STAT 631 Homework 5

Jack Cunningham (jgavc@tamu.edu)

10/9/24

```r
load("HW05.Rdata")
```

1)

a)

In the case with short selling there are explicit solutions for the minimum variance portfolio. These are:

$$w_{\text{min.v}} = \frac{\Sigma^{-1}1}{1^T\Sigma^{-1}1}, \text{Mean} = \frac{\mu^T\Sigma^{-1}1}{1^T\Sigma^{-1}1}, \text{Variance} = \frac{1}{1^T\Sigma^{-1}1}$$

First we find the mean vector μ and covariance matrix Σ.

```r
y.mu <- apply(y,MARGIN = 2, FUN = mean)
y.mu
```

```
     AMZN        KO       NKE       PFE      TSLA       UNH       URI         V
0.4206254 0.1692590 0.2216723 0.1501095 0.6957931 0.4128102 0.5011812 0.3970618
```

```r
y.S <- var(y)
y.S
```

```
          AMZN       KO       NKE      PFE      TSLA       UNH       URI
AMZN 17.892637 2.431855  5.576128 2.564539 11.972386  2.977977  8.972939
KO    2.431855 6.238607  3.820411 2.972533  4.151542  3.972146  5.440775
NKE   5.576128 3.820411 14.638695 2.812862  9.371497  4.268385 10.709588
PFE   2.564539 2.972533  2.812862 8.787569  3.860339  4.452199  5.824990
TSLA 11.972386 4.151542  9.371497 3.860339 57.338308  6.231686 16.666210
UNH   2.977977 3.972146  4.268385 4.452199  6.231686 12.173168  7.763183
```

```
URI   8.972939 5.440775 10.709588 5.824990 16.666210  7.763183 39.914885
V     5.837551 3.829269  5.747779 3.145227  7.124705  4.292491  8.992063
            V
AMZN 5.837551
KO   3.829269
NKE  5.747779
PFE  3.145227
TSLA 7.124705
UNH  4.292491
URI  8.992063
V    9.207900
```

Now we can compute the weights.

```r
one_vector <- rep(1, 8)
y.S_inv <- solve(y.S)
w_min.v <- y.S_inv%*%one_vector/as.numeric((t(one_vector)%*%y.S_inv%*%one_vector))
w_min.v
```

```
            [,1]
AMZN  0.11244506
KO    0.42307306
NKE   0.08199088
PFE   0.26278105
TSLA -0.01238153
UNH   0.07393439
URI  -0.07294053
V     0.13109762
```

The expected return of this portfolio is:

```r
expected_return <- as.numeric(t(y.mu)%*%y.S_inv%*%one_vector)/
  as.numeric(t(one_vector)%*%y.S_inv%*%one_vector)
expected_return
```

```
[1] 0.2139305
```

The risk of this portfolio is:

```r
risk <- sqrt(1/as.numeric(t(one_vector)%*%y.S_inv%*%one_vector))
risk
```

```
[1] 2.086775
```

b)

The tangency portfolio also has an explicit solution with short selling allowed. With $\mu_{ex} = \mu - \mu_f 1$:

$$w_T = \frac{\Sigma^{-1}\mu_{ex}}{1^T\Sigma^{-1}\mu_{ex}}$$

The annual risk-free rate is 4.37%. The weekly risk-free rate is 4.37%/52. The allocation weights of the tangency portfolio are below:

```r
weekly_risk_free <- .0437/52
mu_excess = y.mu - weekly_risk_free
w_tangency = y.S_inv %*% mu_excess/
  as.numeric(one_vector^T %*% y.S_inv %*% mu_excess)
w_tangency
```

```
             [,1]
AMZN  0.200277036
KO   -0.151625522
NKE  -0.184406419
PFE  -0.151336852
TSLA  0.118670461
UNH   0.520386933
URI   0.008547379
V     0.639486985
```

The mean and variance of the tangency portfolio have the explicit solutions:

$$\text{Mean} = \frac{\mu^T\Sigma^{-1}\mu_{ex}}{1^T\Sigma^{-1}\mu_{ex}}, \text{Variance} = \frac{\mu_{ex}^T\Sigma^{-1}\mu_{ex}}{(1^T\Sigma^{-1}\mu_{ex})^2}$$

So the expected return of the tangency portfolio is:

```
expected_return_tangency <- as.numeric((t(y.mu) %*% y.S_inv %*% mu_excess)/
  (t(one_vector)%*% y.S_inv %*% mu_excess))
expected_return_tangency
```

```
[1] 0.5505735
```

And the risk of the tangency portfolio is:

```
risk_tangency <- sqrt((t(mu_excess)%*%y.S_inv%*%mu_excess)/
  (t(one_vector)%*%y.S_inv%*%mu_excess)^2)
risk_tangency
```

```
         [,1]
[1,] 3.351737
```

c)

To plot the efficient frontier requires us to compute the risk for various expected returns. The explicit solution is below:

$$\text{Amat} = [\mu, 1], \text{H} = \text{Amat}^T \Sigma^{-1} \text{Amat} = \begin{bmatrix} C & B \\ B & A \end{bmatrix}, \Delta = \det(\text{H})$$

