

Risk Management Homework 2

Nitish Ramkumar

Collaborators: Carlos Quicazan, Justin Ge, Yuying Wang

Question 1

The bank for my group was Morgan Stanley.

Technique

The technique used by Morgan Stanley is historical simulation for major market risk factors and Monte Carlo simulation for name-specific risk in certain equity and fixed income exposures.

Data

The data used by Morgan Stanley include historical observation of daily changes in key market indices or other market factors (“market risk factors”) and information on the sensitivity of the portfolio values to these market risk factor changes.

Time Horizon

The Company’s VaR model uses approximately 4 years of historical data to characterize potential changes in market risk factors.

Confidence Level

A 95% 1-day VaR confidence level was used for the model. The Company’s average 95%/one-day Trading VaR for fiscal 2008 was \$98 million. The most frequently occurring value was between \$97 million and \$103 million, while for approximately 92% of trading days during the fiscal year, VaR ranged between \$88 million and \$109 million.

Number of Exceptions

Assuming no intra-day trading, for a 95% one-day VaR, the expected number of times that trading losses should exceed VaR during the fiscal year is 13, and, in general, if trading losses were to exceed VaR more than 21 times in a year, the accuracy of the VaR model could be questioned.

Changes in VaR methodology

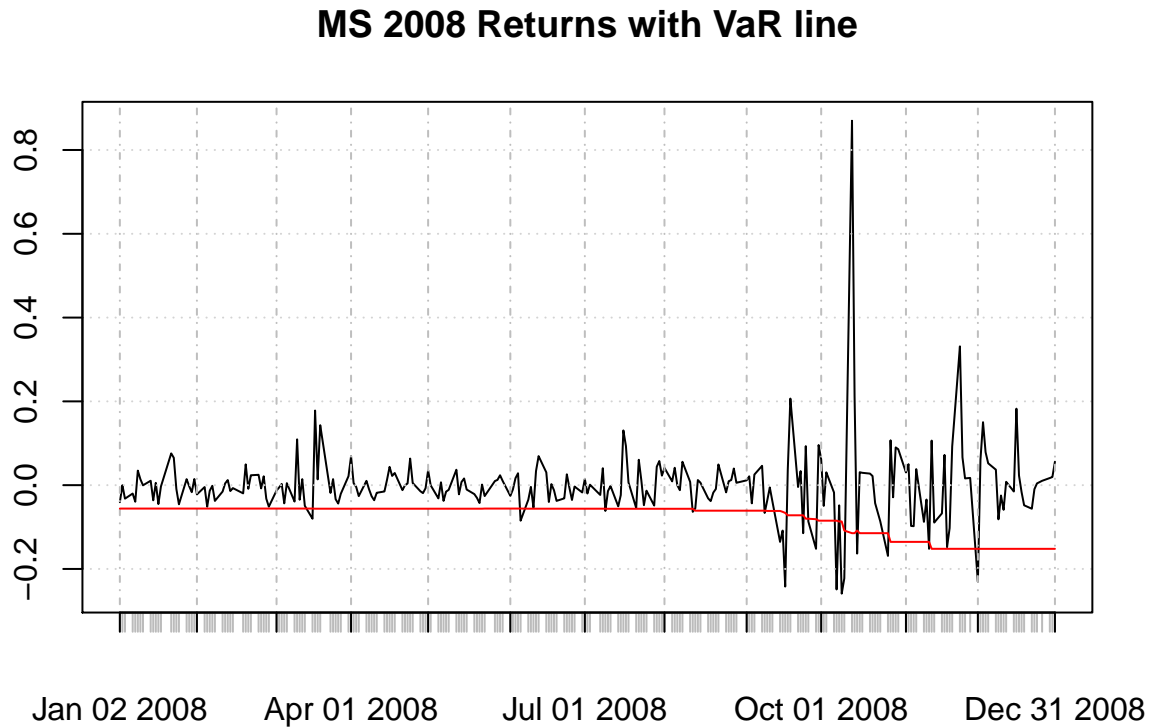
In response to increased levels of market volatility realized during fiscal 2008, the Company has reviewed the appropriateness of the implementation of its VaR models and has made certain changes to more accurately capture risks generated by certain fixed income products. These changes include additional historical time series that provide broader product coverage of subprime consumer and other mortgage products as well as updated mappings of risk exposures to historical price time series.

