Diversification Strategy to Mitigate Risk

NEERAJ BHATT

Harrisburg University of Science & Technology

ANLY 515: Risk Modelling & Assessment

**Introduction:**

The obvious purpose of investing for anyone is to make money. Investment Bankers and Traders spend their entire time trying to predict what the market is likely to be in future so as to capitalize on the opportunities that exist in future. One of the major concerns with investment is risk. As the saying goes “No risk no returns”. However, knowing the risk is more than just knowing how much the stock is going to vary using the standard deviation. While standard deviation does capture the volatility of an asset there does exist possibilities of rare events that can cause an investor to lose a lot of money. These measures are Value at Risk and Expected Shortfall that will be covered in this study. These measures tell you the downside of an asset returns that can be used to access the riskiness of an asset.

One of the best ways to mitigate risks are diversification. This involves investing in a class of different assets. For example investing in different stocks from different industries could mitigate the exposure of volatility in a single stock or even within an industry. The current pandemic is a prime example of that where the airline industry has been badly hit due to lack of travel whereas the retail industry and the tech stocks did reasonably well during the pandemic. So the risk of the airline industry could be mitigated by the tech industry and retail industry stocks. However, diversification does not just apply to stocks from different industries but also among different assets. If a portion of the portfolio is focused on currencies and commodities. Commodities have seen shorter fall in the pandemic so adding them to the portfolio could further reduce the risk. The focus of this study will be on diversification and specifically comparing diversification within an asset to diversification between assets.

**Purpose:**

It is extremely important to have a portfolio containing different asset classes it is also important to note that having too many assets is also not a wise strategy. Over diversification can sometimes expose you to more risk from multiple industries and even reduce your upside. Also, assets within the same class especially stocks tend to be correlated, which increases the risk. It can also make tracking each asset difficult. So the purpose of our study is to find the best possible diversification in the portfolio to minimize risks and address our hypothesis.

**Hypothesis:**

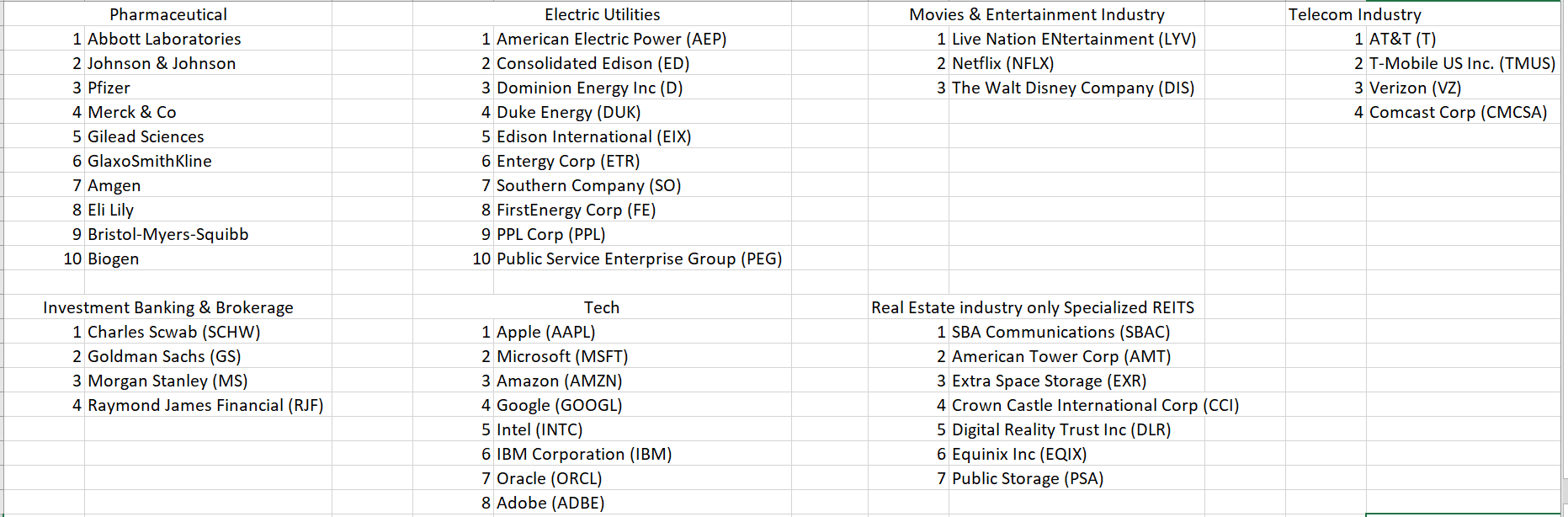
A portfolio across multiple different asset classes is likely to perform better than the portfolio within a single asset asset class but over diversification into too many assets is not risk averse.

**Methodology:**

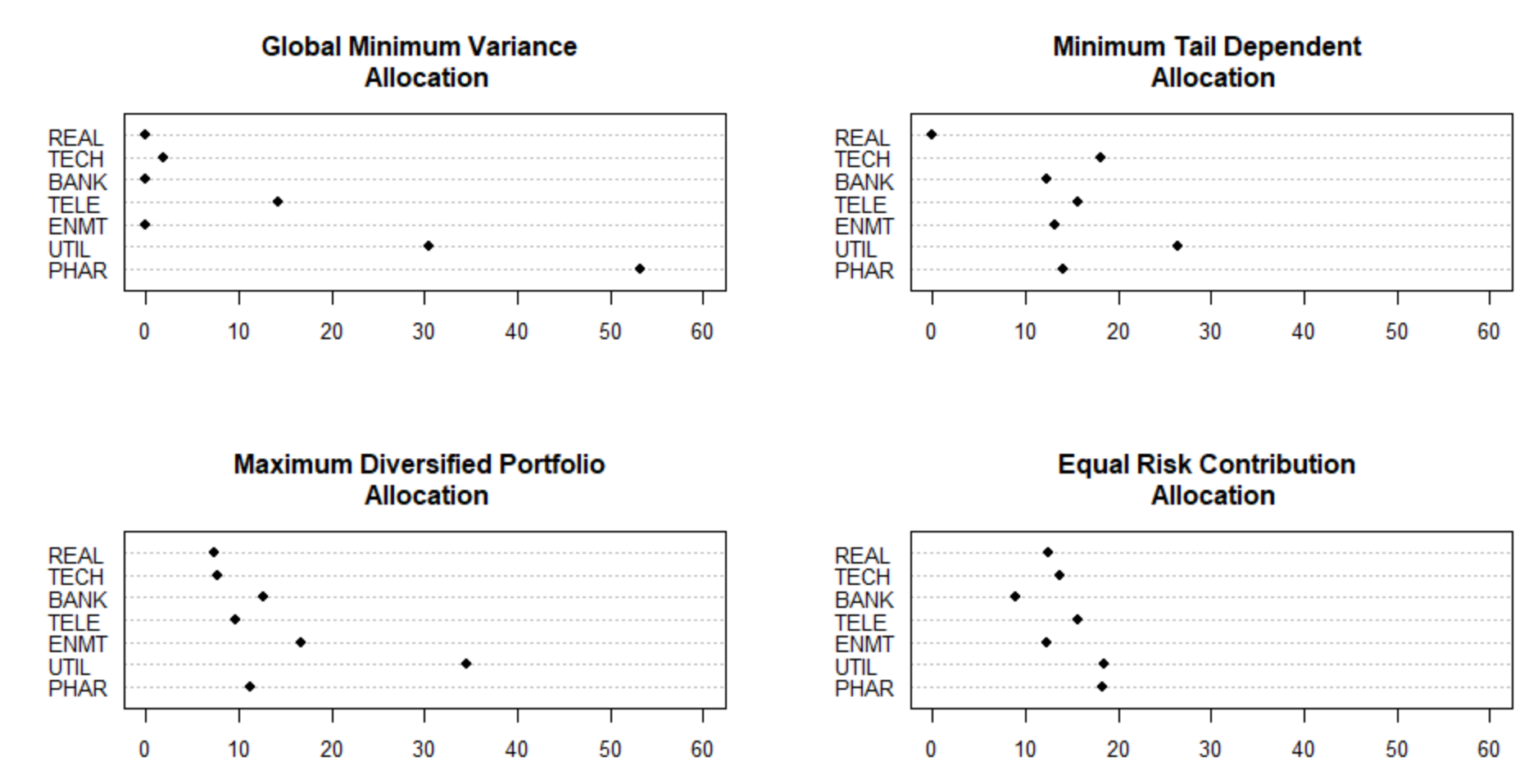
The different assets to be considered are stocks, currencies & commodities. First, an industry analysis would be done by computing index averages of stocks from multiple different industries and comparing their performance to SP500 using different portfolios. Stocks will be picked from the industries with high weights in the most optimal portfolio. The three currencies to be used will be Yen, Euro, and the British Pounds. Finally, three commodities selected are base metals, gold, and crude oil. A portfolio will be constructed with these assets and expected shortfall and value at risk will be used to evaluate them. Garch and Garch Copula models will be used predict the volatility of the assets individually and after modelling their dependence to compute expected shortfall at 95% level. Since different assets are listed differently instead of daily returns monthly returns will be used.

