Mean-Variance Analysis, CAPM

Guneykan Ozkaya

October 21, 2019

```
library("data.table")
library("tinytex")
```

```
df <- read.csv(file="/Users/guneykan/Desktop/Mean_Variance_CAPM/17_Industry_Portfolios.CSV")
```

- Consider the 17 industry portolios as the investable risky assets. Use the available historical data to estimate the vector of expected returns, E(R), and the variance-covariance matrix, V.
- Data taken from Prof. French website, it can be found in the repository, I cleaned data before importing, frequency is monthly.
- Note that, we divide returns by 100 due to returns are given in percent terms.

```
df<-data.table(df)
df[, X:=NULL]
df<-df/100
head(df)
```

```
##       Food   Mines     Oil  Clths   Durbl   Chems  Cnsum   Cnstr   Steel
## 1:  0.0048  0.0378 -0.0141 0.0603 -0.0162  0.0846 0.0142  0.0231  0.0407
## 2:  0.0291  0.0069  0.0360 0.0019 -0.0196  0.0570 0.0584  0.0433  0.0217
## 3:  0.0120  0.0110 -0.0368 0.0026  0.0024  0.0548 0.0121 -0.0006  0.0015
## 4: -0.0306 -0.0079 -0.0102 0.0037 -0.0607 -0.0476 0.0069 -0.0479 -0.0385
## 5:  0.0637  0.0438 -0.0001 0.0222 -0.0195  0.0527 0.0463  0.0245  0.0386
## 6: -0.0057  0.0166  0.0292 0.0218  0.0379  0.0520 0.0227  0.0430  0.0375
##      FabPr   Machn    Cars   Trans   Utils   Rtail   Finan   Other
## 1:  0.0877  0.0379  0.1743  0.0184  0.0704  0.0013  0.0037  0.0122
## 2: -0.0556  0.0235  0.0396  0.0457 -0.0169 -0.0068  0.0446  0.0311
## 3: -0.0413 -0.0065  0.0557  0.0030  0.0204  0.0021 -0.0123  0.0182
## 4: -0.0513 -0.0329 -0.0829 -0.0292 -0.0263 -0.0226 -0.0516 -0.0088
## 5:  0.0357  0.0454 -0.0028  0.0218  0.0371  0.0644  0.0224  0.0138
## 6: -0.0385  0.0129  0.1080  0.0348 -0.0017  0.0061  0.0268  0.0182
```

- Obtaining Variance-Covariance Matrix (V), inverse Variance-Covariance Matrix (V^{-1}), and mean return of each industry ($E(R)$).
- Note that, we also create 2 seperate vectors namely "vec_1s" and "vec_1s_t" which are vector of 1s with 17 elements and the transpose of this vector. We do this for the sake of practically.

```
df_m<-as.matrix(df)
cov_matrix<-cov(df_m)
cov_matrix_inv<-solve(cov_matrix)
mean_rets<-unlist(c(df[, lapply(.SD, mean)]))
vec_1s<-c(rep(1,17))
vec_1s_t<-t(vec_1s)
```

A) Here we are finding the composition of two portfolios in the frontier, the mean, variance, and the standard deviation of those portfolios, and the covariance between them.

$$\text{We obtain } \omega_1 \text{ and } \omega_2 \text{ through,} \quad \omega = \lambda_1 V^{-1} E(R) + \lambda_2 V^{-1} 1_N$$

First we assign $\lambda_1 = 1$ and $\lambda_2 = 0$, then we assign $\lambda_1 = 0$ and $\lambda_2 = 1$.

We obtain:

$$\omega_1 = \frac{V^{-1} E(R)}{1_N^{'} V^{-1} E(R)} \quad \text{and} \quad \omega_2 = \frac{V^{-1} 1_N}{1_N^{'} V^{-1} 1_N}$$

Note that, we divide ω_1 by $1_N^{'} V^{-1} E(R)$ and ω_2 by $1_N^{'} V^{-1} 1_N$ to force the sum of the wieghts to be 1.

```
# Obtaining Omega 1
w_1_nomin<-cov_matrix_inv%*%mean_rets
w_1_denom<-1/vec_1s_t%*%cov_matrix_inv%*%mean_rets
w_1<-w_1_nomin%*%w_1_denom

# Obtaining Omega 2
w_2_nomin<-cov_matrix_inv%*%vec_1s
w_2_denom<-1/vec_1s_t%*%cov_matrix_inv%*%vec_1s
w_2<-w_2_nomin%*%w_2_denom

# Obtaining mean returns and the variance of each portfolio
port_1_mean_ret<-sum(w_1 * mean_rets)
port_2_mean_ret<-sum(w_2 * mean_rets)
var_w_1<-t(w_1)%*%cov_matrix%*%w_1
var_w_2<-t(w_2)%*%cov_matrix%*%w_2
```

```
# Obtaining returns of each industry with their corresponding weights in each month.
port_1_asset_rets<-t(matrix(mapply("*", t(df_m), w_1), nrow = ncol(df), ncol = nrow(df)))
port_2_asset_rets<-t(matrix(mapply("*", t(df_m), w_2), nrow = ncol(df), ncol = nrow(df)))

# Obtaining returns of each portfolio in each month
port_1_rets<-matrix(apply(port_1_asset_rets, 1, sum), nrow = nrow(df), ncol = 1)
port_2_rets<-matrix(apply(port_2_asset_rets, 1, sum), nrow = nrow(df), ncol = 1)

# Obtaining Covariance between 2 Portfolios (w_1 and w_2)
cov_1_2<-cov(port_1_rets, port_2_rets)
```

- Mean Returns, Variances, Std. Deviations and the Covariance

```
cat("", "Omega 1 Mean Return:", port_1_mean_ret,"\n",
    "Omega 2 Mean Return:", port_2_mean_ret, "\n",
    "Omega 1 Variance:", var_w_1, "\n",
    "Omega 2 Variance:", var_w_2, "\n",
    "Omega 1 Std. Dev.:", sqrt(var_w_1), "\n",
    "Omega 2 Std. Dev.:", sqrt(var_w_2), "\n",
    "Covariance Between Omega 1 and Omega 2:", cov_1_2)
```

```
##  Omega 1 Mean Return: 0.01021514
##  Omega 2 Mean Return: 0.008957041
##  Omega 1 Variance: 0.001499317
##  Omega 2 Variance: 0.001314662
##  Omega 1 Std. Dev.: 0.03872102
##  Omega 2 Std. Dev.: 0.03625826
##  Covariance Between Omega 1 and Omega 2: 0.001314662
```

B) Denote by p and $(1-p)$ the weights of those two portfolios in a combined portfolio, p, and obtain the mean and variance of the resulting combined portfolios for different values of p.

