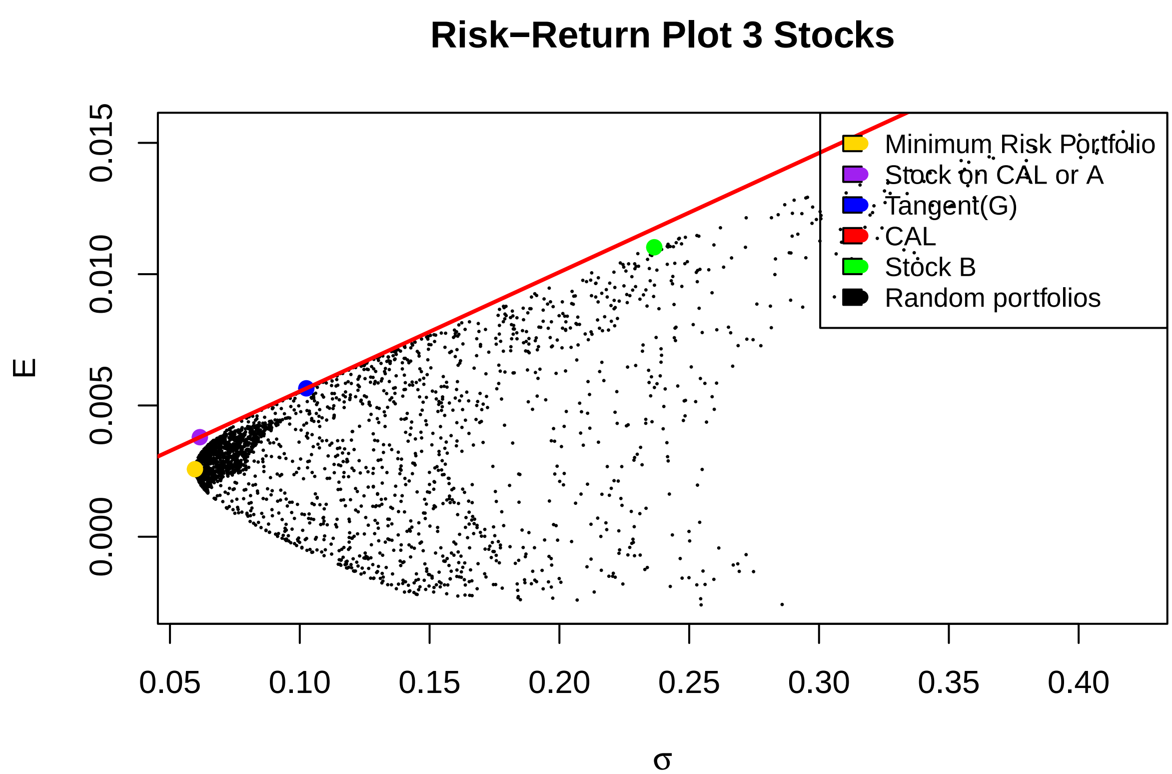
```
## [1] "Covariance between portfolios A and B"
```

```
print(cov_AB)
```

```
##            [,1]
## [1,] 0.02264823
```

```
plot(sqrt(vars_plot_3), means_plot_3,pch=19,cex = 0.2,lwd=0.1,ylab = 'E', xlab = expression(sigma), mai

points(min_risk_sigmap_3,min_risk_Rp_3, pch=19,lwd=1,col="gold")
#points(sqrt(var_A), mean_A,pch=19, ylab = 'E', xlab = expression(sigma), main="Risk-Return Plot",col="
points(sqrt(var_B), mean_B,pch=19, ylab = 'E', xlab = expression(sigma), main="Risk-Return Plot",col="g
points(sigmag,Rg, pch=19,lwd=1,col="blue")
```