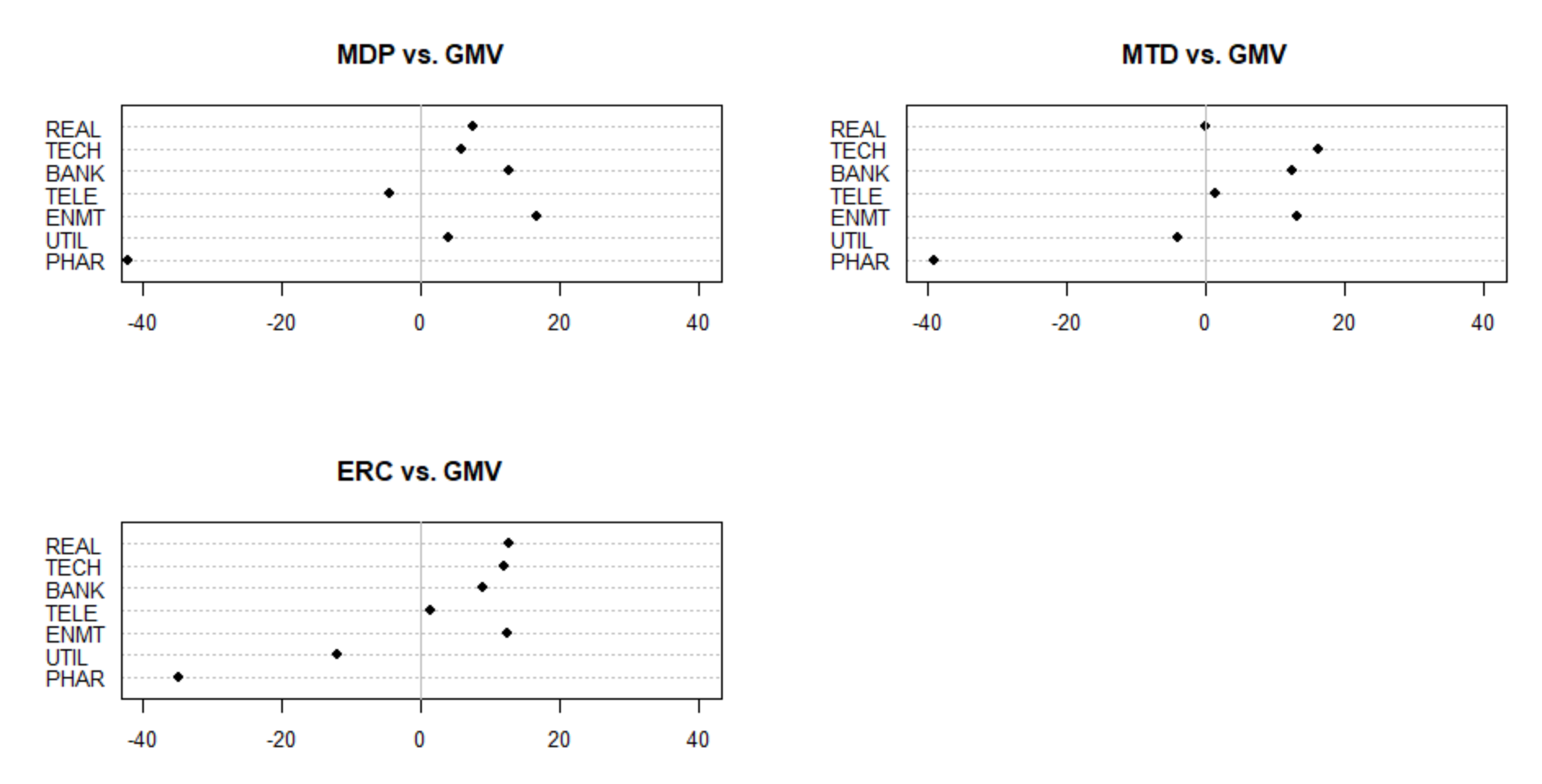
**Industry Analysis:**

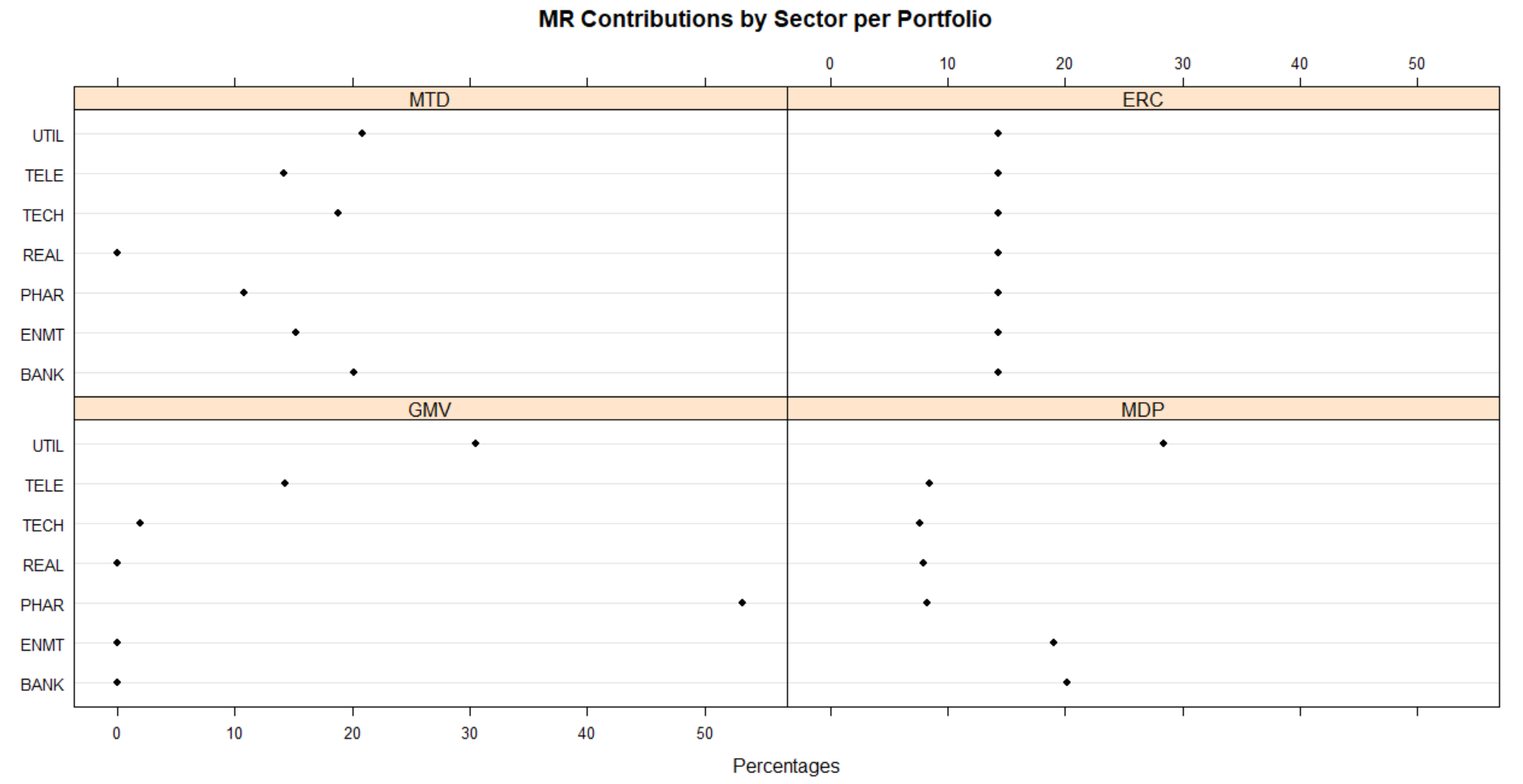
Pharmaceutical, Real Estate, Finance & Banking, Technology, Electric Utilities, Movies & Entertainment, and Telecommunications. The following are the companies taken in each industry.



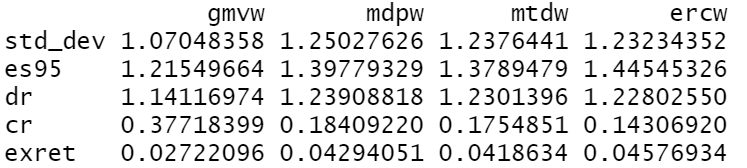
The market capitalization of each company was used for calculating the industrial index. The data was collected from y-charts1. The period of data is from 2008-01-01 to 2020-10-06. This period was taken to ensure that both the recessions in 2008 housing crisis and the current pandemic crisis are captured. For each day, the market caps were summed. The market cap of the very first day was divided by a divisor that made sure the first price is 1000 and then the same divisor was used for all other days. This normalizes the price for each industry making comparisons possible. The same procedure was followed for the SP500 data, which was collected from Yahoo Finance2. Four portfolios were constructed using these indices Global Minimum Variance Portfolio (GMVP), Most Diversified Portfolio (MDP), Minimum Tail Dependent Portfolio (MTDP), and Equal risk contributed portfolio (ERC). The allocation of weights can be seen below.







