FERM 504 Project I

Selim Ünal

December 25, 2020

1 Tools Used for Project

For the construction of efficient set, finding of portfolio returns and efficient frontier, I will use R instead of Excel. I will pass along all code I use with comments. While Excel is a great tool for analysis of one dimensional data, I believe it lacks necessary tools for proper analysis of multi dimensional data.

2 Preparation of Input Data

Following code prepares necessary R libraries and gets monthly closure data from Yahoo for portfolio tickers.

```
library(quantmod)
library(ggplot2)
library(IntroCompFinR)

#Start date of portfolio data
from.dat <- as.Date("01/01/10", format="%m/%d/%y")

#End date of portfolio data
to.dat <- as.Date("12/02/20", format="%m/%d/%y")

# Tickers for portfolio data
tickers=c("QQQ","VOE", "VEA", "VWO", "USRT",
 "SHY", "IGIB", "EMB", "GLD","SPLV")

# Get data and store them in separate xts objects
for (x in tickers)
  getSymbols(x,periodicity = "monthly", src="yahoo",
  from = from.dat, to = to.dat)
```

As SPLV data starts from May 2011, I add NA values to SPLV data to make all data same length and then put all data into single length data frame.

```
splv=c(rep(NA,16),Cl(SPLV))
data <- cbind(Cl(QQQ),Cl(VOE), Cl(VEA), Cl(VWO), Cl(USRT),
        Cl(SHY), Cl(IGIB), Cl(EMB), Cl(GLD),splv)
```

Now that we have all data, let's calculate mean return and standard deviation and keep them in separate vectors. Important note here is that annualized return and standard deviation does not work properly in my code so I work around without annualizing security returns but I use monthly risk free rate instead of annual rate.

```
data_mean=c()
data_sd=c()

# Prepare a new data frame for returns with one less row
# as first row will be NA when return is calculated
new_data=data[1:131,]

for (x in 1:9){
  vec=as.vector(data[,x])
  diff_vec=diff(vec)
  diff_vec <- diff_vec[!is.na(diff_vec)]
  vec=diff_vec/vec[1:length(diff_vec)]
  data_mean=c(data_mean,mean(vec))
  data_sd=c(data_sd,sd(vec))
  new_data[,x]=vec
}

# I have to calculate SPLV separately as it is
# not of same length
splv=as.vector(Cl(SPLV))
splv=diff(splv)/splv[1:115]
data_mean=c(data_mean,mean(splv))
data_sd=c(data_sd, sd(splv))
splv=c(rep(0,16),splv)
new_data[,"splv"]=splv
```

3 Markowitz Optimization

First, we need to calculate covariance matrix of the portfolio snd calculate portfolio return and risk for an equal weighted portfolio of 10 assets.

```
#As SPLV data starts from May 2011, we need to calculate
# covariance matrix on data from May 2011 to Dec 2020
cov_matrix=cov(new_data[17:131,])
```

```
# Prepare an equal weight portfolio
weight=rep(1/10,10)

# Portfolio return can be calculated as
# scalar multiplication of weight and return vectors
return_portfolio=sum(weight*data_mean)
# Portfolio return is 6.40% per annum(0.53% monthly) for equal weight

# Portfolio standard deviation can be calculated
# as matrix multiplication of covariance and weight vectors
# followed by scalar multiplication of weight vector
# and result vector in the first step
sd_portfolio=sqrt(sum((cov_matrix %*% weight) * weight))

# Portfolio risk is 9.58% per annum(2.77% monthly) for equal weight
```

Second, we calculate portfolio with minimum risk. For that we use globalMin.portfolio function of R package IntroCompFinR. We assume no shorting is present for the construction of portfolio.

```
gmin.port = globalMin.portfolio(data_mean, cov_matrix,
shorts = FALSE)
```

Global minimum portfolio has, on annual basis, return of 0.54% (0.04% on monthly basis) and risk of 0.76% (0.22% on monthly basis)with weights 2.35% of VOE and 97.65% SHY. The other securities will have zero weight.

