Project6

Juan Piao

5/23/2020

Project6.

a. Assume the multigroup model holds with short sales allowed. Find the composition of the optimal portfolio and place it on the plot of part (b).

```
#Read your csv file:
a <- read.csv("stockData.csv", sep=",", header=TRUE)
#Convert adjusted close prices into returns:
r1 <- (a[-1,3:ncol(a)]-a[-nrow(a),3:ncol(a)])/a[-nrow(a),3:ncol(a)]
cor_matrix<-cor(r1[2:31])
group_cor<-matrix(rep(0, 25), 5)

for(i in 1:5){
  #print("i is ")
  for( j in 1:5){
    #print (j)
    sub <- cor_matrix[((i-1)*6+1):(6*i), ((j-1)*6+1):(6*j) ]
    if(i==j)
      group_cor[i,j]<- (sum(sub)-6)/12
    group_cor[i,j]<-sum(sub)/36
  }

}

## average return for each stock
means<-colMeans(r1)[2:31]
## standard deviation of the return for each stock
sigma <- (diag(cov(r1))^.5)[2:31]

A<-matrix(rep(0, 25), 5)
for(i in 1:5){
  for(j in 1:5){
    A[i,j] <- 6*group_cor[i,j]/(1-group_cor[i,i])
  }
}
diag(A)<- diag(A)+1
R_f<- 0.005

ratio<- (means-R_f)/sigma
C<- rep(0, 5)
for(i in 1:5){
  C[i]<- sum(ratio[((i-1)*6+1):(6*i)]/(1-group_cor[i,i]))
```

```
}

thi<-solve(A) %*% C
#thi

### cut-off point for each group

cut_off<- group_cor %*% thi
z<- rep(0, 30)
for (i in 1:5){
  z[((i-1)*6+1):(6*i)] <- 1/sigma[((i-1)*6+1):(6*i)]/(1-group_cor[i, i]) * (ratio[((i-1)*6+1):(6*i)]-cu

}

x_mg<-z/sum(z)
```

b. Evaluate your portfolios that you constructed in the previous projects. In your analysis you should include the following:

```
xs<- c()

#Read your csv file:
a2 <- read.csv("stockData-2nd.csv", sep=",", header=TRUE)
#Convert adjusted close prices into returns:
r2 <- (a2[-1,3:ncol(a2)]-a2[-nrow(a2),3:ncol(a2)])/a2[-nrow(a2),3:ncol(a2)]

r2_market<- as.matrix(r2[,1])
#Market (S&P500) performance for the period 2017-01-01 to 2020-04-01:
plot(cumprod(1+r2_market), col="green", lwd=2, type="l", ylim=c(0.6 ,2.6), ylab = "")

#====================================================

#Equal allocation weights:
x <- rep(1/30,30)
xs<- rbind(xs, x)

rp <-  as.matrix(r2[ ,2:31]) %*% x

points(cumprod(1+rp), col="blue", lwd=2, type="l")



#====================================================
 #the minimum risk portfolio(Historical data)
x<- c(0.107725355,  0.012326167,  0.057306374,  -0.074052263,   0.063748588,   -0.009821677,  -0.017705

xs<- rbind(xs, x)
rp <-  as.matrix(r2[ ,2:31]) %*% x

points(cumprod(1+rp), col="pink", lwd=2, type="l")

#====================================================
```

```r
 #the minimum risk portfolio(Single Index Model)
x<-c( -0.001457469, -0.014504793,  0.013751016, -0.019140102,  0.026575490, -0.018386505,  0.055201261,
-0.031337958, -0.040435361, -0.005669566, -0.001622187,  0.140630495,  0.064632253,  0.012131682,
 0.022073505, -0.038407934,  0.052175411, -0.010816757,  0.065381844,  0.030274693,  0.075995750,
 -0.001133786, -0.008008371)
xs<- rbind(xs, x)
rp <-  as.matrix(r2[ ,2:31]) %*% x

points(cumprod(1+rp), col="purple", lwd=2, type="l")

#====================================================
 #the optimum portfolio using historical data with Rf equal to 0.005
x<- c( 0.1598911524,  -0.1104241473, 0.0933535631, -0.0794699011, 0.1783072471, 0.0108672795,
-0.0875140290, 0.2237366317, -0.1215312252, 0.0710850896, 0.3596013404,-0.3084326072, 0.5511108217,
0.4404663962, -0.1479924086, 0.0417453996, -0.2705893021, 0.0069470660, -0.0007269433, -0.2353652933,
0.0408267955, 0.1892911981, -0.2287385620, 0.2900275677, 0.0792995596, -0.0258340844, -0.0471392552,
-0.0147785581,0.0339442060, -0.0919649973)

xs<- rbind(xs, x)
rp <-  as.matrix(r2[ ,2:31]) %*% x

points(cumprod(1+rp), col="sienna", lwd=2, type="l")

#====================================================
#the optimum portfolio using Single index model with Rf equal to 0.005 When short sales allowed
r1<- r2[ , -22]  # remove the stock with negative beta

x<-c( -0.029999688, -0.081893392,  0.027896762, -0.034776020,  0.164765214,  0.009525522,  0.049312502,
  0.143970496, -0.109510387,  0.026919977,  0.169651649,  0.008348592,  0.034948393,  0.212238587,
 -0.002792625, -0.082158628,  0.117783552, -0.078482519,  0.034334949,  0.008267688,  0.014315231,
 -0.247265476,  0.120692452,  0.078701180, -0.002884357,  0.239370188,  0.091978386,  0.047313202,
  0.069428571)
xs<- rbind(xs, x)
```