We can see that in GMVP most of the weight was given to pharma while in MDP & MTDP most allocation was provided to electric utilities. ERC as expected as about equal allocation of weights. The allocations in Pharma drop significantly in other portfolios. There is considerable difference between the sector allocations across the portfolios. As expected pharma industry has the most risk associated in the GMVP at over 50%. Expected Shortfall, Standard Deviation, diversification ratio, concentration ratio and expected returns for each portfolio are given below.



Not surprisingly GMVP has the lowest standard deviation whereas MDP has the highest diversification ratio and ERC has the lowest concentration ratio. Expected shortfall is the highest for ERC whereas expected returns are the highest for the MDP. The data would be split into 80% train and 20% test. Below are the beta estimates for the training data to understand co-movement with the market.



Utilities have the lowest co-movement with the market whereas banking have the highest co-movement with the market. Let’s look at Tau, which are the Kendall rank correlation estimates. Pharma, utility, and telecom appear to have the lowest co-movement with the market.



Interestingly tech had a lower than beta but the highest correlation with the market.

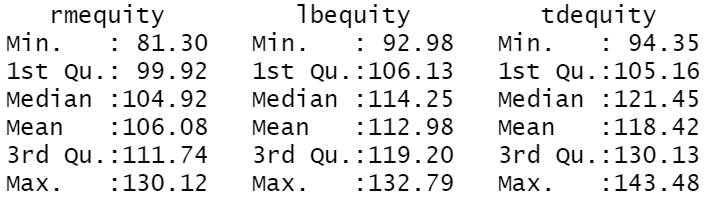
Based on co-movement beta we get the following sectors with the following weights.



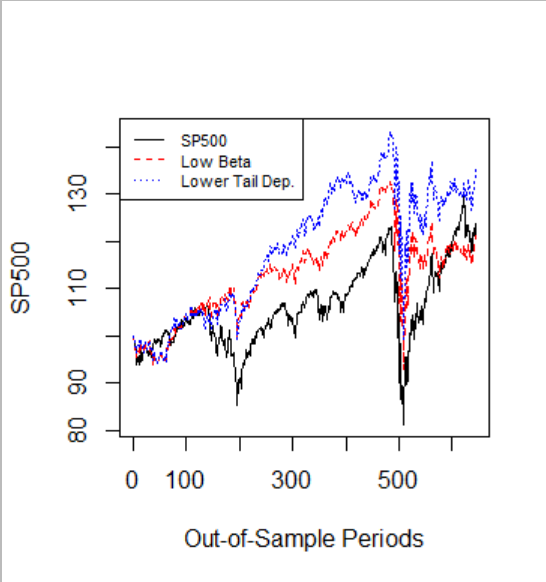
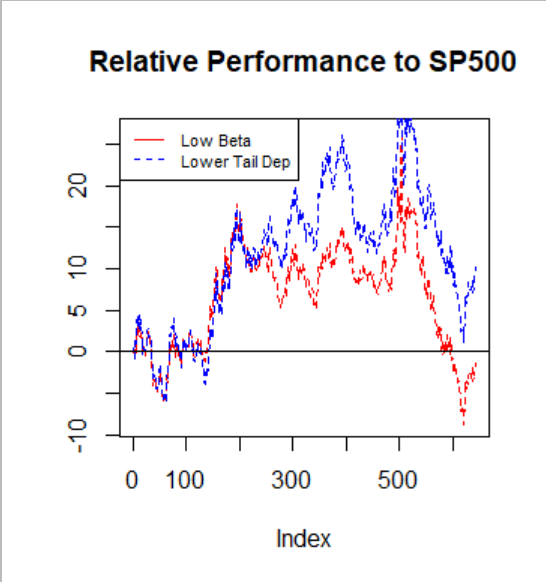
Based on lambda, a measure of interdependence between each sector and the market at the lower tail we get:



We can see that Telecom and utility are part of both the portfolios. These weights will be used to test on out of sample test data to access which portfolio would provide higher returns for lower risk.



On average and median tail dependent portfolio returns are significantly higher than low beta portfolio and the market.

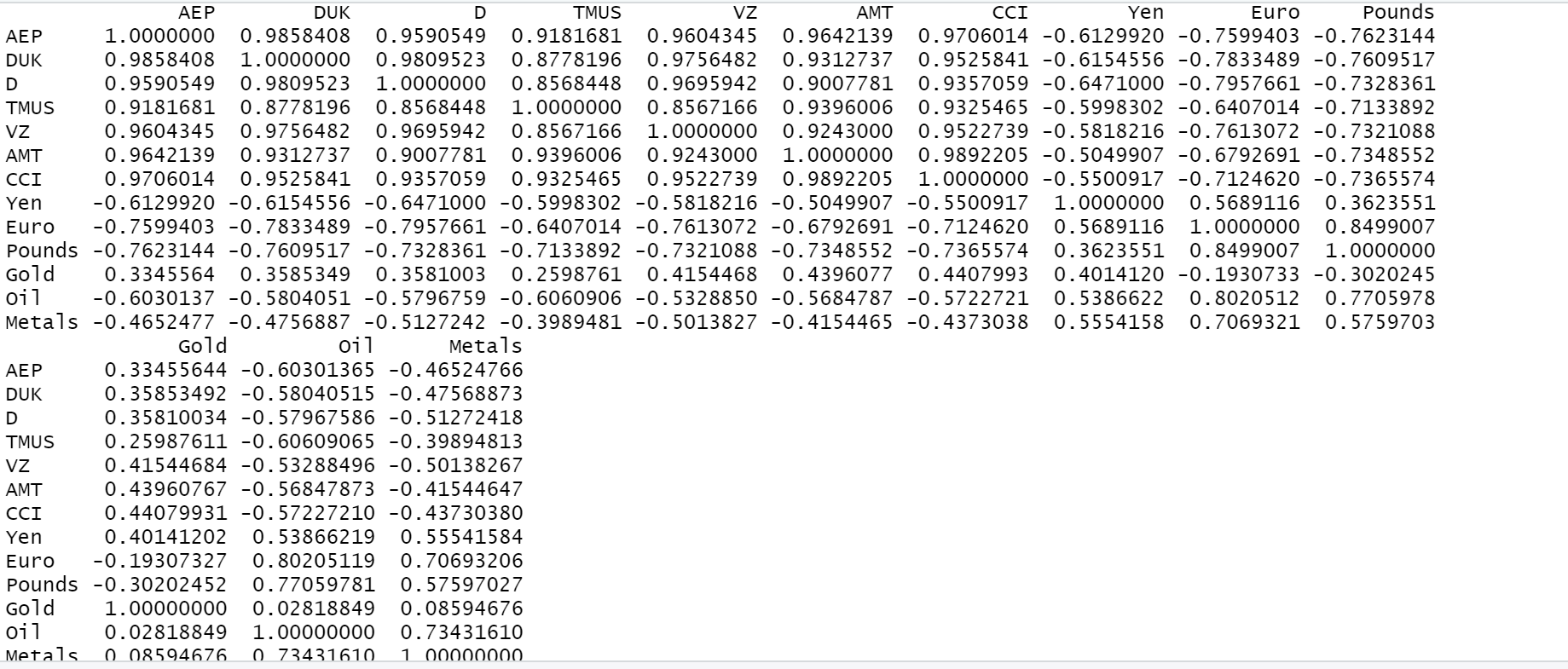
We can see that both portfolios consistently outperform the market with low tail dependent portfolio significantly outperforming the low beta portfolio. Thus we would pick the low tail dependent portfolio for out stocks. Thus we would pick stocks from electric utilities, Telecommunications & Real Estate Industry with specialized reits.

**Portfolio:**

We plan to have 13 assets in our portfolio out of which there will be 7 stocks, 3 currencies, 3 commodities. We will pick 3 stocks from the electric utility industry, 2 each from telecom and real estate to keep the approx. 46, 25, 25 balance the low tail dependent portfolio gave us. The stocks chosen are American Electric Power (AEP), Duke Energy (DUK), Dominion Energy (D), T-Mobile (TMUS), Verizon (VZ), American Tower Corp (AMT), Crown Castle International Corp (CCI). The assets would be used to create two portfolios, the global minimum variance portfolio (GMVP) and Most Diversified Portfolio (MDP).

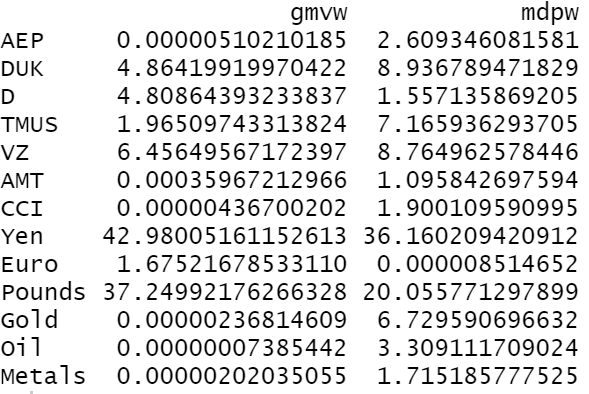
**Analysis:**

The first part of the analysis was to examine correlation between assets.



