Homework 5

Joyce Yu Cahoon

7.7

Show that the solution to 7.18 is given by 7.17. As a further generalization, what is the solution to the following problem: minimizing with respect the symmetric matrix $\boldsymbol{S}$:

$$||\hat{\boldsymbol{\Sigma}}_\lambda - \boldsymbol{S}||_F^2 \quad \text{such that} \quad \lambda_{\min}(\boldsymbol{S}) \geq \delta$$

We can show the solution to 7.18 is given by 7.17 since:

$$\begin{aligned}
||\hat{\boldsymbol{\Sigma}}_\lambda - \boldsymbol{S}||_F^2 &= tr[(\hat{\boldsymbol{\Sigma}}_\lambda - \boldsymbol{S})(\hat{\boldsymbol{\Sigma}}_\lambda - \boldsymbol{S})^T] \\
&= tr(\hat{\boldsymbol{\Sigma}}\hat{\boldsymbol{\Sigma}}^T) - tr(\hat{\boldsymbol{\Sigma}}\boldsymbol{S}^T) - tr(\boldsymbol{S}\hat{\boldsymbol{\Sigma}}^T) + tr(\boldsymbol{S}\boldsymbol{S}^T) \\
&= tr(\hat{\boldsymbol{\Sigma}}\hat{\boldsymbol{\Sigma}}^T) - 2tr(\hat{\boldsymbol{\Sigma}}\boldsymbol{S}^T) + tr(\boldsymbol{S}\boldsymbol{S}^T)
\end{aligned}$$

Taking the derivation of this relative to $\hat{\boldsymbol{\Sigma}}$ results in $2\hat{\boldsymbol{\Sigma}} - 2\boldsymbol{S} = 0$ which leads to 7.17 in which:

$$\hat{\boldsymbol{\Sigma}}^+ = \boldsymbol{\Lambda}^T diag(\lambda_1^+, \ldots, \lambda_p^+)\boldsymbol{\Lambda}$$

7.10

Let $\hat{\boldsymbol{\Sigma}}$ be an estimated volatility matrix of a true volatility. Show that for any portfolio allocation $\boldsymbol{w}$, the relative estimation error is bounded by:

$$|\frac{\boldsymbol{w}^T\hat{\boldsymbol{\Sigma}}\boldsymbol{w}}{\boldsymbol{w}^T\boldsymbol{\Sigma}\boldsymbol{w}} - 1| \leq ||\boldsymbol{\Sigma}^{-1/2}\hat{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}^{-1/2} - \boldsymbol{I}_p||$$

As proven in 7.8:

$$tr[\hat{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}^{-1} - \boldsymbol{I}_p]^2 = ||\boldsymbol{\Sigma}^{-1/2}\hat{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}^{-1/2} - \boldsymbol{I}_p||_F^2$$

Thus:

$$tr[\hat{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}^{-1} - \boldsymbol{I}_p] = ||\boldsymbol{\Sigma}^{-1/2}\hat{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}^{-1/2} - \boldsymbol{I}_p||_F$$

We can then simplify:

$$\begin{aligned}
|\frac{\boldsymbol{w}^T\hat{\boldsymbol{\Sigma}}\boldsymbol{w}}{\boldsymbol{w}^T\boldsymbol{\Sigma}\boldsymbol{w}} - 1| &= |\frac{\boldsymbol{w}^T(\hat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma})\boldsymbol{w}}{\boldsymbol{w}^T\boldsymbol{\Sigma}\boldsymbol{w}}| \\
&= ||\frac{\hat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma}}{\boldsymbol{\Sigma}}|| \\
&= ||\hat{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}^{-1} - \boldsymbol{I}_p|| \\
&\leq ||\hat{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}^{-1} - \boldsymbol{I}_p||_F \\
&\leq tr[\hat{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}^{-1} - \boldsymbol{I}_p] \\
&= ||\boldsymbol{\Sigma}^{-1/2}\hat{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}^{-1/2} - \boldsymbol{I}_p||_F
\end{aligned}$$

7.11

Suppose that we have 100 investable stocks, labeled as 1 through 100 and classified as "Consumer Non-durables", "Consumer durables", "Manufacturing", "Energy", "Business equipment", "Telecommunications", "Shops", "Health", "Utilities", and "Others". Let $w_1, \ldots, w_{100}$ be the portfolio weights. If the first 10 stocks are labeled as "Consumer Non-durables", the second 10 stock are in "Consumer durables", etc, write down the constraints of the portfolio.