1. Time plots of the performance of all portfolios compared to the S&P500.

```
## Warning in rbind(xs, x): number of columns of result is not a multiple of vector
## length (arg 2)
```

```r
rp <-  as.matrix(r1[ ,2:30]) %*% x

points(cumprod(1+rp), col="red", lwd=2, type="l")

#====================================================
#the optimum portfolio using Single index model with Rf equal to 0.005 When short sales not allowed
x<- c( 0.000000000, 0.000000000, 0.000000000, 0.000000000, 0.094387331, 0.000000000, 0.020089493, 0.145
xs<- rbind(xs, x)
```

```
## Warning in rbind(xs, x): number of columns of result is not a multiple of vector
## length (arg 2)
```

```r
rp <-  as.matrix(r1[ ,2:30]) %*% x

points(cumprod(1+rp), col="darkcyan", lwd=2, type="l")
```

```r
#====================================================
#the optimum portfolio using Constant Correlation model with Rf equal to 0.005 When short sales allowed
x<- c( -0.031846941, -0.054796052,  0.037647475, -0.045207392,  0.212571163,  0.008226806,  0.008033077
  0.092349378, -0.175385944, -0.057973156,  0.106341965, -0.211676978,  0.129702923,  0.265839337,
  0.049899765, -0.030450552,  0.173691555, -0.130245987,  0.087860522,  0.027272446, -0.034537768,
 -0.067086580,  0.094298095,  0.099006830, -0.034095745,  0.295273261,  0.035622872,  0.037652069,
 0.112013556)
xs<- rbind(xs, x)
```

```
## Warning in rbind(xs, x): number of columns of result is not a multiple of vector
## length (arg 2)
```

```r
rp <-  as.matrix(r1[ ,2:30]) %*% x

points(cumprod(1+rp), col="aquamarine1", lwd=2, type="l")


#====================================================
#the optimum portfolio using Constant Correlation model with Rf equal to 0.005 When short sales not all
x<-c( 0.000000000, 0.000000000, 0.000000000, 0.000000000, 0.152750105, 0.000000000, 0.000000000, 0.0367
xs<- rbind(xs, x)
```

```
## Warning in rbind(xs, x): number of columns of result is not a multiple of vector
## length (arg 2)
```