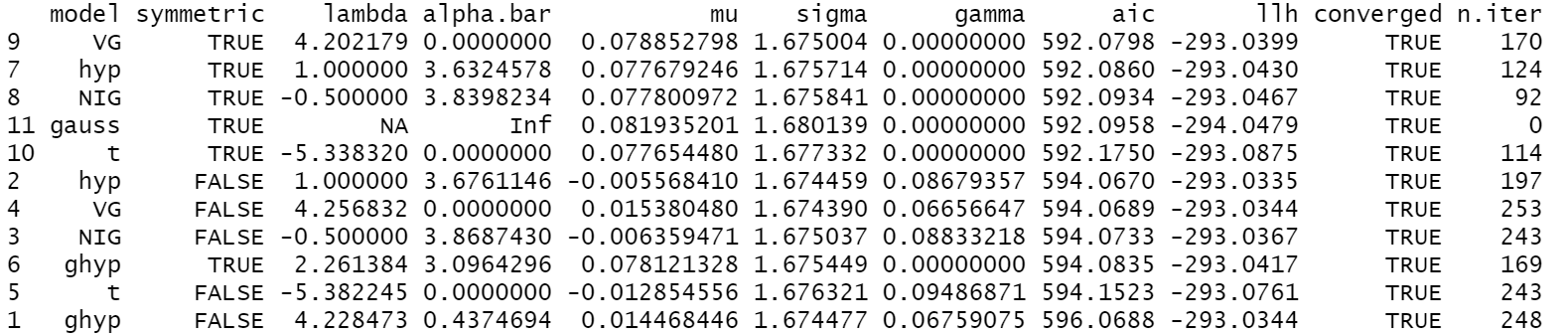
There exists significant correlation across multiple assets as we can see American electric power is highly correlated with Duke Energy and there also exists high correlation between T-Mobile and Verizon. Interestingly Gold, Oil and Metals have the lowest correlation with all assets. However, Base Metals and Oil are highly correlated. This clearly tells us that garch Copula. Oil has a very high correlation with Pounds and Euros. This makes sense as Oil prices do significantly impact exchange rates3.

The weights for both GMVP and MDP are provided below.



We can see that in both cases Yen got allocated he highest amount of weight followed by pounds. Japanese Yen and the British Pounds are two of the topmost traded currencies and thus this makes sense4.

The expected returns for the GMVP portfolio is about 0.08% whereas for the MDP portfolio is about 0.33%.

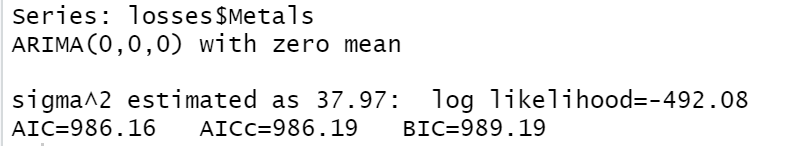


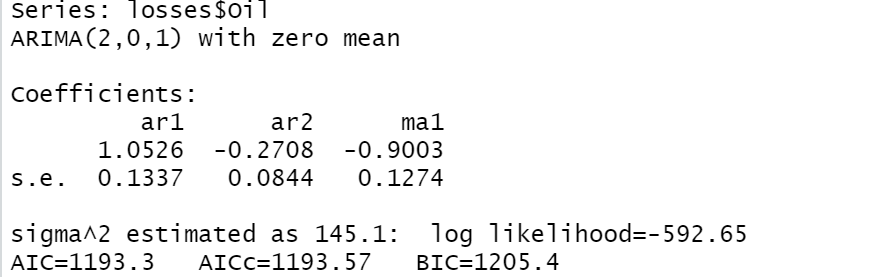
The Variance Gamma distribution with symmetry is the best distribution for GMVP as it has the lowest AIC value. Similarly for MDP too the optimal distribution is Variance Gamma with symmetry.

The largest expected daily loss at 95% level for GMVP is 2.67% whereas for MDP is 3.09%. Expected shortfall at 95% level for GMVP is about 3.52% and for MDP is 3.86%.

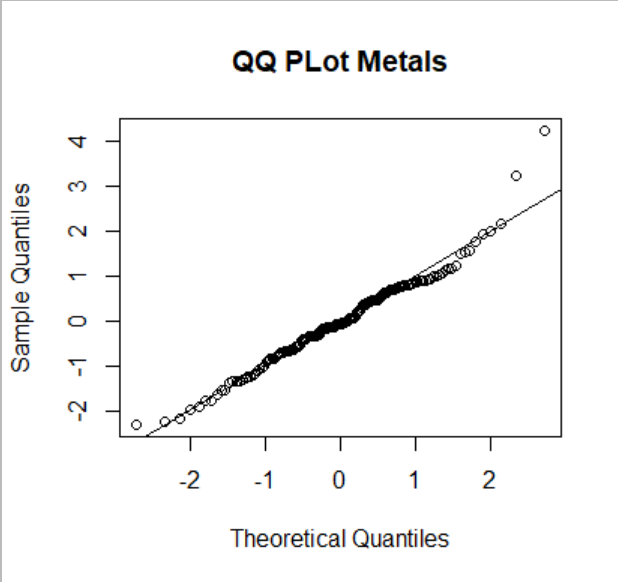
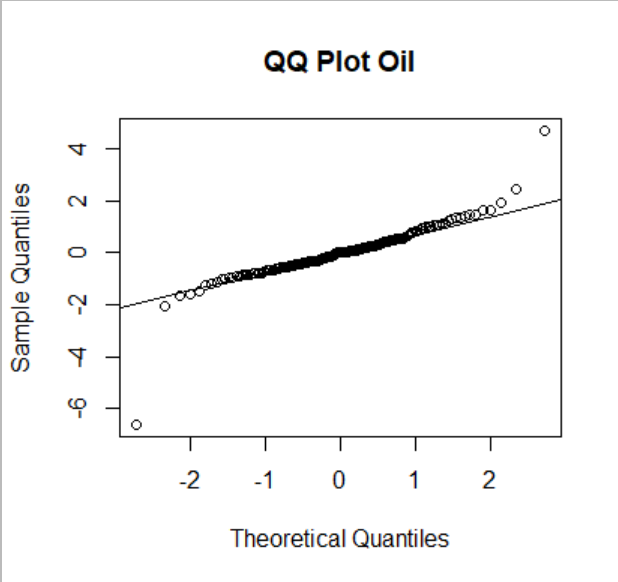
The next step is to predict the volatility using arima and garch model.

Below are the results of arima models for metals and oil.

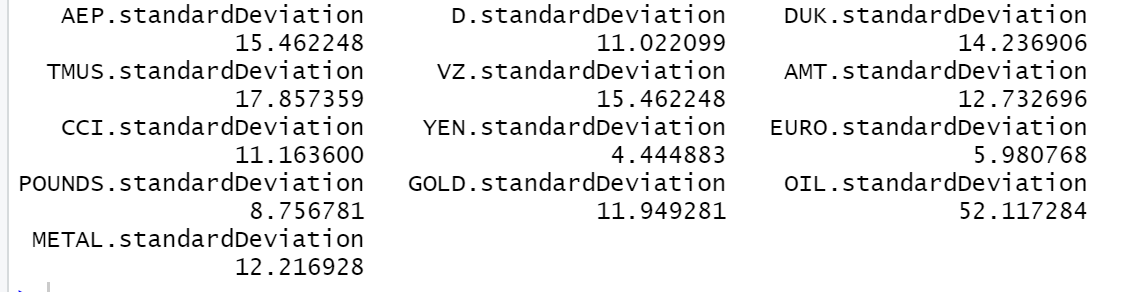




Similarly all 13 assets were fitted with different models based on auto.arima

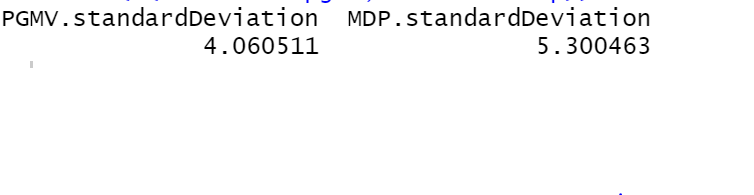


The expected shortfall for each asset at 95% level is stated below.



The individual asset expected shortfall is extremely high.

Let’s look at the garch expected shortfall of the two portfolios.



The expected shortfall for the portfolios are significantly lower than the individual ones. PGMV has a lower expected shortfall than MDP.

**Conclusion:**

We saw that having a portfolio of multiple different assets can significantly minimize risks than simply having a portfolio comprising of a single asset class like stocks. We also saw, in our industry analysis that mitigating downside risk yields in a much more optimal result as we observed with the minimum tail dependent portfolio. Another important result we observed was that due to high amount of correlation between assets the optimal global minimum variance portfolio picked a significantly higher weight for Yen and British pounds and very small percentage in some stocks thus proving our hypothesis that over diversifying into too many assets are not likely to mitigate risks. Instead concentrating in a few important assets is a smarter strategy.