We obtain return of each portfolio through:

$$E(R_P) = pE(R_{\omega_1}) + (1-p)E(R_{\omega_2})$$

We obtain variance of each portfolio through:

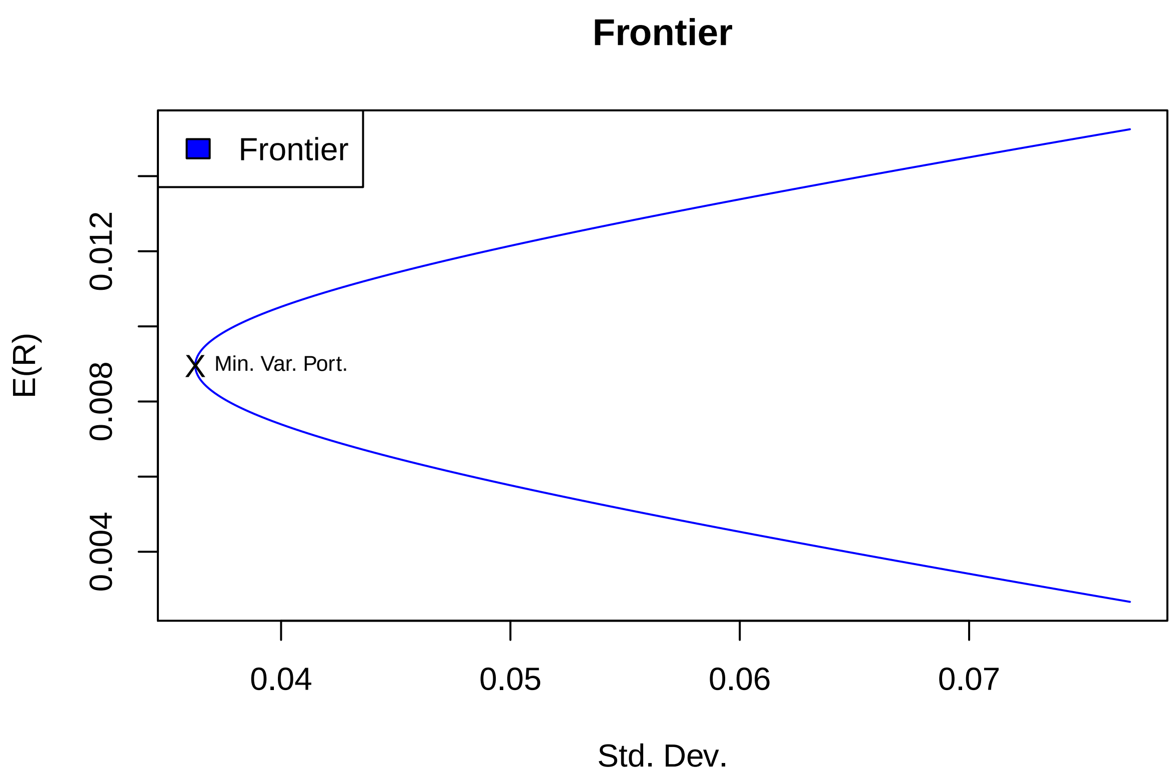
$$Var(R_P) = p^2 Var(R_{\omega_1}) + (1-p)^2 Var(R_{\omega_2}) + 2p(1-p)Cov(R_{\omega_1}, R_{\omega_2})$$

```
# Generating a sequence of Ps
p<-seq(5, -5, -0.01)

# Holder vectors (We will assign return and variance of each Portfolio P.)
port_rets<-c(1:length(p))
port_var<-c(1:length(p))
for(a in c(1:length(p))){
  port_rets[a]<-(p[a]*port_1_mean_ret)+((1-p[a])*port_2_mean_ret)
  port_var[a]<-(p[a]**2)*var_w_1 + ((1-p[a])**2)*var_w_2 + 2*p[a]*(1-p[a])*cov_1_2
}
```

C) Lets draw the frontier in the mean-standard deviation space and locate the minimum variance portfolio and the efficient frontier in the graph.

- In order to obtain frotier we need to obtain different combinations of portfolios denoted by ω_1 and ω_2. In question (B), we obtained different combinations of ω_1 and ω_2 by using a sequence of different weights (p and $(1-p)$). Plotting those portfolios will give us the Frontier. The portfolio denoted by ω_2 is the Minumum Variance Portfolio.