Question 2

2a

Based on the historical method, we can calculate the VaR at each day. We can follow these steps for each day:
1) Find all data before the current date 2) after arranging them, take the $\text{ceil}[(1-c) * \text{number of observation}]$ th value

The plot of the VaR line along with the returns for Morgan Stanley at 2008 is as below:



2b

Once we have calculated the VaR for every day, we can now back test it by counting the number of times the returns have gone less than the daily VaR over the year. The significance of this can be concluded using the chi-sq test.

The results of the chi-square test is as below

```
## Test Val ChiSq Val
## 40.673278 3.841459
```

As it can be seen, there is a strong rejection of the null hypothesis that the VaR is set correctly. This might be because of how badly the VaR has done during the financial crisis period.

2c

As mentioned in the Morgan Stanley report, this work also concluded with 18 exceptions. The paper mentions that it will question the methodology if there are 21 exceptions. This is not necessarily the case, as we reject the VaR levels convincingly.

Question 3

3a

Lets get all the price data using quantmod (or excel only for UBS, as quantmod doesnt return any value. Used CRSP instead). Some companies (especially foreign like BNP Paribas) dont trade on same day as the US banks. So we can standardize the dates by using na.locf() after a merge.

After that returns can be calculated using the data. The latest returns are as below:

```
## [1] "GS" "JPM" "C" "BCS" "MS" "DB" "BAC" "BNP" "CS"
```