**Appendix:**

**Industry data preparation script:**

# setting the working directory

setwd('C:/Users/bneer/OneDrive/Desktop/Harrisburg-Risk Modelling/Final Project')

# getting the required packages

library(QRM)

library(qrmdata)

library(PerformanceAnalytics)

library(xts)

library(quantmod)

library(FRAPO)

library(timeSeries)

library(fBasics)

library(ghyp)

library(fExtremes)

library(ismev)

library(evir)

library(tseries)

library(forecast)

library(tidyverse)

options(scipen = 999)

# getting the data for four asset classes: abbots, bonds, commodities, currencies

# abbots: Comparing multiple industries

# Healthcare companies

# 1. abbot Laboratories (Health Care Equipment)

abbot <- as.xts(data.frame(getSymbols('ABT', src = 'yahoo',

from = '2008-01-01', to = '2020-10-06',

auto.assign = F)))

abbot\_df <- as.data.frame(abbot)

abbot\_df <- cbind(Date = index(abbot), data.frame(abbot\_df, row.names = NULL))

head(abbot\_df)

tail(abbot\_df)

plot.xts(abbot$ABT.Adjusted)

# Market\_cap

abbot\_cap <- read.csv('Pharma companies/ABT\_data.csv')

abbot\_cap$Period <- as.Date(abbot\_cap$Period, format = "%Y-%m-%d %H:%M:%S")

head(abbot\_cap,2)

abbot\_cap <- abbot\_cap %>%

map\_df(rev)

colnames(abbot\_cap)[1] <- "Date"

abbot\_cap[,2] <- abbot\_cap[,2]/1000

head(abbot\_cap,2)

abbot\_df$Date <- as.Date(abbot\_df$Date)

abbot <- inner\_join(abbot\_df, abbot\_cap, by = "Date")

abbot <- abbot[,c(1,7,8)]

head(abbot,2)

#--------------------------------------------------------------------------------

from = "2008-01-01"

to = "2020-10-06"

get\_data <- function(ticker, path, from, to, df\_xts = F){

stock\_xts <- as.xts(data.frame(getSymbols(ticker, src = 'yahoo',

from = from, to = to,

auto.assign = F)))

stock\_df <- as.data.frame(stock\_xts)

stock\_df <- cbind(Date = index(stock\_xts), data.frame(stock\_df, row.names = NULL))

stock\_df$Date <- as.Date(stock\_df$Date)

stock\_cap <- read.csv(paste(path, '/', ticker,"\_data.csv", sep = ""))

stock\_cap$Period <- as.Date(stock\_cap$Period)

stock\_cap <- stock\_cap %>%

map\_df(rev)

colnames(stock\_cap)[1] <- "Date"

#stock\_cap[,2] <- stock\_cap[,2]/1000

stock <- inner\_join(stock\_df, stock\_cap, by = "Date")

stock <- stock[,c(1,7,8)]

stock\_ts <- timeSeries(stock[,2:ncol(stock)], charvec = as.character(stock$Date))

if(df\_xts){

return(stock\_xts)

}else{

return(stock\_ts)

}

}

pharma\_path <- 'Pharma companies'

# Pharma Industries

# 1) Abbott Laboratories

abbot\_ts <- get\_data("ABT", path = pharma\_path, from = from, to = to)

head(abbot\_ts,2)

# 2) Johnson & Johnson

jnj\_ts <- get\_data("JNJ", path = pharma\_path, from = from, to = to)

head(jnj\_ts, 2)

# 3) Pfizer

pfizer\_ts <- get\_data("PFE", path = pharma\_path, from = from, to = to)

head(pfizer\_ts, 2)

# 4) Merck & Co

merck\_ts <- get\_data("MRK", path = pharma\_path, from = from, to = to)

head(merck\_ts, 2)

# 5) Gilead Sciences

gilead\_ts <- get\_data("GILD", path = pharma\_path, from = from, to = to)

head(gilead\_ts, 2)

# 6) GlaxoSmithKline

gsk\_ts <- get\_data("GSK", path = pharma\_path, from = from, to = to)

head(gsk\_ts, 2)

# 7) Amgen

amgen\_ts <- get\_data("AMGN", path = pharma\_path, from = from, to = to)

head(amgen\_ts, 2)

# 8) Eli Lily

lily\_ts <- get\_data("LLY", path = pharma\_path, from = from, to = to)

head(lily\_ts, 2)

# 9) Bristol-Myers Squibb

myers\_ts <- get\_data("BMY", path = pharma\_path, from = from, to = to)

head(myers\_ts, 2)

# 10) Biogen

biogen\_ts <- get\_data("BIIB", path = pharma\_path, from = from, to = to)

head(biogen\_ts, 2)

# Gathering market caps

date = as.character(abbot\_df$Date)

pharma\_market <- data.frame(abbot\_cap = abbot\_ts$Abbott.Laboratories.Market.Cap,

jnj\_cap = jnj\_ts$Johnson...Johnson.Market.Cap,

pfizer\_cap = pfizer\_ts$Pfizer.Inc.Market.Cap,

merck\_cap = merck\_ts$Merck...Co.Inc.Market.Cap,

gilead\_cap = gilead\_ts$Gilead.Sciences.Inc.Market.Cap,

gsk\_cap = gsk\_ts$GlaxoSmithKline.PLC.Market.Cap,

amgen\_cap = amgen\_ts$Amgen.Inc.Market.Cap,

lily\_cap = lily\_ts$Eli.Lilly.and.Co.Market.Cap,

myers\_cap = myers\_ts$Bristol.Myers.Squibb.Co.Market.Cap,

biogen\_cap = biogen\_ts$Biogen.Inc.Market.Cap)

dim(pharma\_market)

head(pharma\_market)

pharma\_market$total <- rowSums(pharma\_market)

head(pharma\_market)

# lets keep the divisor by using the first day in such a way that we get a value

# of 1000.

divisor <- pharma\_market\_ts$total[1]/1000

divisor

pharma\_market$PHAR <- round(pharma\_market$total/divisor, 2)

head(pharma\_market)

pharma\_market\_ts <- timeSeries(pharma\_market, charvec = date)

head(pharma\_market\_ts)

# writing a function to compute the industry indices for any industry

get\_index <- function(industry\_abbrev, vector\_ticker, path, from, to){

stock\_xts <- as.xts(data.frame(getSymbols(vector\_ticker[1], src = 'yahoo',

from = from, to = to,

auto.assign = F)))

stock\_df <- as.data.frame(stock\_xts)

stock\_df <- cbind(Date = index(stock\_xts), data.frame(stock\_df, row.names = NULL))

stock\_df$Date <- as.Date(stock\_df$Date)

date <- as.character(stock\_df$Date)

ind\_cap <- data.frame(matrix(0, nrow = length(date),

ncol = length(vector\_ticker)))

for(i in 1:length(vector\_ticker)){

ind\_market <- get\_data(vector\_ticker[i], path = path, from = from, to = to)

ind\_cap[,i] <- ind\_market[,2]

}

names(ind\_cap) <- paste0(vector\_ticker, "\_cap")

ind\_cap$total <- rowSums(ind\_cap)

divisor <- ind\_cap$total[1]/1000

ind\_cap[,industry\_abbrev] <- round(ind\_cap$total/divisor, 2)

return(ind\_cap)

}

#-------------------------------------------------------------------------------

# Phara Industry

pharma\_path <- 'Pharma companies'

pharma\_index <- get\_index("PHAR", c("ABT", "JNJ", "PFE", "MRK",

"GILD", "GSK", "AMGN", "LLY",

"BMY", "BIIB"), pharma\_path, from, to)

head(pharma\_index,3)

colnames(pharma\_index)

# perfectly consistent

#-----------------------------------------------------------------------------

# Electric Utility industry

# Companies:

# 1) American Electric Power (AEP)

# 2) Consolidated Edison (ED)

# 3) Dominion Energy Inc (D)

# 4) Duke Energy (DUK)

# 5) Edison International (EIX)

# 6) Entergy Corp (ETR)

# 7) Southern Company (SO)

# 8) FirstEnergy Corp (FE)

# 9) PPL Corp (PPL)

# 10) Public Service Enterprise Group (PEG)

utility\_path <- "Utility companies"

utility\_index <- get\_index("UTIL", c("AEP", "ED", "D", "DUK",

"EIX", "ETR", "SO",

"FE", "PPL", "PEG"), utility\_path, from, to)

head(utility\_index,3)

colnames(utility\_index)

#-------------------------------------------------------------------------------

# Movies & Entertainment industry

# Companies:

# 1) Live Nation ENtertainment (LYV)

# 2) Netflix (NFLX)

# 3) The Walt Disney Company (DIS)

movies\_path <- "Movies companies"

movies\_index <- get\_index("ENMT", c("LYV", "NFLX", "DIS"),

movies\_path, from, to)

head(movies\_index,3)

colnames(movies\_index)