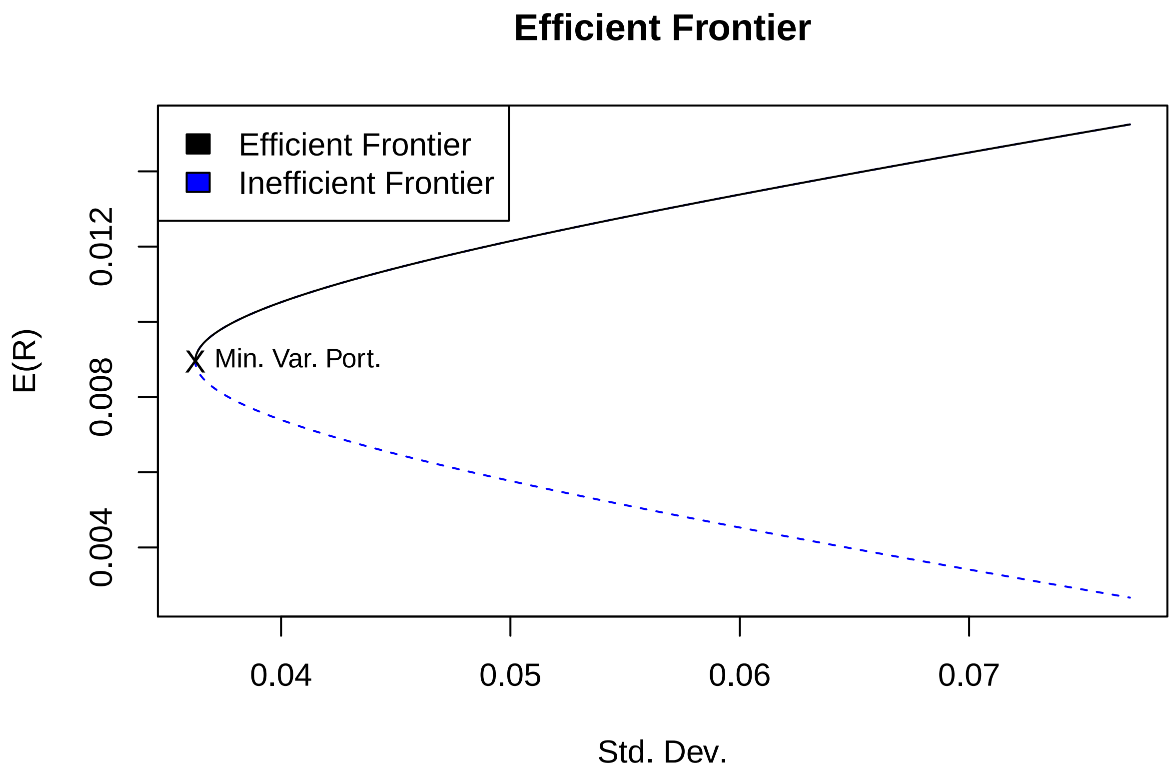
```
# Drawing frontier and locating Minumum Variance Portfolio.
port_std<-sqrt(port_var)
plot(port_std, port_rets,
main="Frontier",
ylab="E(R)",
xlab = "Std. Dev.",
type="l",
col="blue")
points(sqrt(var_w_2), port_2_mean_ret, pch="X", lwd=10)
text(sqrt(var_w_2), port_2_mean_ret, labels="Min. Var. Port.", cex= 0.7, pos = 4)
legend("topleft",
c("Frontier"),
fill=c("blue")
)
```



- In order to obtain efficient frontier in the absence of a Risk Free asset, we need to consider all the portfolios above the Minumum Variance Portfolio in terms of Expected Return and Std. Deviation. Since all the portfolios below Minumum Variance Portfolio will have same standard deviation but lower expected return compared to portfolios above the Minumum Variance Portfolio, portfolios that are below the Minumum Variance Portfolio are inefficient.

```r
# Locating portfolios which have same Std. Dev. but higher returns
eff_frontier_ret<-c()
eff_frontier_var<-c()
for(a in c(1:length(port_var))){
  if(port_var[a]>=var_w_2&port_rets[a]>=port_2_mean_ret){
    eff_frontier_ret<-c(eff_frontier_ret, port_rets[a])
    eff_frontier_var<-c(eff_frontier_var, port_var[a])
  }else{
    next
  }
}
```

```r
# Drawing efficient and inefficient frontier
plot(port_std, port_rets,
main="Efficient Frontier",
ylab="E(R)",
xlab = "Std. Dev.",
type="l",
lty=2,
col="blue")
points(sqrt(var_w_2), port_2_mean_ret, pch="X", lwd=10)
lines(sqrt(eff_frontier_var), eff_frontier_ret)
text(sqrt(var_w_2), port_2_mean_ret, labels="Min. Var. Port.", cex= 0.8, pos = 4)
legend("topleft",
c("Efficient Frontier", "Inefficient Frontier"),
fill=c("black", "blue")
)
```



D) Suppose you are willing to invest in a portfolio with a standard deviation as high as 5% per month, but not more than that. What portfolio should we choose in order to maximize the expected return of your investment?

- Given the Risk and Return relationship, in order to maximize our expected return we need to fix our Std. Dev. to 5% since portfolios with higher Std. Devs. will yield higher expected returns.
- First we define the Variance formula of the portfolio which we introduced in part (B). Then we impose a constraint on it by fixing it to 5%

The constrained function is given by:

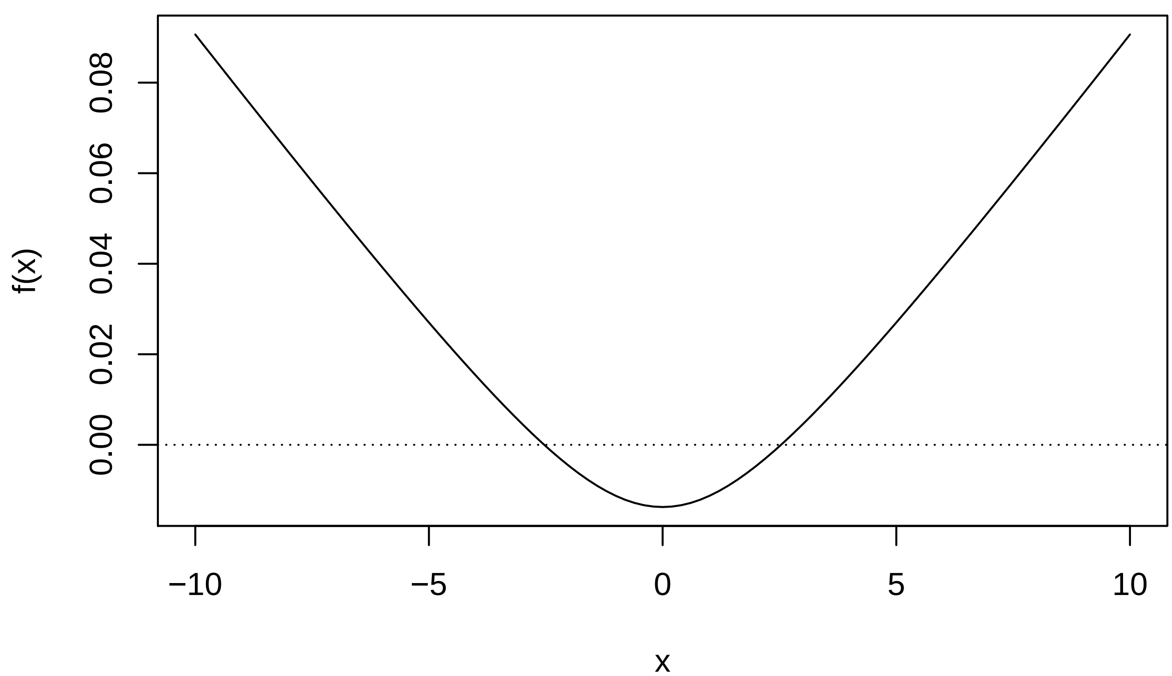
$$Var(R_P) = p^2 Var(R_{\omega_1}) + (1-p)^2 Var(R_{\omega_2}) + 2p(1-p)Cov(R_{\omega_1}, R_{\omega_2}) = 0.05^2$$