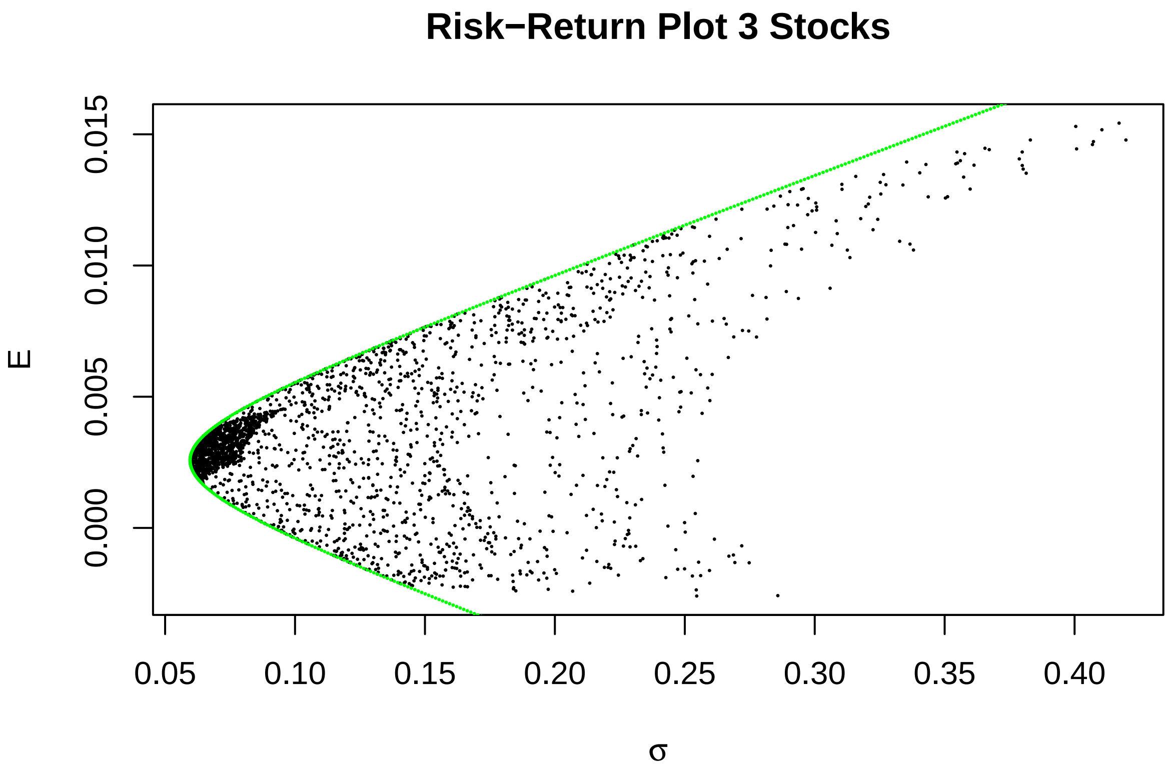
```
points(part_g_pf_sigma,part_g_pf_R, pch=19,lwd=1,col="purple")
abline(a = R_f, b = (Rg - R_f)/sigmag , lwd = 2, col = "red")


legend("topright",
       legend=c("Minimum Risk Portfolio", "Stock on CAL or A", "Tangent(G)","CAL", "Stock B","Random po
       col=c("gold","purple", "blue","red","green","black"),
       pch = 19,
       fill =c("gold","purple", "blue","red","green","black"),
       cex=0.8)
```



Part 3: Use your answers to (1) and (2) to trace out the efficient frontier of the stocks Exxon-Mobil, McDonalds, Boeing. Use a different color to show that the frontier is located on top of the cloud of points from question (e).

```
vec = c(var_A,cov_AB,cov_AB,var_B)
covmat_AB = matrix(vec, nrow = 2, byrow = TRUE)

x1 = seq(from = -5,to = 5, by = 0.01)
x2 = 1 - x1

vars_plot = rep(0,length(x1))
for (i in 1:length(x1)){
  coef_temp = c(x1[i],x2[i])
  means_plot[i] = t(coef_temp) %*% mean_AB
  vars_plot[i] = t(coef_temp) %*% covmat_AB %*% coef_temp
}
```

```
plot(sqrt(vars_plot_3), means_plot_3,pch=19,cex = 0.2,lwd=0.1,ylab = 'E', xlab = expression(sigma), mai
points(sqrt(vars_plot), means_plot,pch=19 , xlab = expression(sigma),cex = 0.1, main="Risk-Return Plot"
```



Part 4:Find the composition of the minimum risk portfolio using the three stocks (how much of each stock) and its expected return, and standard deviation.

```
min_risk_weight_vector_3 <- inv_covmat_3  %*% ones_3 /as.numeric(t(ones_3)  %*%  inv_covmat_3 %*% ones_
min_risk_varp_3 <- t(min_risk_weight_vector_3) %*% covmat_3 %*% min_risk_weight_vector_3
min_risk_Rp_3 <- t(min_risk_weight_vector_3) %*% means_3
min_risk_sigmap_3 <- sqrt(min_risk_varp_3)
plot(sqrt(vars_plot_3), means_plot_3,pch=19,cex = 0.2,lwd=0.1,ylab = 'E', xlab = expression(sigma), mai
print("Minimum Variance Portfolio Composition")
```

```
## [1] "Minimum Variance Portfolio Composition"
```

```
print(min_risk_weight_vector_3)
```

```
##         [,1]
## P1 0.5269063
## P4 0.2536533
## P5 0.2194404
```

```
print("Expected return of min-var portfolio")
```

```
## [1] "Expected return of min-var portfolio"
```

```
print(min_risk_Rp_3)
```

```
##              [,1]
## [1,] 0.00257322
```

```r
print("STD of Minimum Variance Portfolio Composition")
```

```
## [1] "STD of Minimum Variance Portfolio Composition"
```

```r
print(min_risk_varp_3)
```

```
##               [,1]
## [1,] 0.003554475
```

```r
points(min_risk_sigmap_3, min_risk_Rp_3,pch=19 , xlab = expression(sigma),cex = 1, main="Risk-Return Plo
points(sqrt(vars_plot), means_plot,pch=19 , xlab = expression(sigma),cex = 0.1, main="Risk-Return Plot"
legend("topright",
       legend=c("Minimum Risk Portfolio","Random portfolios","Efficient Frontier/Traced with A and B"),
       col=c("blue","black","green"),
       pch = 19,
       fill =c("blue","black","green"),
       cex=0.8)
```