#--------------------------------------------------------------------------------

# Telecom industry

# Companies:

# 1) AT&t (T)

# 2) T-Mobile US Inc. (TMUS)

# 3) verizon (VZ)

# 4) Comcast Corp (CMCSA)

telecom\_path <- "Telecom companies"

telecom\_index <- get\_index("TELE", c("T", "TMUS", "VZ", "CMCSA"),

telecom\_path, from, to)

head(telecom\_index,3)

colnames(telecom\_index)

#-------------------------------------------------------------------------------

# Investment Banking & Brokerage

# Companies:

# 1) Charles Scwab (SCHW)

# 2) Goldman Sachs (GS)

# 3) Morgan Stanley (MS)

# 4) Raymond James Financial (RJF)

banking\_path <- "Finance companies"

banking\_index <- get\_index("BANK", c("SCHW", "GS", "MS", "RJF"),

banking\_path, from, to)

head(banking\_index,3)

colnames(banking\_index)

#-------------------------------------------------------------------------------

# Tech industry with the tech giants

# Companies:

# 1) Apple (AAPL)

# 2) Microsoft (MSFT)

# 3) Amazon (AMZN)

# 4) Google (GOOGL)

# 5) Intel (INTC)

# 6) IBM Corporation (IBM)

# 7) Oracle (ORCL)

# 8) Adobe (ADBE)

tech\_path <- "Tech companies"

tech\_index <- get\_index("TECH", c("AAPL", "MSFT", "AMZN", "GOOGL",

"INTC", "IBM", "ORCL", "ADBE"),

tech\_path, from, to)

head(tech\_index,3)

colnames(tech\_index)

#-------------------------------------------------------------------------------

# Real Estate industry mainly REITS Specialized Reits

# Companies:

# 1) SBA Communications (SBAC)

# 2) American Tower Corp (AMT)

# 3) Extra Space Storage (EXR)

# 4) Crown Castle International Corp (CCI)

# 5) Digital Reality Trust Inc (DLR)

# 6) Equinix Inc (EQIX)

# 7) Public Storage (PSA)

real\_path <- "Real estate companies"

real\_index <- get\_index("REAL", c("SBAC", "AMT", "EXR", "CCI",

"DLR", "EQIX", "PSA"),

real\_path, from, to)

head(real\_index,3)

colnames(real\_index)

#-------------------------------------------------------------------------------

# getting the SP500 market index

sp500\_xts <- as.xts(data.frame(getSymbols("^GSPC", src = 'yahoo',

from = from, to = to,

auto.assign = F)))

sp500\_df <- as.data.frame(sp500\_xts)

sp500\_df <- cbind(Date = index(sp500\_xts), data.frame(sp500\_df, row.names = NULL))

sp500\_df$Date <- as.Date(sp500\_df$Date)

head(sp500\_df,2)

divisor\_sp500 <- sp500\_df$GSPC.Adjusted[1]/1000

sp500\_df$SPI <- round(sp500\_df$GSPC.Adjusted/divisor\_sp500, 2)

head(sp500\_df, 3)

#-------------------------------------------------------------------------------

# industry data

industry\_df <- data.frame(SPI = sp500\_df$SPI, PHAR = pharma\_index$PHAR,

UTIL = utility\_index$UTIL, ENMT = movies\_index$ENMT,

TELE = telecom\_index$TELE, BANK = banking\_index$BANK,

TECH = tech\_index$TECH, REAL = real\_index$REAL)

industry\_ts <- timeSeries(industry\_df, charvec = date)

head(SPISECTOR,3)

head(industry\_ts, 3)

industry\_df2 <- data.frame(Date = date,

SPI = sp500\_df$SPI, PHAR = pharma\_index$PHAR,

UTIL = utility\_index$UTIL, ENMT = movies\_index$ENMT,

TELE = telecom\_index$TELE, BANK = banking\_index$BANK,

TECH = tech\_index$TECH, REAL = real\_index$REAL)

#write.csv(industry\_df2, file = "industry\_data.csv", row.names = F)

**Industry Analysis Script:**

setwd('C:/Users/bneer/OneDrive/Desktop/Harrisburg-Risk Modelling/Final Project')

library(QRM)

library(qrmdata)

library(PerformanceAnalytics)

library(xts)

library(quantmod)

library(FRAPO)

library(timeSeries)

library(fBasics)

library(ghyp)

library(fExtremes)

library(ismev)

library(evir)

library(tseries)

library(forecast)

library(tidyverse)

library(copula)

options(scipen = 999)

industry\_df <- read.csv('industry\_data.csv')

head(industry\_df)

date <- as.character(industry\_df$Date)

industry\_ts <- timeSeries(industry\_df[,-1], charvec = date)

head(industry\_ts)

indexes <- interpNA(industry\_ts[,-1], method = "linear")

head(indexes)

returns <- returnseries(indexes, method = "discrete", trim = T)

head(returns)

covar <- cov(returns)

covar

gmvw <- Weights(PGMV(returns)) # Global Minimum Variance Portfolio

mdpw <- Weights(PMD(returns)) # Most Diversified Portfolio

mtdw <- Weights(PMTD(returns)) # Minimum Tail Dependent Portfolio

ercw <- Weights(PERC(covar)) # Equal risk contributed portfolio

# combining results

w <- cbind(gmvw, mdpw, mtdw, ercw)

w

# Marginal Risk Contributions

mar\_risk\_cont <- apply(w, 2, mrc, Sigma = covar)

rownames(mar\_risk\_cont) <- colnames(indexes)

colnames(mar\_risk\_cont) <- c("GMV", "MDP", "MTD", "ERC")

mar\_risk\_cont

# plot of allocations

oldpar <- par(no.readonly = TRUE)

par(mfrow = c(2, 2))

dotchart(gmvw, xlim = c(0, 60), main = "Global Minimum Variance \n Allocation", pch = 19)

dotchart(mtdw, xlim = c(0, 60), main = "Minimum Tail Dependent \n Allocation", pch = 19)

dotchart(mdpw, xlim = c(0, 60), main = "Maximum Diversified Portfolio \n Allocation", pch = 19)

dotchart(ercw, xlim = c(0, 60), main = "Equal Risk Contribution \n Allocation", pch = 19)

# comparison of allocations

oldpar <- par(no.readonly = TRUE)

dotchart(mdpw - gmvw, xlim = c(-20, 20), main = "MDP vs. GMV",

pch = 19)

abline(v = 0, col = "grey")

dotchart(mtdw - gmvw, xlim = c(-20, 20), main = "MTD vs. GMV",

pch = 19)

abline(v = 0, col = "grey")

dotchart(ercw - gmvw, xlim = c(-20, 20), main = "ERC vs. GMV",

pch = 19)

abline(v = 0, col = "grey")

# comparing allocations between portfolios

sector <- factor(rep(rownames(mar\_risk\_cont), 4),

levels = sort(rownames(mar\_risk\_cont)))

port <- factor(rep(colnames(mar\_risk\_cont), each = 7),

levels = colnames(mar\_risk\_cont))

mrcdf <- data.frame(MRC = c(mar\_risk\_cont), port, sector)

mrcdf

dotchart(mdpw - gmvw, xlim = c(-40, 40), main = "MDP vs. GMV",

pch = 19)

abline(v = 0, col = "grey")

dotchart(mtdw - gmvw, xlim = c(-40, 40), main = "MTD vs. GMV",

pch = 19)

abline(v = 0, col = "grey")

dotchart(ercw - gmvw, xlim = c(-40, 40), main = "ERC vs. GMV",

pch = 19)

abline(v = 0, col = "grey")

par(oldpar)

dotplot(sector ~ MRC | port, groups = port, data = mrcdf,

xlab = "Percentages",

main = "MR Contributions by Sector per Portfolio",

col = "black", pch = 19)

dotplot(port ~ MRC | sector, groups = sector, data = mrcdf,

xlab = "Percentages",

main = "MR Contributions by Portfolio per Sector",

col = "black", pch = 19)

# converting returns to decimals

returns\_dec <- returns / 100

head(returns\_dec)

# computing portfolio returns for each portfolio

pret <- apply(w, 2, function(x) returns\_dec %\*% x / 100)

head(pret)

# standard deviation

std\_dev <- apply(pret, 2, sd) \* 100

std\_dev

# Expected Shortfall at 95% level

es95 <- apply(pret, 2, function(x)

abs(ES(R = x, method = "modified") \* 100))

es95

# Diversification Ratio

dr <- apply(w, 2, dr, Sigma = covar)

# Concentration Ratio

cr <- apply(w, 2, cr, Sigma = covar)

cr

# Expected Returns

exret <- apply(pret, 2, function(x) mean(x)\*100)

# combining the results

res <- rbind(std\_dev, es95, dr, cr, exret)

res

# extracting all other columns than the date columns

industry\_sub <- industry\_df[,2:ncol(industry\_df)]

head(industry\_sub)

# converting to time series

date\_ind <- as.character(industry\_df$Date)

industry\_sub\_ts <- timeSeries(industry\_sub, charvec = date\_ind)

head(industry\_sub\_ts)