$$Var(R_P) - 0.05^2 = p^2 Var(R_{\omega_1}) + (1-p)^2 Var(R_{\omega_2}) + 2p(1-p)Cov(R_{\omega_1}, R_{\omega_2}) - 0.05^2 = 0$$

```r
# Defining the variance formula with 5% constraint
f<-function(x){
  sqrt((x^2)*var_w_1+((1-x)^2)*var_w_2+2*x*(1-x)*cov_1_2)-0.05
}
```

- Then we find the roots of the function, which will give us two candidates for p. These two candidates correspond to weights of 2 different portfolios in the frontier, both will have the same standard deviation (since we fixed it to 5%), but one of them will have a higher expected return. Meaning 1 of the portfolios will be in the efficient frontier while the other one will be located in the inefficient frontier. In order to maximize our expected return, we need to choose the one that is located in the efficient frontier (the one that yields higher expected return).

```r
#Checking intervals of the roots, suppressWarnings for some nonsense warning
suppressWarnings({curve(f, from = -10, to = 10); abline(h = 0, lty = 3)})
```




```r
#Obtain the roots
candidate_p_1<-uniroot(f, lower = -5, upper = 0)$root
candidate_p_2<-uniroot(f, lower = 0, upper = 5)$root

#check the returns with respect to each root
candidate_p_1_ret<-t(c((candidate_p_1*w_1+(1-candidate_p_1)*w_2)))%*%mean_rets
candidate_p_2_ret<-t(c((candidate_p_2*w_1+(1-candidate_p_2)*w_2)))%*%mean_rets
```

- Candidates

```
cat("", "Candidate P_1 Ret:", candidate_p_1_ret,"\n",
    "Candidate P_2 Ret:", candidate_p_2_ret, "\n",
    "Thus we choose candidate 2")
```

```
##  Candidate P_1 Ret: 0.005769543
##  Candidate P_2 Ret: 0.01214454
##  Thus we choose candidate 2
```

```
ret_5_std<-candidate_p_2_ret
std_5<-sqrt((candidate_p_2^2)*var_w_1+((1-candidate_p_2)^2)*var_w_2+
                2*candidate_p_2*(1-candidate_p_2)*cov_1_2)
opt_w_5_std<-(candidate_p_2*w_1+(1-candidate_p_2)*w_2)
```

- Expected Return and the Std. Deviation of the Portfolio

```
cat("", "Expected Return of the Portfolio with Max. 5% Std. Dev.:", ret_5_std,"\n",
    "Std. Dev. of the Portfolio:", std_5, "\n")
```

```
##  Expected Return of the Portfolio with Max. 5% Std. Dev.: 0.01214454
##  Std. Dev. of the Portfolio: 0.04999983
```

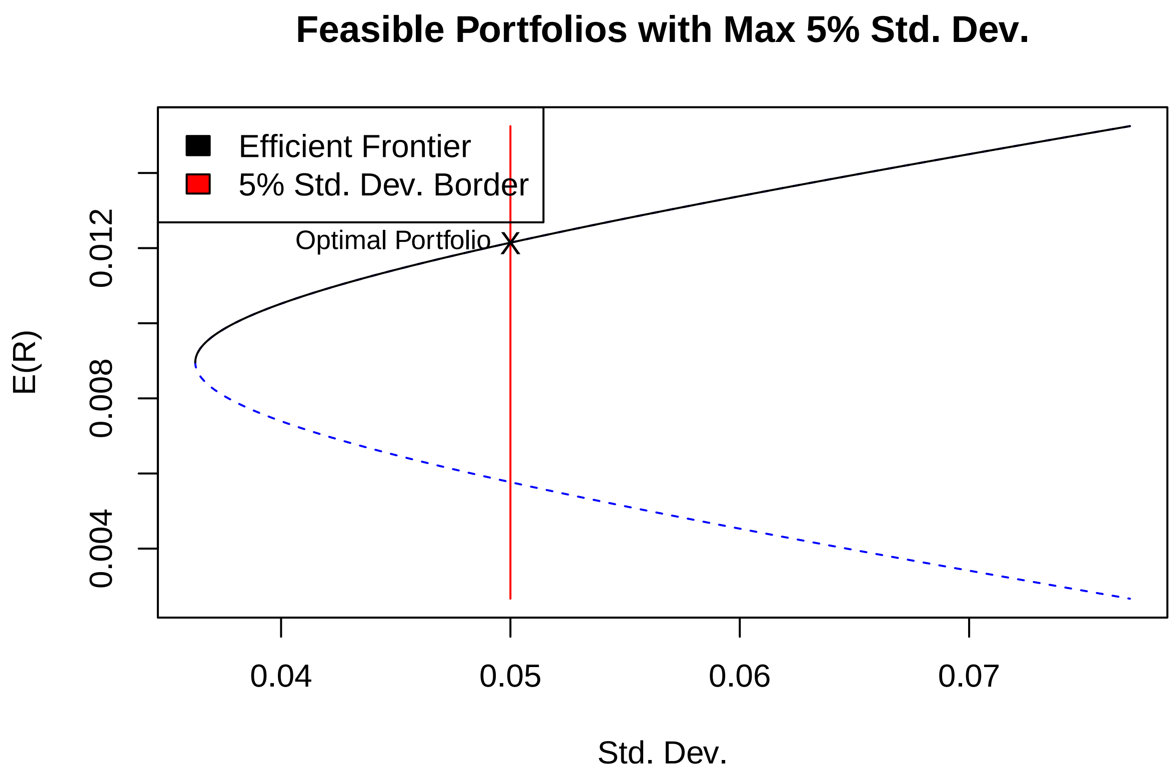
- Weight Composition of the Portfolio

```
opt_w_5_std
```

```
##              [,1]
## Food   0.41405152
## Mines  0.15736761
## Oil    0.27947080
## Clths  0.10733416
## Durbl -0.55158226
## Chems  0.19113013
## Cnsum  0.47306797
## Cnstr -0.10883322
## Steel -0.55923695
## FabPr -0.07216121
## Machn  0.58350550
## Cars   0.28401891
## Trans  0.03005285
## Utils -0.05161968
## Rtail  0.15479769
## Finan -0.04798373
## Other -0.28338009
```