1. the "health stocks" are no more than 15% and "energy stocks" are no more than 30%.
2. no exposure to Telecommunications.
3. exposure to "Consumer durables" but gross exposure is zero.

7.12

Let the study period be Jan 2001 to Jan 2015. Apply the sample covariance matrix, the FF 3 factor model, and the RiskMetrics with $\lambda = .94$ to obtain the time varying covariance matrix for Dell, Ford, GE, IBM, Intel, J&J, Merck, 3-mo Tres, and S&P500 at the beginning of each month. Optimize the portfolio and holds for the next 21 days. Compute the risk of such a portfolio and compare it with the equally weighted portfolio.

First, we get the data we need:

```r
start <- as.Date("2001-01-01")
end <- as.Date("2015-01-01")
assets <- c(# "DVMT" # DELL is DROPPED since not provided by Quantmod
  "F",
  "GE",
  "IBM",
  "INTC",
  "JNJ",
  "MRK",
  "SPY")
getSymbols(assets, from = start, to = end)
# get the 3 mo tresury data
getSymbols("DGS3MO", src = "FRED")
# combine all info
closing.prices <- merge.xts(DGS3MO,
                            F[, 4],
                            GE[, 4],
                            IBM[, 4],
                            INTC[, 4],
                            JNJ[, 4],
                            MRK[, 4],
                            SPY[, 4])
# filter out to only dates of interest
data <- closing.prices["2001-01-01/2015-01-01"]
# save
saveRDS(data, "~/workspace/st790-financial-stats/hw5/covdata.rds")
```

Now we can reorganize the data within our desired time frame:

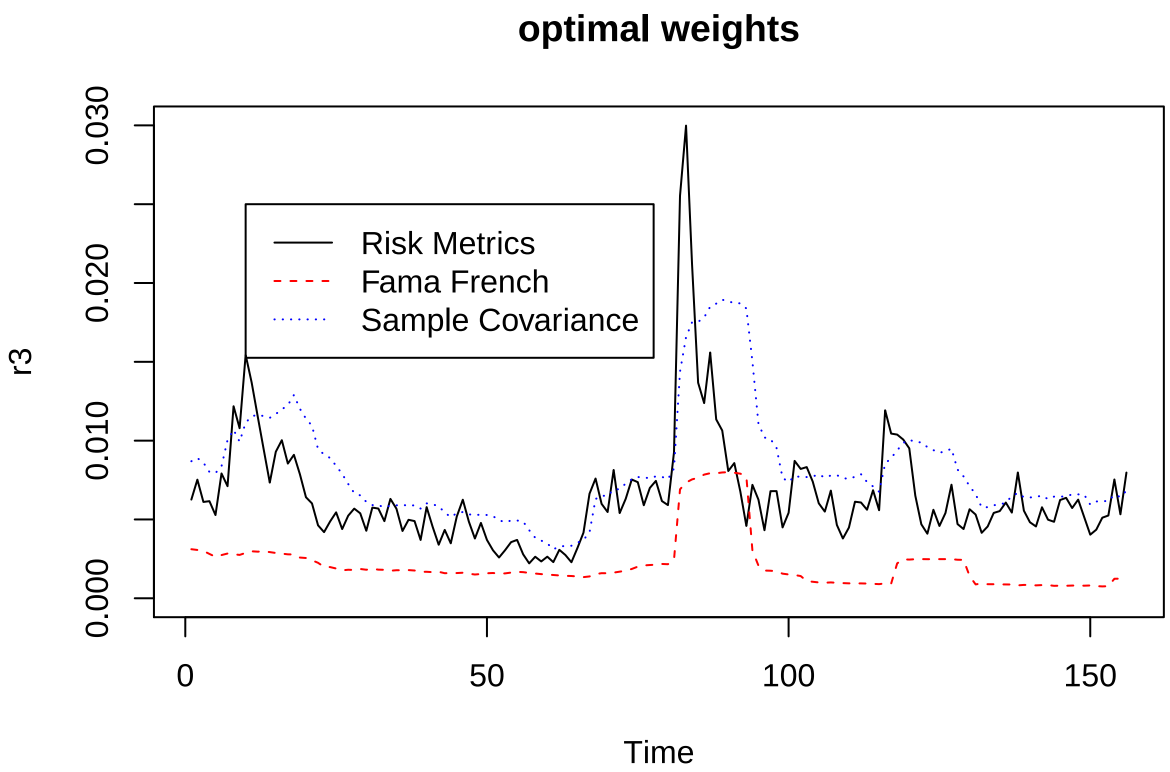
```
data <- readRDS("~/workspace/st790-financial-stats/hw5/covdata.rds")

# break it down monthly
by_month <- data.frame()
for(i in 1:ncol(data)){
  temp <- data[, i]
  monthly <- monthlyReturn(temp)
  by_month <- cbind(by_month, monthly)
  colnames(by_month)[i] <- strsplit(names(temp), "[.]")[[1]][1]
}
by_month <- na.omit(by_month)
# get the Fama French data
FF <- matrix(scan("~/workspace/st790-financial-stats/hw5/F-F_Research_Data_Factors_daily.txt", s
# match dates
D <- time(by_month)
D <- paste0(substr(D, 1, 4),
            substr(D, 6, 7),
            substr(D, 9, 10))
ind <- rep(0, length(D))
for(i in 1:length(ind)){
  ind[i] = (1:dim(FF)[1])[D[i] == FF[,1]]
}
ff <- FF[ind, ]
# break it down daily
dat <- na.omit(data)
by_day <- data.frame()
for(i in 1:ncol(dat)){
  temp <- dat[, i]
  daily <- dailyReturn(temp)
  by_day <- cbind(by_day, daily)
  colnames(by_day)[i] <- strsplit(names(temp), "[.]")[[1]][1]
}
by_day <- na.omit(by_day)
# get the FF by day
days <- time(by_day)
days <- paste0(substr(days, 1, 4),
            substr(days, 6, 7),
            substr(days, 9, 10))
ind2 <- rep(0, length(days))
for(i in 1:length(ind2)){
  ind2[i] = (1:dim(FF)[1])[days[i] == FF[,1]]
}
ff_days <- FF[ind2, ]
saveRDS(by_day, "~/workspace/st790-financial-stats/hw5/byday.rds")
saveRDS(ff_days, "~/workspace/st790-financial-stats/hw5/ff_byday.rds")
saveRDS(by_month, "~/workspace/st790-financial-stats/hw5/bymonth.rds")
saveRDS(ff, "~/workspace/st790-financial-stats/hw5/ff_bymonth.rds")
```