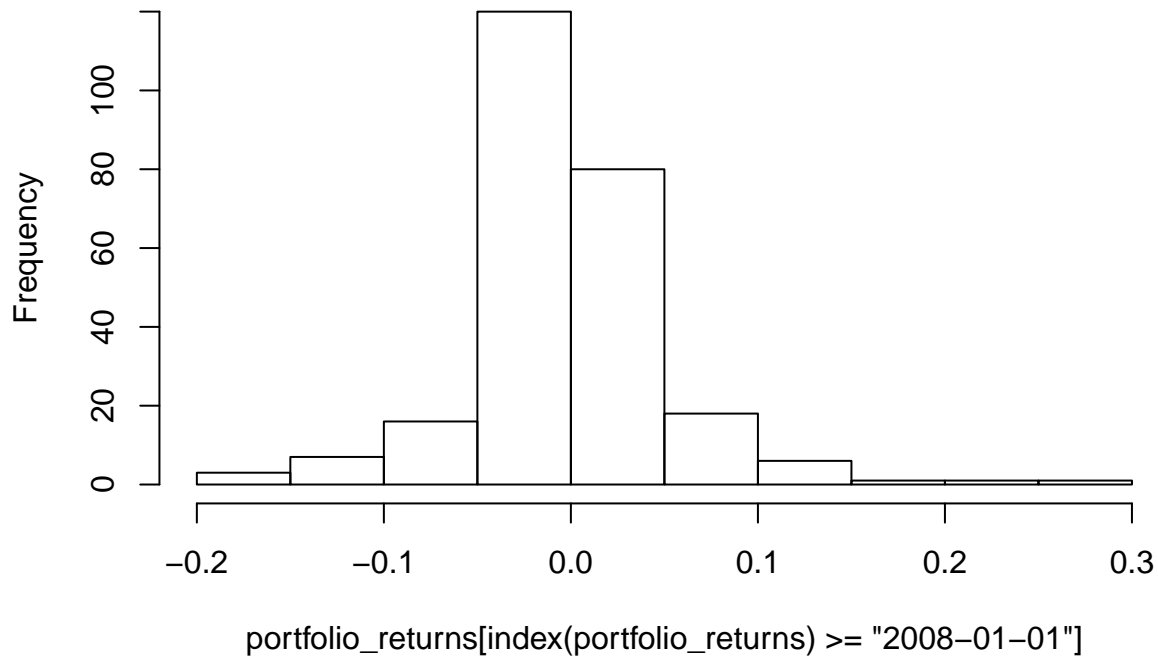
```
## [1] "BNP.PA"
```

Table 1: latest daily returns

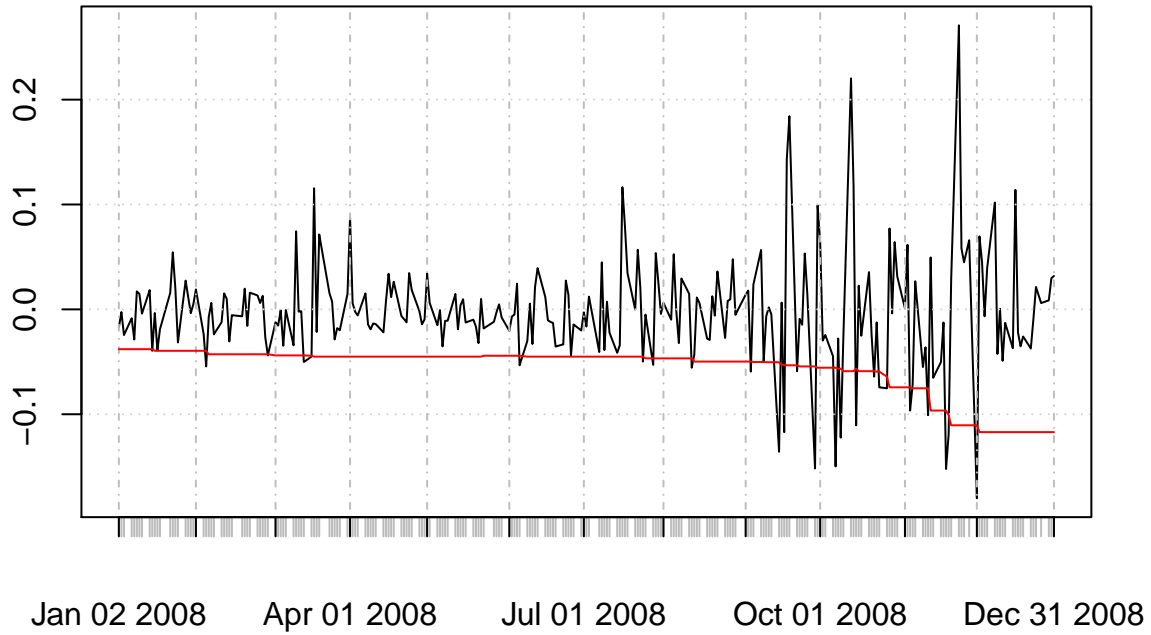
GS	UBS	JPM	Citi	Barclays	MS	DB	Bank Of America	BNP	CS
0.0284	0.0178	0.0168	-0.0132	0.1187	0.056	0.0042	0.0634	0.0083	0.0269

Now, we can calculate the Portfolio VaR of returns, by performing similar steps as before, but this time on the portfolio returns, which is nothing but the weighted sum of individual returns. The portfolio VaR is **\$1755668**

Portfolio returns



Portfolio returns with VaR line



3b

DVaR and CVaR can be calculated in multiple ways.

Increment amount and recalculate VaR

In this method, we can increase the weights of each bank individually and re-calculate the Portfolio VaR accordingly.

The difference in portfolio VaR divided by the the increase in weights will give us the DVaR of the bank. Let us use the increment of \$1

Table 2: DVaR using increment logic

	DVaR
GS	0.1391624
UBS	0.1549839
JPM	0.1219931
Citi	0.1092063
Barclays	0.0241796
MS	0.2421603
DB	0.0752659
Bank Of America	0.0795262
BNP	0.0327071
CS	0.0953032

Table 3: CVaR using increment logic

	CVaR
GS	139162.43
UBS	309967.85
JPM	121993.15
Citi	218412.66
Barclays	24179.62
MS	484320.56
DB	75265.93
Bank Of America	159052.39
BNP	32707.10
CS	190606.46

We can confirm the CVaRs by adding them up and comparing with the Portfolio Var

```
#Portfolio VaR
abs(var_port_08[length(var_port_08),]*totalAmt*1000000)
```

```
##           [,1]
## 2008-12-31 1755668
```

```
#Sum of CVaR
sum(cvar_new)
```

```
## [1] 1755668
```

Beta of Components

To compute the DVaR, we can get the beta of each bank with the portfolio using

$$\beta_i = \frac{cov_{i,P}}{\sigma_P^2}$$

After calculating Beta, $DVaR_i = \beta_i * \text{VaR of portfolio}$

Sample of the DVaR results for stock returns (in %) for 31st December is as below:

Table 4: DVaR using Beta logic

GS	UBS	JPM	Citi	Barclays	MS	DB	Bank Of America	BNP	CS
0.1022	0.123	0.0996	0.1411	0.1048	0.1619	0.0977	0.1236	0.0481	0.102