- Optimal portfolio with 5% Std. Dev. expressed graphically

```
plot(port_std, port_rets,
main="Feasible Portfolios with Max 5% Std. Dev.",
ylab="E(R)",
xlab = "Std. Dev.",
type="l",
lty=2,
col="blue")
lines(sqrt(eff_frontier_var), eff_frontier_ret)
lines(rep(0.05, length(port_rets)), port_rets, col="red")
points(std_5, ret_5_std, pch="X", lwd=10)
text(std_5, ret_5_std, labels="Optimal Portfolio", cex= 0.8, pos = 2)
legend("topleft",
c("Efficient Frontier", "5% Std. Dev. Border"),
fill=c("black", "red")
)
```



E) From now on suppose we introduce a risk-free asset paying an interest rate $R_f = 0.4\%$ per month. And we are going to do the same analysis in the presence of a risk free asset

- We obtain the composition of the tangeny portfolio through:

$$\omega_T = \frac{V^{-1}\left(E(R) - R_f 1_N\right)}{1_N' V^{-1}\left(E(R) - R_F 1_N\right)}$$

```r
# Obtaining composition, expected return and the std. dev. of the tangency portfolio (above formula)
r_f<-0.4/100
tang_port_nomin<-cov_matrix_inv%*%(mean_rets-(r_f*vec_1s))
tang_port_denom<-1/vec_1s_t%*%cov_matrix_inv%*%(mean_rets-(r_f*vec_1s))
tang_port_w<-tang_port_nomin%*%tang_port_denom
tang_port_ret<-t(tang_port_w)%*%mean_rets
tang_port_std<-sqrt(t(tang_port_w)%*%cov_matrix%*%tang_port_w)
```

- Tangency Portfolio Mean Return and the Std. Deviation

```r
cat("", "Tangency Port. Mean Return:", tang_port_ret,"\n",
    "Tangency Port. Std. Dev.:", tang_port_std)
```

```
##  Tangency Port. Mean Return: 0.01123033
##  Tangency Port. Std. Dev.: 0.04378999
```

- In order to locate the new efficient frontier we need to obtain Capital Market Line

Capital Market Line (CML) is given by:

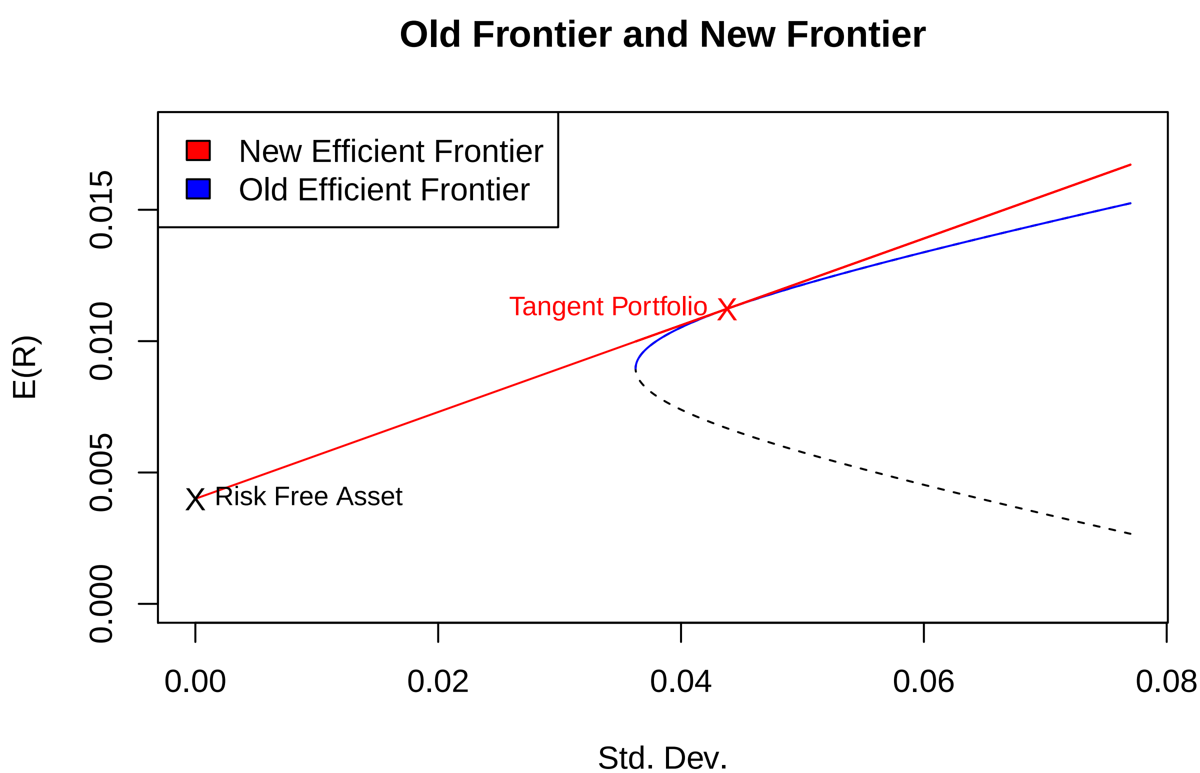
$$E(R_e) = R_f + \frac{E(R_T) - R_f}{\sigma_T}\sigma_e$$

```r
# Defining CML see above, x denotes to std. dev. of risky portfolio, r denotes to r_f
cml<-function(x, r){
  r+(tang_port_ret-r)*(x/tang_port_std)
}

# Locating risk free asset on CML
eff_ports<-c(cml(0, r_f))

# Obtaining all the portfolios on CML (new efficient frontier)
efficient_rets<-c()
for(a in port_std){
  z<-cml(a, r_f)
  eff_ports<-c(eff_ports, z)
}
```