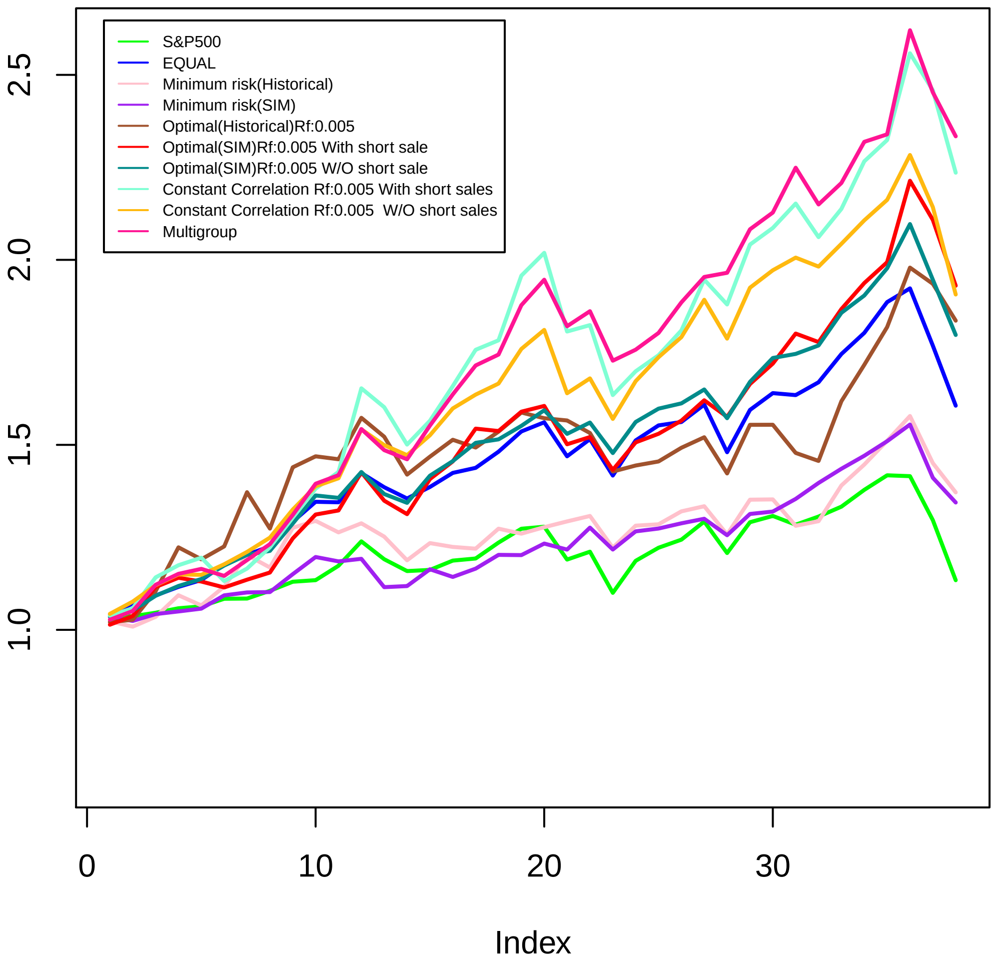
```r
rp <-  as.matrix(r1[ ,2:30]) %*% x

points(cumprod(1+rp), col="darkgoldenrod1", lwd=2, type="l")


#====================================================
#the optimum portfolio using Multi-group model with Rf equal to 0.005 When short sales allowed
rp <-  as.matrix(r2[ ,2:31]) %*% x_mg
xs<- rbind(xs, x_mg)

points(cumprod(1+rp), col="deeppink", lwd=2, type="l")

#====================================================
#Add a legend:
par(oma=c(4,1,1,1), cex = 0.5)
#par(mar = c(2, 1, 1, 1))
#par(xpd=TRUE)
legend("topleft", lty=1, xpd = TRUE,  inset = c(0,
    0), c('S&P500','EQUAL', 'Minimum risk(Historical)', 'Minimum risk(SIM)', 'Optimal(Historical)Rf:0.00
```




```
# xpd = TRUE tells R that it is OK to plot outside the region
#inset = c(x,y) tells R how to move
# the legend relative to the 'bottom' location bty = 'n' means that 'no' box
# will be drawn around it pch and col are the types and colors of points cex
# = 2 makes the legend twice as large as other fonts
```

```
## average return for the market
market_means<-colMeans(r1)[1]
## standard deviation of the return for the market
market_sigma <- sqrt((diag(cov(r1))^.5)[1])

Sharpe<-rep(0, 9)
Diff_excess<- rep(0, 9)
Treynor<- rep(0, 9)
Jensen<- rep(0,9)

for (i in 1:9){
  w <- xs[i, ]
  R_p <- w %*% means
  sigma_p <- sqrt(w %*% cov(r2[ ,2:31]) %*% w)
  ###### Sharpe ratio #########
  Sharpe[i]<- (R_p -R_f)/sigma_p
```

```
  ###### Differential excess return #########
  R_p_line<- R_f + (market_means - R_f)/market_sigma * sigma_p
  Diff_excess[i] <- R_p - R_p_line
  ###### Treynor measure ############
  Treynor[i] <- R_p- R_f

  ###### Jensen differential performance index ############
  rp <-  as.matrix(r2[ ,2:31]) %*% w
  q <- lm( rp ~ r2_market)
    beta_p <- q$coefficients[2]

  R_p_prime = R_f + (market_means- R_f)*beta_p
  Jensen[i] <- R_p - R_p_prime
}

performance_index<-cbind(Sharpe, Diff_excess, Treynor, Jensen)
row.names(performance_index) <- c('EQUAL', 'Minimum risk(Historical)', 'Minimum risk(SIM)', 'Optimal(Hi
performance_index
```