CVaR (Component VaR) can also be calculated using the Beta. The formula is

$$CVaR_i = \beta_i * (AmountInvested)_i * \text{Portfolio VaR}$$

Table 5: CVaR based on Beta logic

	2008-12-31
GS	102163.20
UBS	245966.38
JPM	99602.49

	2008-12-31
Citi	282273.36
Barclays	104828.33
MS	323704.39
DB	97730.18
Bank Of America	247261.85
BNP	48052.30
CS	204085.67

The CVaR results can be confirmed by summing up the results and checking if it matches with the portfolio VaR (calculated using the historical method on the portfolio returns)

```
#Sum of component VaR
abs(xts(apply(cvar,1,sum),index(cvar))[nrow(var_port_08),])
```

```
##           [,1]
## 2008-12-31 1755668
```

```
#Portfolio VaR
abs(var_port_08[nrow(var_port_08)]*15000000)
```

```
##           [,1]
## 2008-12-31 1755668
```

3c

The CVaR indicates how much each bank has impact on the portfolio VaR. As it can be seen Morgan Stanley has the highest impact on the portfolio VaR.

As explained before, the CVaR's should add up to the Portfolio VaR.

3d

The decision can be based on various factors.

CVaR

Based on VaR, Morgan Stanley has the highest CVaR. This means that Morgan Stanley can take more risk, so it is worth increasing the weight for Morgan Stanley.

RAROC

$$\text{RAROC} = \frac{\text{Profit}}{\text{VaR}(\text{dollars})}$$

Profit was calculated using the average returns of all the days over the data period

The RAROC of the banks are as follows

Table 6: RAROC for banks

	2008-12-31
GS	-6.0028
UBS	-6.1154

	2008-12-31
JPM	0.5954
Citi	-5.9530
Barclays	-6.4261
MS	-2.0943
DB	-7.0259
Bank Of America	-4.4343
BNP	-8.2445
CS	-3.2010

Based on the RAROC values, we should be overweighting JP Morgan as it is the only company with positive RAROC.