- Old Frontier and the New Frontier

```r
plot(c(0, port_std), c(NaN, port_rets),
main="Old Frontier and New Frontier",
ylab="E(R)",
xlab = "Std. Dev.",
type="l",
```

```r
lty=2,
col="black",
ylim=c(0,0.018))
lines(sqrt(eff_frontier_var), eff_frontier_ret, col ="blue")
lines(c(0, port_std), eff_ports, col = "red")
points(tang_port_std, tang_port_ret, pch="X", lwd=10, col="red")
points(0, r_f, pch="X", lwd=10)
text(tang_port_std, tang_port_ret, labels="Tangent Portfolio", cex= 0.8, pos = 2, col = "red")
text(0, r_f, labels="Risk Free Asset", cex= 0.8, pos = 4)
legend("topleft",
c("New Efficient Frontier","Old Efficient Frontier"),
fill=c("red","blue")
)
```



F) Assume that an investor's preferences can be represented by the following expected utility function: $E(R_p) - \alpha\sigma_p^2$, with $\alpha > 0$. Using the two-fund separation theorem, we are going to solve first analytically the investor's optimal investment in the tangency portfolio, p, as a function of α ,$E(R_T)$, and σ_T^2. Then we are going to assume $\alpha = 4$, and we will use the previous solution to compute p as well as the investor's optimal investment in each one of the 18 assets as a fraction of her wealth.

- Analytical solution

Investor's Problem:

$$Max \quad U\left(E(R_p),\sigma_p^2\right) = E(R_p) - \alpha\sigma_p^2$$

Since she is going to divide her wealth between tangency portfolio and risk free asset, her problem can be written as:

$$Max \quad U\left(E(R_p),\sigma_p^2\right) = pE(R_T) + (1-p)R_f - \alpha p^2\sigma_T^2$$

In order to maximize utility function, first, we need to check second order condition:

$$\frac{\partial\ U(E(R_p),\sigma_p^2)}{\partial\ p} = E(R_T) - R_f - 2\alpha p\sigma_T^2$$

and

$$\frac{\partial^2\ U(E(R_p),\sigma_p^2)}{\partial\ p^2} = -2\alpha\sigma_T^2 < 0,\ \text{since } \alpha > 0 \text{ and } \sigma_T^2 > 0\text{, second order condition holds.}$$

Thus:

$$\frac{\partial\ U(E(R_p),\sigma_p^2)}{\partial\ p} = E(R_T) - R_f - 2\alpha p^*\sigma_T^2 = 0$$

where p^* is the proportion of investor's wealth which should be invested in tangency portfolio in order to maximize her utility

p^* can be written as:

$$p^* = \frac{E(R_T) - R_f}{2\alpha\sigma_T^2}$$

- Now Assume $\alpha = 4$

```r
alpha<-4
# Using the formula we obtained above
p_nomin<-(tang_port_ret-r_f)
p_denom<-2*alpha*tang_port_std^2
p_opt<-p_nomin/p_denom
# Proportion of her wealth in risky assets
risky_asset_w<-tang_port_w%*%p_opt


# Proportion of her wealth in risk free asset
r_f_w<-1-sum(tang_port_w%*%p_opt)

# Investor's optimal weights in each of 18 assets
inv_opt_w<-as.matrix(c(r_f_w, risky_asset_w)*100)
rownames(inv_opt_w)<-c("Risk Free", names(df))
colnames(inv_opt_w)<-"Weights(%)"
```

```r
# Investor's expected return and the std. dev. of her portfolio
inv_ret<-p_opt*tang_port_ret+(1-p_opt)*r_f
inv_std<-sqrt((p_opt^2)*tang_port_std^2)
```