$$\text{Risk}_{opt} = \sqrt{\frac{Am^2 - 2Bm + C}{\Delta}}$$

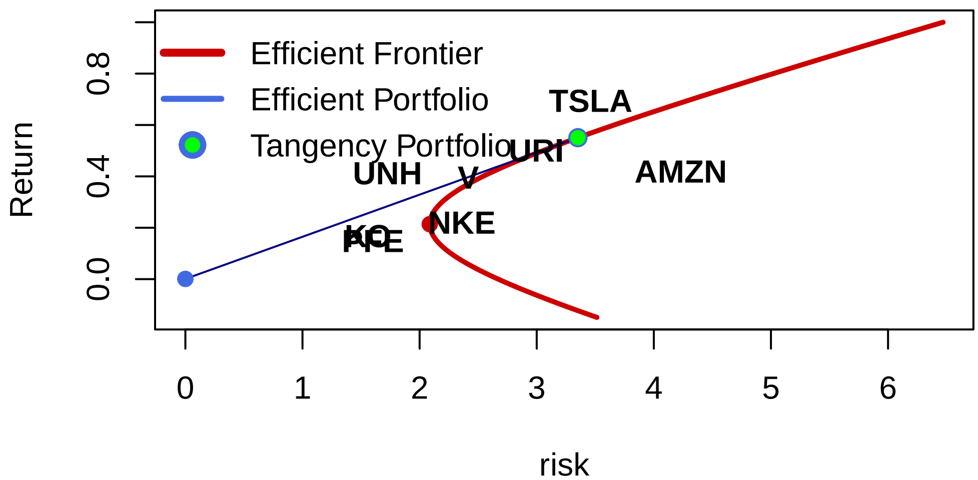
```
m.R = seq(-.15,1,0.001)
Amat = cbind(y.mu, one_vector)
H = t(Amat)%*%y.S_inv%*%Amat
A = H[2,2];B = H[1,2];C = H[1,1]; Delta = det(H)
sd.R = sqrt((A*m.R^2 - 2*B*m.R + C)/Delta)
```

Our plot is below:

```
mu.min = -.15
plot(sd.R,m.R, type = "l", xlim= c(0, max(sd.R)), xlab = "risk", ylab = "Return")
lines(sd.R[m.R > mu.min],m.R[m.R > mu.min], lwd = 2.5, col = "red3")
lines(c(0,risk_tangency), c(weekly_risk_free, expected_return_tangency), col = "navy")
points(0, weekly_risk_free, pch = 19, col = "royalblue")
points(risk_tangency,expected_return_tangency, pch = 21, col = "royalblue", bg = "green", ce
```

```r
points(risk, expected_return, pch = 19, col = "red3")
for(i in 1:8){
  text(sqrt(y.S[i]), y.mu[i], syb[i], font = 2)
}

legend("topleft",c("Efficient Frontier", "Efficient Portfolio", "Tangency Portfolio"), lty =
                   y.intersp = 1.2, bty = "n", xjust = 5)
```



d)

To find the efficient portfolio with risk of 2.5% we use the fact that:

$$\sigma_p = w_r \sigma_r$$

where σ_p is the risk of the portfolio, w_r is the weight on the tangency portfolio and σ_r is the risk of the tangency portfolio.

So then:

$$w_r = \frac{\sigma_p}{\sigma_r}$$

Where σ_p is the allowed risk of 2.5% and σ_r is the risk of the tangency portfolio is 3.351737%. So the weight is:

```r
w_r = 2.5/risk_tangency
w_r
```

```
          [,1]
[1,] 0.7458818
```

Now with the weight we use the fact that the return of the portfolio is:

$$\mu_p = w_r \mu_r + (1 - w_r)\mu_{\text{rf}}$$

where w_r is the weight on the tangency portfolio, μ_r is the return of the tangency portfolio and μ_{rf} is the risk free return. So the return of the efficient portfolio with allowed risk 2.5% is:

```r
mu_p = w_r*expected_return_tangency + (1 - w_r)*weekly_risk_free
mu_p
```

```
          [,1]
[1,] 0.4108763
```

e)

To find the efficient portfolio of a target return 0.55% we reverse the steps of part d. First we find the weight that corresponds to this return through:

$$\mu_p = w_r \mu_r + (1 - w_r)\mu_{\text{rf}}$$

$$w_r = \frac{\mu_p - \mu_{\text{rf}}}{\mu_r - \mu_{\text{rf}}}$$

So the weight on the tangency portfolio is:

```r
w_r_e <- (.55 - weekly_risk_free)/(expected_return_tangency - weekly_risk_free)
w_r_e
```

```
[1] 0.9989568
```

So the risk of this portfolio can be computed through:

$$\sigma_p = w_r \sigma_r$$

```
risk_e <- w_r_e*risk_tangency
risk_e
```

```
         [,1]
[1,] 3.348241
```

f)

We go through the same procedure as part e to start:

$$w_r = \frac{\mu_p - \mu_{\text{rf}}}{\mu_r - \mu_{\text{rf}}}$$

```
w_r_f = (.85 - weekly_risk_free)/(expected_return_tangency - weekly_risk_free)
w_r_f
```

```
[1] 1.544676
```

We find that in order to get this target return of 0.85% we would need a weight over 1 in the tangency portfolio. This is not feasible as we would need to take out a loan to make up the difference. Therefore we look to the efficient frontier to find a portfolio of risky assets with the desired return. We can find the portfolio by using the two constraints:

$$\text{Amat} = \begin{bmatrix} \mu & 1 \end{bmatrix}, \text{bvec} = \begin{bmatrix} 0.85\% \\ 1 \end{bmatrix}$$

```
library(quadprog)
Amat = cbind(y.mu, one_vector)
bvec = c(.85,1)
zeros = rep(0,8)
out = solve.QP(Dmat = y.S, dvec = zeros, Amat = Amat, bvec = bvec, meq = 2)
w_f = out$solution; names(w_f) = syb
cat("Portfolio:"); w_f
```

```
Portfolio:
```

```
      AMZN          KO         NKE         PFE        TSLA         UNH 
0.27839904 -0.66279023 -0.42135306 -0.51967333  0.23523444  0.91748341 
       URI           V 
0.08102667  1.09167306 
```

```r
sd.R[which(m.R == .85)]
```

```
[1] 5.377229
```