# creating data for markets and assets

train\_rows <- round(0.80 \* nrow(industry\_sub\_ts), 0)

test\_rows <- round(0.20 \* nrow(industry\_sub\_ts), 0)

train\_rows + test\_rows

rm <- returnseries(industry\_sub\_ts[1:train\_rows,1], trim = T)

head(rm)

ra <- returnseries(industry\_sub\_ts[1:train\_rows,2:ncol(industry\_sub\_ts)], trim = T)

head(ra)

# beta - co-movement with the market

beta <- apply(ra, 2, function(x) cov(x, rm) / var(rm))

beta

# tau - kendall rank correlation

tau <- apply(ra, 2, function(x) cor(x, rm, method = "kendall"))

tau

# copula parameter estimates

theta <- copClayton@iTau(tau)

theta

# lambda- estimates of interdependence between each stock and the SP500 at the

# lower tail

lambda <- copClayton@lambdaL(theta)

lambda

# selecting the betas below median betas

idxBeta <- beta < median(beta)

idxBeta[idxBeta]

beta[idxBeta]

# inverse log weighted scaled portfolio weights

wBeta <- -1 \* log(abs(beta[idxBeta]))

wBeta <- wBeta / sum(wBeta) \* 100

wBeta

# selecting the lambdas below median lambda

idxTD <- lambda < median(lambda)

idxTD[idxTD]

beta[idxTD]

# inverse log weighted scaled portfolio weights

wTD <- -1 \* log(lambda[idxTD])

wTD <- wTD / sum(wTD) \* 100

wTD

# testing portfolio with these weights on out of sample data

rmo <- returnseries(industry\_sub\_ts[train\_rows:nrow(industry\_sub\_ts),1],

method = "discrete", percentage = F) + 1

head(rmo)

rao <- returnseries(industry\_sub\_ts[train\_rows:nrow(industry\_sub\_ts),

2:ncol(industry\_sub\_ts)],

method = "discrete", percentage = F) + 1

head(rao)

nrow(rm) + nrow(rmo)

# set the first observation to 1 for rmo

rmo[1] <- 100

head(rmo)

# computing the cumulative performance of the index

rmequity <- cumprod(rmo)

rmequity

# picking columns from rao that were chosen by the low beta portfolio

lbequity <- rao[, idxBeta]

head(lbequity)

# assigning the beta vector as the first row of lbequity

lbequity[1, ] <- wBeta

head(lbequity)

# cumulative performance of the low beta portfolio in the out of sample

# period

lbequity <- rowSums(apply(lbequity, 2, cumprod))

head(lbequity)

# picking columns from rao that were chosen by the low tail dependence portfolio

tdequity <- rao[, idxTD]

head(tdequity)

# assigning the beta vector as the first row of tdequity

tdequity[1, ] <- wTD

head(tdequity)

# cumulative performance of the low tail dependent portfolio in the out of sample

# period

tdequity <- rowSums(apply(tdequity, 2, cumprod))

head(tdequity)

# combining market performance, low beta portfolio performance and

# low tail dependent performance

y <- cbind(rmequity, lbequity, tdequity)

summary(y)

# time series plots for out of sample periods

par(mfrow = c(1, 1))

plot(rmequity, type = "l", ylim = range(y), ylab = "SP500",

xlab = "Out-of-Sample Periods")

lines(lbequity, lty = 2, col = "red")

lines(tdequity, lty = 3, col = "blue")

legend("topleft",

legend = c("SP500", "Low Beta", "Lower Tail Dep."),

lty = 1:3,

cex = 0.70, col = c("black", "red", "blue"))

# creating a barplot of relative performance of SP500, LOw Beta Portfolio

# and Low Tail Dependency Portfolio

relout <- rbind((lbequity / rmequity - 1) \* 100,

(tdequity / rmequity - 1) \* 100)

head(relout)

relout <- relout[, -1]

plot(relout[1,], ylab = '', type = "l", lty = 2, col = "red",

main = "Relative Performance to SP500")

lines(relout[2,], lty = 2, col = "blue")

abline(h = 0)

legend("topleft",

legend = c("Low Beta", "Lower Tail Dep"),

lty = 1:3,

cex = 0.70, col = c("red", "blue"))

**Portfolio Analysis Script:**

setwd('C:/Users/bneer/OneDrive/Desktop/Harrisburg-Risk Modelling/Final Project')

library(QRM)

library(qrmdata)

library(PerformanceAnalytics)

library(xts)

library(quantmod)

library(FRAPO)

library(timeSeries)

library(fBasics)

library(ghyp)

library(fExtremes)

library(ismev)

library(evir)

library(tseries)

library(forecast)

library(tidyverse)

library(copula)

library(ghyp)

library(fBasics)

options(scipen = 999)

# duration

from = "2008-01-01"

to = "2020-10-01"

get\_date <- function(ticker, from, to){

stock\_xts <- as.xts(data.frame(getSymbols.yahoo(ticker, src = 'yahoo',

return.class = "xts",

index.class = "Date",

from = from, to = to,

auto.assign = F,

periodicity = "monthly")))

stock\_df <- as.data.frame(stock\_xts)

stock\_df <- cbind(Date = index(stock\_xts), data.frame(stock\_df, row.names = NULL))

stock\_df$Date <- as.Date(stock\_df$Date)

date <- as.character(stock\_df$Date)

return(date)

}

data\_prep <- function(ticker, from, to){

stock\_xts <- as.xts(data.frame(getSymbols.yahoo(ticker, src = 'yahoo',

return.class = "xts",

index.class = "Date",

from = from, to = to,

auto.assign = F,

periodicity = "monthly")))

stock\_df <- as.data.frame(stock\_xts)

stock\_df <- cbind(Date = index(stock\_xts), data.frame(stock\_df, row.names = NULL))

stock\_df$Date <- as.Date(stock\_df$Date)

date <- as.character(stock\_df$Date)

stock\_df <- stock\_df[,c(1,7)]

colnames(stock\_df)[2] <- ticker

stock\_ts <- timeSeries(stock\_df[,2], charvec = date)

colnames(stock\_ts) <- ticker

return(stock\_ts)

}

curr <- function(ticker, currency\_name, from, to){

currency\_xts <- as.xts(data.frame(getSymbols.yahoo(ticker, src = 'yahoo',

return.class = "xts",

index.class = "Date",

from = from, to = to,

auto.assign = F,

periodicity = "monthly")))

currency\_df <- as.data.frame(currency\_xts)

currency\_df <- cbind(Date = index(currency\_xts), data.frame(currency\_df, row.names = NULL))

currency\_df$Date <- as.Date(currency\_df$Date)

date <- as.character(currency\_df$Date)

currency\_df <- currency\_df[,c(1,7)]

colnames(currency\_df)[2] <- currency\_name

currency\_df[,2] <- 1/currency\_df[,2]

currency\_ts <- timeSeries(currency\_df[,2], charvec = date)

colnames(currency\_ts) <- currency\_name

return(currency\_ts)

}

comm <- function(ticker, commodity\_name, from, to){

commodity\_xts <- as.xts(data.frame(getSymbols.yahoo(ticker, src = 'yahoo',

return.class = "xts",

index.class = "Date",

from = from, to = to,

auto.assign = F,

periodicity = "monthly")))

commodity\_df <- as.data.frame(commodity\_xts)

commodity\_df <- cbind(Date = index(commodity\_xts), data.frame(commodity\_df, row.names = NULL))

commodity\_df$Date <- as.Date(commodity\_df$Date)

date <- as.character(commodity\_df$Date)

commodity\_df <- commodity\_df[,c(1,7)]

colnames(commodity\_df)[2] <- commodity\_name

commodity\_ts <- timeSeries(commodity\_df[,2], charvec = date)

colnames(commodity\_ts) <- commodity\_name

return(commodity\_ts)

}

# stocks

aep <- data\_prep("AEP", from, to)

duk <- data\_prep("DUK", from, to)

d <- data\_prep("D", from, to)

t <- data\_prep("TMUS", from, to)

vz <- data\_prep("VZ", from, to)

amt <- data\_prep("AMT", from, to)

cci <- data\_prep("CCI", from, to)

# currencies

from\_curr = "2008-01-01"

to\_curr = "2020-09-01"

yen <- curr("JPY=X", "Yen", from\_curr, to\_curr)

euro <- curr("EUR=X", "Euro", from\_curr, to\_curr)

pounds <- curr("GBP=X", "Pounds", from\_curr, to\_curr)

# Commodities

gold <- comm("GC=F", "Gold", from, to)

oil <- comm("CL=F", "Oil", from, to)

base\_metals <- comm("DBB", "Metals", from, to)

# gathering all

assets <- cbind(aep, duk, d, t, vz, amt, cci, yen, euro, pounds, gold, oil,

base\_metals)

head(assets)

# computing the portfolio

# covariance

V <- cov(assets)