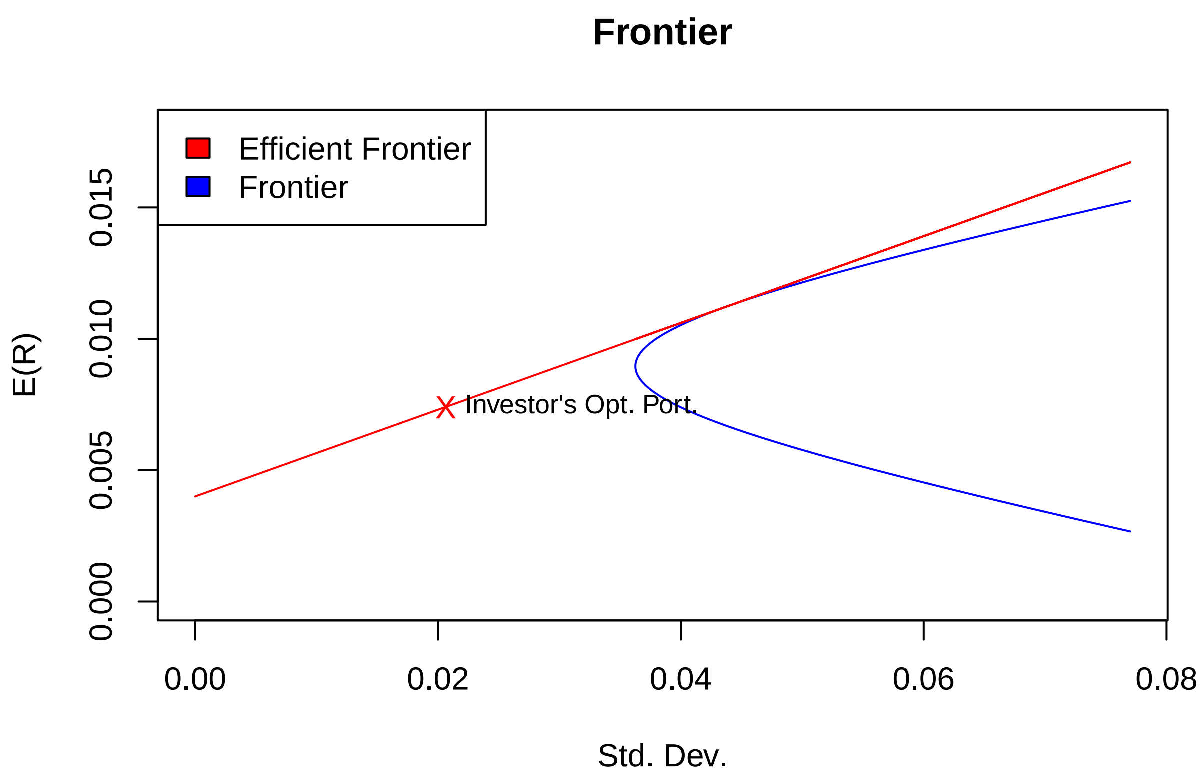
- Proportion of investor's wealth in each asset

```r
inv_opt_w
```

```
##            Weights(%)
## Risk Free  52.8677088
## Food       18.8565005
## Mines       6.4647633
## Oil        11.3612120
## Clths       6.1724163
## Durbl     -20.7415603
## Chems       6.0971424
## Cnsum      19.7059974
## Cnstr      -6.2028913
## Steel     -20.4685166
## FabPr      -0.8844782
## Machn      17.7518563
## Cars        9.9984348
## Trans       0.5961059
## Utils       1.3029899
## Rtail       4.6370313
## Finan      -6.6881704
## Other      -0.8265422
```

- Locating Investor's portfolio in the mean-variance diagram, we see that investor is risk-averse thus she needs to invest in Risk free asset and short the risky portfolio (Tangency Portfolio). Note that tangency portfolio does not mean it is risk free, it just means it is the optimal portfolio.

```r
plot(c(0, port_std), c(NaN, port_rets),
main="Frontier",
ylab="E(R)",
xlab = "Std. Dev.",
type="l",
col="blue",
ylim=c(0,0.018))
lines(c(0, port_std), eff_ports, col = "red")
points(inv_std, inv_ret, pch="X", lwd=5, col="red")
text(inv_std, inv_ret, labels="Investor's Opt. Port.", cex= 0.8, pos = 4, col = "black")
legend("topleft",
c("Efficient Frontier","Frontier"),
fill=c("red","blue")
)
```



G) Suppose now that the lending and borrowing rates for the risk-free asset are $R_{f,l} = 0\%$ and $R_{f,b} = 0.6\%$ per month, respectively. Lets see what happens to efficient frontier.

```
# Defining borrowing and lending rates
r_f_l<-0
r_f_b<-0.6/100

# Obtaining new tangency portfolio with respect to borrowing rate
tang_port_nomin_new<-cov_matrix_inv%*%(mean_rets-(r_f_b*vec_1s))
tang_port_denom_new<-1/vec_1s_t%*%cov_matrix_inv%*%(mean_rets-(r_f_b*vec_1s))
tang_port_w_new<-tang_port_nomin_new%*%tang_port_denom_new
tang_port_ret_new<-t(tang_port_w_new)%*%mean_rets
tang_port_std_new<-sqrt(t(tang_port_w_new)%*%cov_matrix%*%tang_port_w_new)

# Obtaining new tangency portfolio with respect to lending rate
tang_port_nomin_new_1<-cov_matrix_inv%*%(mean_rets-(r_f_l*vec_1s))
tang_port_denom_new_1<-1/vec_1s_t%*%cov_matrix_inv%*%(mean_rets-(r_f_l*vec_1s))
tang_port_w_new_1<-tang_port_nomin_new_1%*%tang_port_denom_new_1
tang_port_ret_new_1<-t(tang_port_w_new_1)%*%mean_rets
tang_port_std_new_1<-sqrt(t(tang_port_w_new_1)%*%cov_matrix%*%tang_port_w_new_1)
# new CML function
cml_1<-function(x, r){
  r+(tang_port_ret_new-r)*(x/tang_port_std_new)
}
cml_2<-function(x, r){
```

```r
  r+(tang_port_ret_new_1-r)*(x/tang_port_std_new_1)
}
```

```r
# Obtaining new efficient portfolios
eff_ports_new<-c(cml_1(0, r_f_b))

# This is for plotting purposes
eff_ports_new_arb<-c(cml_1(0, r_f_b))
efficient_rets_new<-c()
for(a in port_std){
  z<-cml_1(a, r_f_b)
  eff_ports_new<-c(eff_ports_new, z)
}
# Plotting Purposes
eff_ports_new<-eff_ports_new[2:length(eff_ports_new)]
for(a in port_std){
  z<-cml_1(a, r_f_b)
  eff_ports_new_arb<-c(eff_ports_new_arb, z)
}
```

```r
eff_ports_new_1<-c(cml_2(0, r_f_l))
eff_ports_new_arb_1<-c(cml_2(0, r_f_l))
efficient_rets_new_1<-c()
for(a in port_std){
  z<-cml_1(a, r_f_l)
  eff_ports_new_1<-c(eff_ports_new_1, z)
}
# Plotting Purposes
eff_ports_new_1<-eff_ports_new_1[2:length(eff_ports_new_1)]
for(a in port_std){
  z<-cml_2(a, r_f_l)
  eff_ports_new_arb_1<-c(eff_ports_new_arb_1, z)
}
```