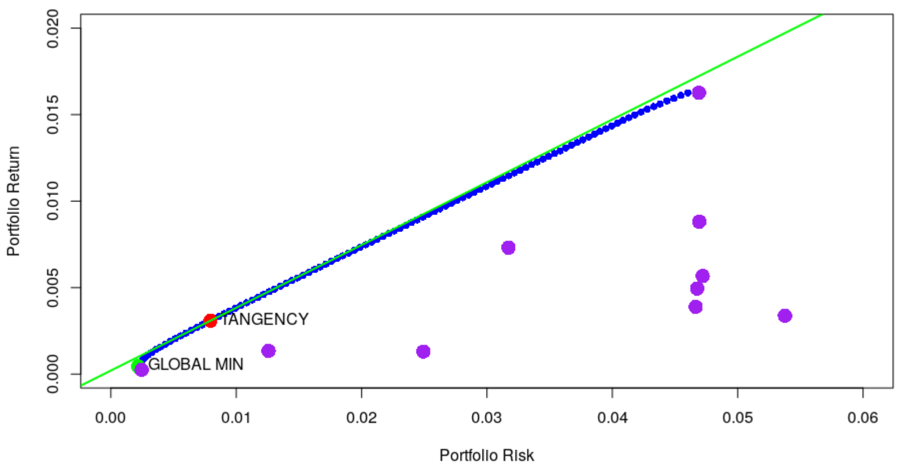
4 Plotting Efficient Frontier

Following code calculates the efficient frontier, tangent portfolio and produces the figure below.

```
# Change risk free rate to monthly basis
r.free=0.0025/12

tan.port <- tangency.portfolio(data_mean, cov_matrix,
        r.free, shorts = FALSE)

gmin.port = globalMin.portfolio(data_mean,
          cov_matrix,shorts = FALSE)

ef <- efficient.frontier(data_mean, cov_matrix,
        nport = 100, alpha.min = 0,
                alpha.max = 1, shorts = FALSE)
```

```
plot(ef$sd, ef$er, col="blue", pch=16,
                xlim=c(0,0.12),ylim=c(0,0.02),
                xlab="Portfolio Risk",ylab="Portfolio Return")
points(gmin.port$sd, gmin.port$er, col="green", pch=16, cex=2)
points(tan.port$sd, tan.port$er, col="red", pch=16, cex=2)
points(data_sd,data_mean,col="purple", pch=16, cex=2)
text(gmin.port$sd, gmin.port$er, labels="GLOBAL MIN", pos=4)
text(tan.port$sd, tan.port$er, labels="TANGENCY", pos=4)
sr.tan = (tan.port$er - r.free)/tan.port$sd
abline(a=r.free, b=sr.tan, col="green", lwd=2)
```

Crucial point is that return of global minimum portfolio should be above risk free rate of return, otherwise we will have trouble constructing efficient tangent portfolio. In the figure, global minimum portfolio has been denoted as green circle and each tickers with purple circle. Blue line denotes efficient frontier and green line denotes tangent line to the eficient frontier from risk free rate.



Sharpe ratio is the slope of tangency line found by difference of tangency portfolio return and risk free rate divided by tangency portfolio risk. In this case, we have Sharpe ratio as 0.36(1.26 annualized) for tangent portfolio and 0.18(0.64 annualized) for equal weighted portfolio. Please note that these have been calculated with monthly return and risk. To make these figures annualized, we have to multiply with square root of 12.