V

# correlation

r <- cor(assets)

r

# returns

R <- returnseries(assets, method = "discrete", trim = T)

head(R)

gmvw <- Weights(PGMV(R)) # Global Minimum Variance Portfolio

mdpw <- Weights(PMD(R)) # Most Diversified Portfolio

# combining results

w <- cbind(gmvw, mdpw)

w

# Portfolio

pret <- apply(w, 2, function(x) R %\*% x / 100)

head(pret)

# timeseries objects

# GMVP

date <- as.character(get\_date("AEP", from, to))

gmvp <- timeSeries(pret[,1], charvec = date[2:length(date)])

colnames(gmvp) <- "GMVP"

head(gmvp)

# MDP

mdp <- timeSeries(pret[,2], charvec = date[2:length(date)])

colnames(mdp) <- "MDP"

head(mdp)

# Expected Returns

apply(pret, 2, mean)

# Fitting the right distribution

aic\_gmvp <- stepAIC.ghyp(gmvp)

aic\_gmvp$fit.table

aic\_gmvp$fit.table[aic\_gmvp$fit.table$aic == min(aic\_gmvp$fit.table$aic),]

aic\_mdp <- stepAIC.ghyp(mdp)

aic\_mdp$fit.table

aic\_mdp$fit.table[aic\_mdp$fit.table$aic == min(aic\_mdp$fit.table$aic),]

# fitting the distribution

gvgfit\_gmvp <- fit.VGuv(gmvp, symmetric = T, na.rm = T)

gvgfit\_mdp <- fit.VGuv(mdp, symmetric = T, na.rm = T)

p <- c(0.01, 0.05, 0.1)

portvar\_gmvp <- abs(qghyp(p, gvgfit\_gmvp))

portvar\_mdp <- abs(qghyp(p, gvgfit\_mdp))

portes\_gmvp <- abs(ESghyp(p, gvgfit\_gmvp))

portes\_gmvp

portes\_mdp <- abs(ESghyp(p, gvgfit\_mdp))

portes\_mdp

# fitting the arima model

losses <- R\*-1

head(R)

head(losses)

auto.arima(losses$AEP) # (0,0,0)

auto.arima(losses$DUK) # (2,0,1)

auto.arima(losses$D) # (1,0,0)

auto.arima(losses$TMUS) # (0,0,0)

auto.arima(losses$VZ) # (2,0,0)

auto.arima(losses$AMT) # (0,0,0)

auto.arima(losses$CCI) # (0,0,0)

auto.arima(losses$Yen) # (0,0,0)

auto.arima(losses$Euro) # (0,0,0)

auto.arima(losses$Pounds) # (0,0,0)

auto.arima(losses$Gold) # (0,0,1)

auto.arima(losses$Oil) # (2,0,1)

auto.arima(losses$Metals) # (0,0,0)

# fitting the models

arimaaep <- arima(losses$AEP, order = c(0,0,0), include.mean = F)

std\_res\_aep <- (arimaaep$residuals - mean(arimaaep$residuals))/sd(arimaaep$residuals)

acf(arimaaep$residuals)

qqnorm(std\_res\_aep)

qqline(std\_res\_aep)

arimaoil <- arima(losses$Oil, order = c(2,0,1), include.mean = F)

std\_res\_oil <- (arimaoil$residuals - mean(arimaoil$residuals))/sd(arimaoil$residuals)

acf(arimaoil$residuals)

qqnorm(std\_res\_oil, main = "QQ Plot Oil")

qqline(std\_res\_oil)

arimagold <- arima(losses$Gold, order = c(0,0,1), include.mean = F)

std\_res\_gold <- (arimagold$residuals - mean(arimagold$residuals))/sd(arimagold$residuals)

acf(arimagold$residuals)

qqnorm(std\_res\_gold, title = "QQ PLot Gold")

qqline(std\_res\_gold)

arimametal <- arima(losses$Metals, order = c(0,0,0), include.mean = F)

std\_res\_metal <- (arimametal$residuals - mean(arimametal$residuals))/sd(arimametal$residuals)

acf(arimametal$residuals)

qqnorm(std\_res\_metal, main = "QQ PLot Metals")

qqline(std\_res\_metal)

# remaining asset models

arimaduk <- arima(losses$DUK, order = c(2,0,1), include.mean = F)

arimad <- arima(losses$D, order = c(1,0,0), include.mean = F)

arimatmus <- arima(losses$TMUS, order = c(0,0,0), include.mean = F)

arimavz <- arima(losses$VZ, order = c(2,0,0), include.mean = F)

arimaamt <- arima(losses$AMT, order = c(0,0,0), include.mean = F)

arimacci <- arima(losses$CCI, order = c(0,0,0), include.mean = F)

arimayen <- arima(losses$Yen, order = c(0,0,0), include.mean = F)

arimaeuro <- arima(losses$Euro, order = c(0,0,0), include.mean = F)

arimapounds <- arima(losses$Pounds, order = c(0,0,0), include.mean = F)

# predicting losses for the next period

aep\_forecast <- predict(arimaaep, n.ahead = 1)

d\_forecast <- predict(arimad, n.ahead = 1)

duk\_forecast <- predict(arimaduk, n.ahead = 1)

tmus\_forecast <- predict(arimatmus, n.ahead = 1)

vz\_forecast <- predict(arimavz, n.ahead = 1)

amt\_forecast <- predict(arimaamt, n.ahead = 1)

cci\_forecast <- predict(arimacci, n.ahead = 1)

yen\_forecast <- predict(arimayen, n.ahead = 1)

euro\_forecast <- predict(arimaeuro, n.ahead = 1)

pounds\_forecast <- predict(arimapounds, n.ahead = 1)

oil\_forecast <- predict(arimaoil, n.ahead = 1)

gold\_forecast <- predict(arimagold, n.ahead = 1)

metal\_forecast <- predict(arimametal, n.ahead = 1)

# accumulating the forecasts

preds <- c(AEP = aep\_forecast$pred[1], D = d\_forecast$pred[1],

DUK = duk\_forecast$pred[1], TMUS = tmus\_forecast$pred[1],

VZ = vz\_forecast$pred[1], AMT = amt\_forecast$pred[1],

OIL = oil\_forecast$pred[1], CCI = cci\_forecast$pred[1],

YEN = yen\_forecast$pred[1], EURO = euro\_forecast$pred[1],

POUNDS = pounds\_forecast$pred[1], GOLD = gold\_forecast$pred[1],

METAL = metal\_forecast$pred[1])

preds

ESgarch <- function(y, p = 0.95){

gfit <- garchFit(~garch(1,1), data = y, cond.dist = "std", trace = F)

sigma <- predict(gfit, n.ahead = 1)[3]

df <- coef(gfit)['shape']

ES <- sigma\*(dt(qt(p,df),df)/(1-p))\*((df+(qt(p,df))^2)/(df-1))

return(ES)

}

es<- c(AEP = ESgarch(losses$AEP)[1], D = ESgarch(losses$D)[1],

DUK = ESgarch(losses$DUK)[1], TMUS = ESgarch(losses$TMUS)[1],

VZ = ESgarch(losses$AEP)[1], AMT = ESgarch(losses$AMT)[1],

CCI = ESgarch(losses$CCI)[1], YEN = ESgarch(losses$Yen)[1],

EURO = ESgarch(losses$Euro)[1], POUNDS = ESgarch(losses$Pounds)[1],

GOLD = ESgarch(losses$Gold)[1], OIL = ESgarch(losses$Oil)[1],

METAL = ESgarch(losses$Metals)[1])

unlist(es)

# portfolio expected shortfall

es\_pgmv <- ESgarch(na.omit(gmvp), p = 0.95) # PGMV

es\_pgmv

es\_mdp <- ESgarch(na.omit(mdp), p = 0.95) # MDP

es\_mdp

unlist(c(PGMV = es\_pgmv, MDP = es\_mdp))

# Copula model

gfitgoy <- lapply(losses, garchFit, formula=~arma(0,0) + garch(1,1),

cond.dist = "std", trace = F)

gprog <- unlist(lapply(gfitgoy, function(x) predict(x, n.ahead = 1)[3]))

gshape <- unlist(lapply(gfitgoy, function(x) x@fit$coef[5]))

gresid <- as.matrix(data.frame(lapply(gfitgoy,function(x) x@residuals / sqrt(x@h.t))))

head(gresid)

U <- sapply(1:13, function(y) pt(gresid[, y], df = gshape[y]))

head(U)

cop <- fit.tcopula(Udata = U, method = "Kendall")

cop$P

set.seed(1)

rcop <- rcopula.t(100000, df = cop$nu, Sigma = cop$P)

qcop <- sapply(1:13, function(x) qstd(rcop[, x], nu = gshape[x]))

head(qcop)

ht.mat <- matrix(gprog, nrow = 100000, ncol = ncol(losses), byrow = TRUE)

head(ht.mat)

Weights(pgmv\_weights2)

pfall <- (qcop \* ht.mat) %\*% gmvw

head(pfall)

tail(pfall)

pfall.es95 <- median(tail(sort(pfall), 5000))

pfall.es95

ESgarch(na.omit(gmvp), p = 0.95)