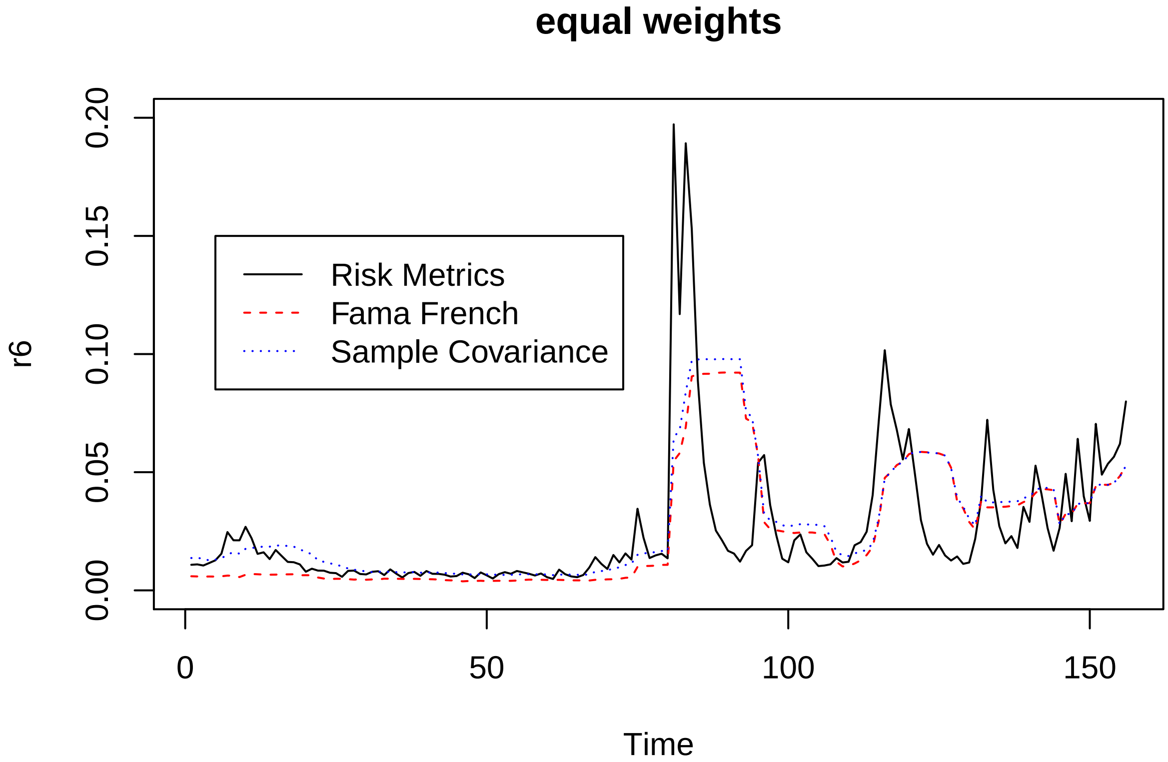
With our data in the desired format, we can now calculate our rolling covariances:

```
by_day <- readRDS("~/workspace/st790-financial-stats/hw5/byday.rds")
ff <- readRDS("~/workspace/st790-financial-stats/hw5/ff_byday.rds")
by_day[is.infinite(by_day)] <- 0
by_month <- readRDS("~/workspace/st790-financial-stats/hw5/bymonth.rds")
# risk metrics
n <- nrow(by_day)
lambda <- .94 # for daily
Sigma <- array(0, c(8, 8, n+1))
for(i in 1:n){
  tmp <- as.numeric(c(as.numeric(by_day[i, ])))
  Sigma[,,i+1] <- lambda*Sigma[,,i] + (1-lambda)*tmp %*% t(tmp)
}
# only need optimize monthly
dates_of_interest <- time(by_month)
# start window in 2002
dates_of_interest <- dates_of_interest[12:length(dates_of_interest)]
r1 <- r2 <- r3 <- r4 <- r5 <- r6 <- numeric(length(dates_of_interest))
equal_weight <- rep(1, 8)/8
for(i in 1:length(dates_of_interest)){
  temp_date <- dates_of_interest[i]
  # find the correct index to get 252 rolling window
  temp_D <- paste0(substr(temp_date, 1, 4),
            substr(temp_date, 6, 7),
            substr(temp_date, 9, 10))
  temp_i <- (1:dim(ff)[1])[temp_D == ff[,1]]
  X <- ff[(temp_i - 252):temp_i, 2:4]
  Y <- by_day[(temp_i - 252):temp_i, ]
  # method 1: sample covariance
  SigmaS <- cov(Y)
  # method 2: fama french
  resid <- resid(lsfit(X, Y))
  N <- dim(Y)[1] # length of window
  SigmaE <- t(resid) %*% resid / (N-4)
  # method 3: risk metrics
  SigmaR <- Sigma[,,temp_i]