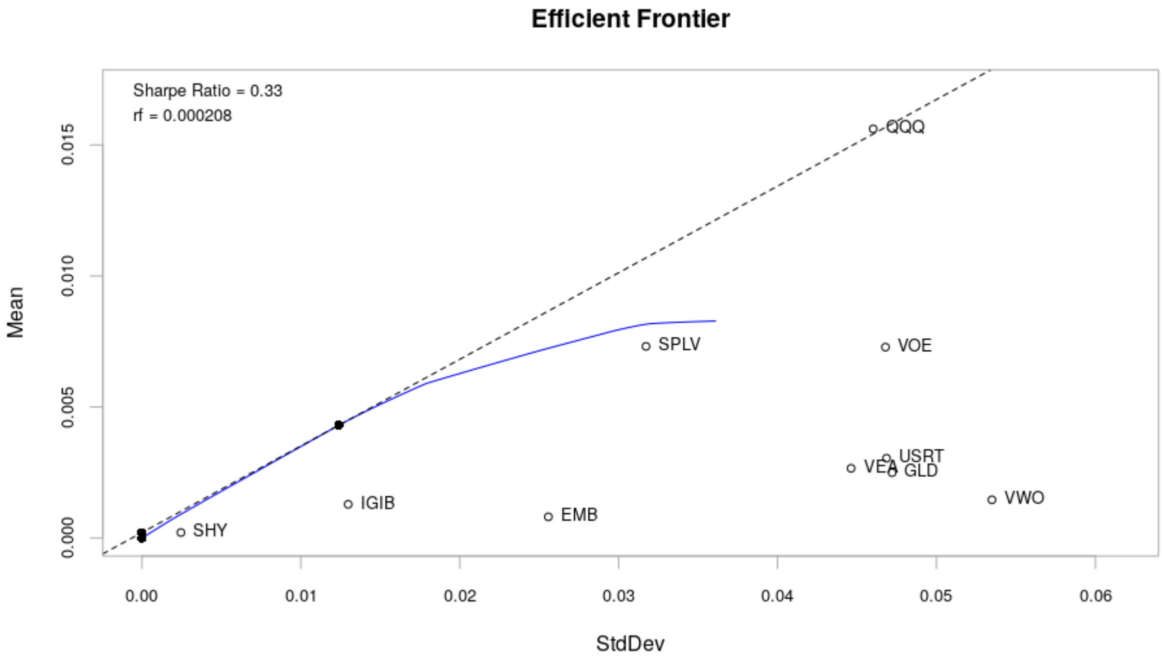
Tangent portfolio has 3.73% return and 2.77% risk on an annual basis. Tangent portfolio has 17% QQQ, 81% SHY, 1% GLD and 1% SPLV.

5 Reality Check

Now that we have finished our preliminary analysis, we need more advanced tools to analyze further constraints. I will use following R libraries in the following section.

```
library(PortfolioAnalytics)
library(DEoptim)
library(ROI)
require(ROI.plugin.quadprog)
require(ROI.plugin.glpk)

# Prepare data
R <- new_data[17:131,]
# change the column names for better legends in plotting
colnames(R) <- tickers
funds <- colnames(R)

# initial portfolio object
init <- portfolio.spec(assets=funds)
# initial constraints
init <- add.constraint(portfolio=init, type="weight_sum", min_sum=0, max_
init <- add.constraint(portfolio=init, type="box", min=0.0,max=0.25)
init <- add.constraint(portfolio=init, type="long-only")

#Construct efficient frontier and chart it
meanvar.ef <- create.EfficientFrontier(R=R, portfolio=init,
        type="mean-StdDev", n.portfolios=250)
chart.EfficientFrontier(meanvar.ef, match.col="StdDev",
                    type="l", RAR.text="Sharpe Ratio", pch=4,
                    rf=r.free,col="blue")
```

Constraints are that all securities will be long, having between zero and 25% weight and total weight will be at most 100%. Subject to these constraints our efficient frontier will be as follows.



Efficient frontier is below and further right than our previous frontier without constraint.Following code will help us identify tangent portfolio.

```
means <- meanvar.ef$frontier[,c("mean")] * 100
stdevs <- meanvar.ef$frontier[,c("StdDev")] * 100
sharpeRatios <- (means - r.free*100) / stdevs
tangencyPortfolio <- meanvar.ef$frontier
        [sharpeRatios == max(sharpeRatios),]
```

In this new setting, we have Sharpe ratio of 0.33 in our tangent portfolio.

Tangent portfolio has 5.17% return and 4.30% risk on an annual basis. Tangent portfolio has 25% QQQ, 25% SHY, 0.5% GLD and 4.6% SPLV.

If our customer is not comfortable with more than 10% volatility, then the we will look for the portfolio in the efficient frontier with at most 10% volatility. This annual constraint is calculated as 2.89% on monthly basis. Portfolios on the blue line with standard deviation below 2.89% may be selected by the customer depending on his/her indifference curve.