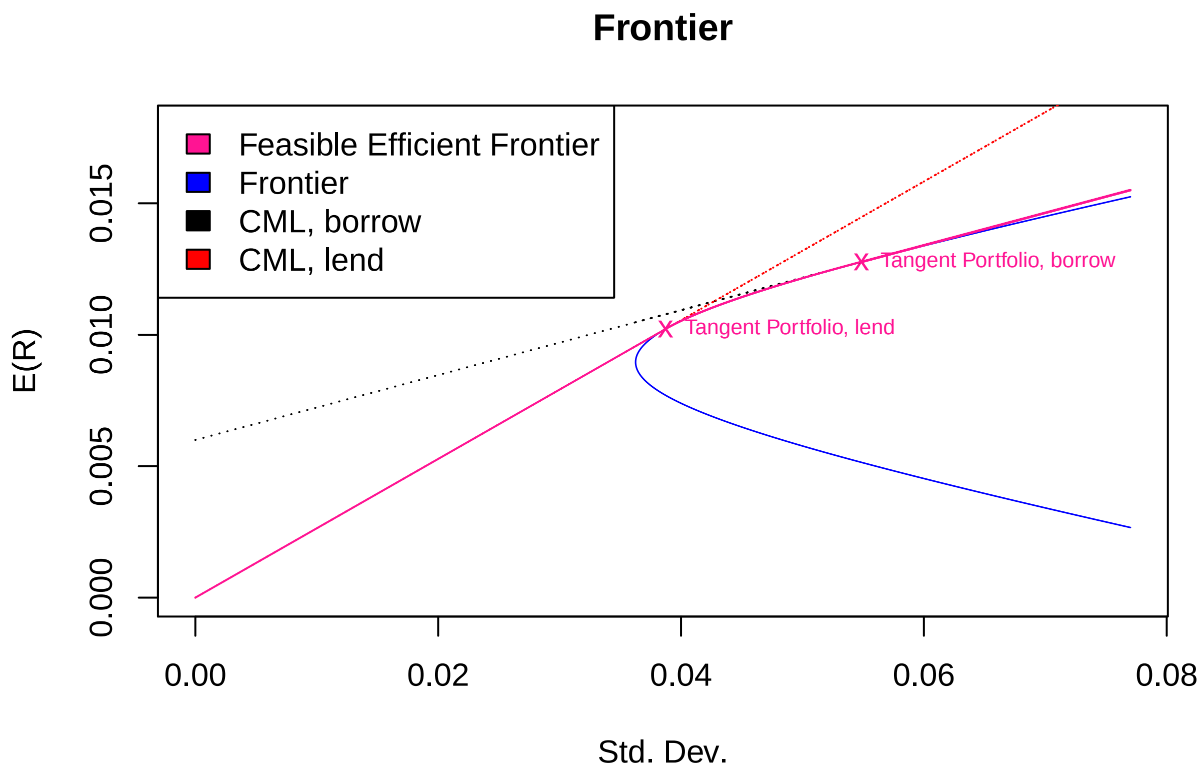
```r
# Obtaining the portfolios where we borrow and invest more on tangeny portfolio
r_f_b_ret<-c()
r_f_b_std<-c()
for(a in c(1:length(port_std))){
  if(port_std[a]>=tang_port_std_new){
    r_f_b_ret<-c(r_f_b_ret, eff_ports_new[a])
    r_f_b_std<-c(r_f_b_std, port_std[a])
  }
}
```

```r
# Obtaining the rest of the efficient frontier, since we will not able to lend, we need to invest in ri
r_f_l_ret<-c()
r_f_l_std<-c()
for(a in c(1:length(port_std))){
  if(port_std[a]<=tang_port_std_new&port_rets[a]<=tang_port_ret_new&port_rets[a]>=tang_port_ret_new_1){
    r_f_l_ret<-c(r_f_l_ret, port_rets[a])
    r_f_l_std<-c(r_f_l_std, port_std[a])
  }else{
    next
```

```
  }
}
```

- New efficient frontier. since we cannot lend but only borrow (we can lend but there is no point of doing so), in order to obtain lower standard deviation compared to tangency portfolio, we need to invest in risky portfolios which located between 2 tangency points, we cannot go long on a risk free asset, because its return is “0” thus there is not point of doing so. The most optimal portfolio with the lowest variance we can invest is tangeny portfolio with a risk-free rate is 0 and short the tangency portfolio if we want to achieve lower standard deviation.

```
plot(c(0, port_std), c(NaN, port_rets),
main="Frontier",
ylab="E(R)",
xlab = "Std. Dev.",
type="l",
col="blue",
ylim=c(0,0.018),
lwd=0.8)
points(tang_port_std_new, tang_port_ret_new, pch="x", lwd=8, col="deeppink1")
points(tang_port_std_new_1, tang_port_ret_new_1, pch="x", lwd=8, col="deeppink1")
lines(c(0, port_std), eff_ports_new_arb, col = "black", lty=3, lwd=1)
lines(c(0, port_std), eff_ports_new_arb_1, col = "red", lty=3, lwd=1)
lines(r_f_b_std, r_f_b_ret, col="deeppink1", type = "s",  lwd=1)
lines(r_f_l_std, r_f_l_ret, col="deeppink1", type = "s",  lwd=1)
segments(0, 0, tang_port_std_new_1, tang_port_ret_new_1, col="deeppink1", lwd=1)
text(tang_port_std_new, tang_port_ret_new, labels="Tangent Portfolio, borrow", cex= 0.7, pos = 4, col =
text(tang_port_std_new_1, tang_port_ret_new_1, labels="Tangent Portfolio, lend", cex= 0.7, pos = 4, col
legend("topleft",
c("Feasible Efficient Frontier","Frontier", "CML, borrow", "CML, lend"),
fill=c("deeppink1","blue", "black", "red")
)
```


Frontier
Feasible Efficient Frontier
Frontier
CML, borrow
CML, lend
Tangent Portfolio, borrow
Tangent Portfolio, lend
E(R)
0.000
0.005
0.010
0.015
0.00
0.02
0.04
0.06
0.08
Std. Dev.