  # optimize
  result1 <- optimalPortfolio(SigmaS, control = list(type = "minvol"))
  result2 <- optimalPortfolio(SigmaE, control = list(type = "minvol"))
  result3 <- optimalPortfolio(SigmaR, control = list(type = "minvol"))
  # compute risk
  r1[i] <- sqrt(result1 %*% SigmaS %*% result1)
  r2[i] <- sqrt(result2 %*% SigmaE %*% result2)
  r3[i] <- sqrt(result3 %*% SigmaR %*% result3)
  # compare to equal weight
  r4[i] <- sqrt(equal_weight %*% SigmaS %*% equal_weight)
  r5[i] <- sqrt(equal_weight %*% SigmaE %*% equal_weight)
```

```
  r6[i] <- sqrt(equal_weight %*% SigmaR %*% equal_weight)
}
# viz
plot(r3, type = "l", ylim = c(0, .03), xlab = "Time", ylabe = "Risk",
     lty = 1)
lines(r2, col = "red", lty =2 )
lines(r1, col = "blue", lty=3)
legend(10, .025, legend = c("Risk Metrics",
                            "Fama French",
                            "Sample Covariance"),
       col=c("black", "red", "blue"),
       lty=1:3)
title("optimal weights")
```




```
# viz
plot(r6, type = "l", ylim = c(0, .2), xlab = "Time", ylabe = "Risk",
     lty = 1)
lines(r5, col = "red", lty =2 )
lines(r4, col = "blue", lty=3)
legend(5, .15, legend = c("Risk Metrics",
                            "Fama French",
                            "Sample Covariance"),
       col=c("black", "red", "blue"),
       lty=1:3)
title("equal weights")
```



equal weights
r6
0.00
0.05
0.10
0.15
0.20
0
50
100
150
Time
Risk Metrics
Fama French
Sample Covariance