2. Calculate the Sharpe ratio, differential excess return, Treynor measure, and Jensen differential performance index.

```
##                                                Sharpe Diff_excess
## EQUAL                                       0.2962314 0.012215291
## Minimum risk(Historical)                    0.2534831 0.011017369
## Minimum risk(SIM)                           0.2714931 0.008758056
## Optimal(Historical)Rf:0.005                 0.3972099 0.022068067
## Optimal(SIM)Rf:0.005 With short sale        0.4971082 0.018734478
## Optimal(SIM)Rf:0.005 W/O short sale         0.3702775 0.012500306
## Constant Correlation Rf:0.005 With short sales 0.4731548 0.022855621
## Constant Correlation Rf:0.005  W/O short sales 0.3729117 0.014887082
## Multigroup                                  0.5623107 0.024775842
##                                                 Treynor      Jensen
## EQUAL                                       0.012069507 0.012728527
## Minimum risk(Historical)                    0.010864016 0.011344568
## Minimum risk(SIM)                           0.008644133 0.009041362
## Optimal(Historical)Rf:0.005                 0.021871052 0.022363693
## Optimal(SIM)Rf:0.005 With short sale        0.018600595 0.019033521
## Optimal(SIM)Rf:0.005 W/O short sale         0.012380668 0.012872890
## Constant Correlation Rf:0.005 With short sales 0.022684080 0.023222017
## Constant Correlation Rf:0.005  W/O short sales 0.014745598 0.015349146
## Multigroup                                  0.024619185 0.025095073
```

```
### single index model when short sales are not allowed
w <- xs[6, ]
R_p <- w %*% means

rp <-  as.matrix(r2[ ,2:31]) %*% w
q <- lm( rp ~ r2_market)
beta_p <- q$coefficients[2]

R_p_prime = R_f + (market_means- R_f)*beta_p
sigma_p_total <- sqrt( beta_p^2* market_sigma^2 + sum((w*sigma)^2))
```

```
beta_prime2 <- sigma_p_total/ market_sigma
R_prime_2 <- R_f + (market_means- R_f)*beta_prime2


###### net selectivity
R_p - R_prime_2
```

3. Decompose the overall evaluation using Fama's decomposition (net selectivity and diversification) for the single index model when short sales are not allowed. Please show this decomposition on the plot expected return against beta.

```
##             [,1]
## [1,] 0.01287708
```

```
###### diversification
R_prime_2 - R_p_prime
```

```
##        X.GSPC
## -4.194244e-06
```

```
points_table<- cbind( c( 1, market_means), c(beta_p, R_p), c(beta_p, R_p_prime), c(beta_prime2, R_prime_
colnames(points_table) <- c("M", "P", "P'", "P''")
```

The net selectivity part is 0.01287708. The diversification part is nearly 0.

```
plot( 0,0, xlim =c(0, 1), ylim= c(0, 0.02), type = "n", xlab="beta", ylab="Return")
points(points_table[1 ,], points_table[2, ], col=1:4, pch=19 )
text(points_table[1 ,], points_table[2, ] , labels = colnames(points_table), cex=0.7, col=1:4, pos=c(3,3
abline(a = R_f, b = market_means-R_f)
abline(v=0, col="red")
```