R Code

```
suppressMessages(require(quantmod))
suppressMessages(require(lubridate))
suppressMessages(require(knitr))

suppressMessages(MS <- getSymbols("MS",auto.assign = FALSE,from='2006-01-01',to='2008-12-31'))
MS$Returns <- (MS$MS.Adjusted - lag(MS$MS.Adjusted))/lag(MS$MS.Adjusted)
MS <- MS[-1]

calculateDailyVar <- function(stocks,dates,c){
  var <- data.frame(Date= as.Date(character()),VaR = double())
  for(count in 1:length(dates)){
    #Find VaR position
    stock_historical <- stocks[index(stocks) < dates[count]]
    varPosition <- ceiling(nrow(stock_historical)*(1-c))
    var[count,1] <- dates[count]

    #find return for that position
    hist_df <- as.data.frame(stock_historical)
    hist_df <- hist_df[order(hist_df[,1]),]
    var[count,2] <- hist_df[varPosition]
  }
  return(var)
}

c <- 0.99
dates <- index(MS[index(MS)>='2008-01-01'])
var_2008 <- data.frame(Date= as.Date(character()),VaR = double())
var08 <- calculateDailyVar(MS$Returns,dates,c)
var08 <- xts(var08$VaR,var08$Date)
colnames(var08) <- "VaR"
plot(MS[index(MS)>='2008-01-01',"Returns"],type="l",main="MS 2008 Returns with VaR line")
lines(var08,col="red")

testChiSq <- function(data,var,confInt){
  m <- nrow(data[data < var,])
}
```

```

n <- nrow(data)
testVal <- -2*log(c^(n-m) * (1-c)^m) + 2*log((1-(m/n))^(n-m)*(m/n)^m)
chisq_val <- qchisq(p=confInt,df=1)
result <- c(testVal,chisq_val)
names(result) <- c("Test Val","ChiSq Val")
return(result)
}

testChiSq(MS[index(MS)>='2008-01-01']$Returns,var08,0.95)

suppressMessages(getSymbols(c("GS","JPM","C","BCS","MS","DB","BAC","BNP","CS"),from='2006-01-01',
                             ,to='2008-12-31',warnings = FALSE))
getSymbols("BNP.PA",from='2006-01-01',to='2008-12-31')
setwd("C:/_UCLA/Quarter3/RiskMgmt/Assignment/Assignment2")
ubs <- read.csv("UBS.csv",header=TRUE,stringsAsFactors = FALSE)
ubs <- xts(ubs[, "Adj.Close."],mdy(ubs[,1]))

#Combine all prices
company_adjPrice <- data.frame(Date=index(GS),GS=GS$GS.Adjusted,JPM=JPM$JPM.Adjusted,
                               Citi=C$C.Adjusted,BARC=BCS$BCS.Adjusted,MS=MS$MS.Adjusted,
                               DB=DB$DB.Adjusted, BAC=BAC$BAC.Adjusted,CS=CS$CS.Adjusted,UBS=ubs)
company_adjPrice <- xts(company_adjPrice[,-1],company_adjPrice$Date)

company_adjPrice <- merge(company_adjPrice,BNP.PA$BNP.PA.Adjusted, join="left")
company_adjPrice <- na.locf(company_adjPrice)
company_adjPrice <- company_adjPrice[,c("GS.Adjusted","UBS","JPM.Adjusted","C.Adjusted","BCS.Adjusted",
                                         "MS.Adjusted","DB.Adjusted","BAC.Adjusted",
                                         "BNP.PA.Adjusted","CS.Adjusted")]

#Compute all returns
cmp_ret <- (company_adjPrice - lag(company_adjPrice))/lag(company_adjPrice)
colnames(cmp_ret) <- c("GS","UBS","JPM","Citi","Barclays","MS","DB","Bank Of America","BNP","CS")
cmp_ret <- cmp_ret[-1]
kable(round(cmp_ret[nrow(cmp_ret),],4),caption="latest daily returns")

#Calculate individual var
var <- matrix(nrow=length(dates),ncol=ncol(cmp_ret))
for(count in 1:ncol(cmp_ret)){
  var[,count] <- matrix(calculateDailyVar(cmp_ret[,count],dates,0.99)[,2],ncol=1)
}
colnames(var) <- c("GS","UBS","JPM","Citi","Barclays","MS","DB","Bank Of America","BNP","CS")
var <- xts(var,dates)

#Calculate portfolio returns
totalAmt <- 15
weights <- c(1/15,2/15,1/15,2/15,1/15,2/15,1/15,2/15,1/15,2/15)
portfolio_returns <- apply(cmp_ret,1,function(x){
  x%*%weights
})
portfolio_returns <- xts(portfolio_returns,index(cmp_ret))
var_port_08 <- calculateDailyVar(portfolio_returns,dates,c)
var_port_08 <- xts(var_port_08[,2],var_port_08[,1])

```



```

hist(portfolio_returns[index(portfolio_returns)>='2008-01-01'],main="Portfolio returns")

plot(portfolio_returns[index(portfolio_returns)>='2008-01-01'],main="Portfolio returns with VaR line")
lines(var_port_08,col="red")

#DVAR
dvar_new <- c()
amount <- c(1000000,2000000,1000000,2000000,1000000,2000000,1000000,2000000,1000000,2000000)
for(count in 1:length(amount)){
  amountNew <- amount
  amountNew[count] <- amountNew[count]+1
  portfolio_val <- apply(cmp_ret,1,function(x){
    x%%amountNew
  })
  portfolio_val <- xts(portfolio_val,index(cmp_ret))

  var_port <- calculateDailyVar(portfolio_val,dates,c)
  dvar_new[count] <- (abs(var_port[nrow(var_port),2]) - abs(var_port_08[length(var_port_08),]
    *totalAmt*1000000))/1
}

dvar_new <- matrix(dvar_new,nrow=1)
colnames(dvar_new) <- c("GS","UBS","JPM","Citi","Barclays","MS","DB","Bank Of America","BNP","CS")
kable(t(dvar_new),col.names = "DVAR",caption = "DVAR using increment logic")

cvar_new <- dvar_new * amount
kable(t(cvar_new),col.names = "CVaR",caption="CVaR using increment logic")

#Portfolio VaR
abs(var_port_08[length(var_port_08),]*totalAmt*1000000)

#Sum of CVaR
sum(cvar_new)

#compute betas
betas_cmp <- double(ncol(cmp_ret))
for(count in 1:ncol(cmp_ret)){
  betas_cmp[count] <- cov(portfolio_returns,cmp_ret[,count])/(sd(portfolio_returns)^2)
}
names(betas_cmp) <- c("GS","UBS","JPM","Citi","Barclays","MS","DB","Bank Of America","BNP","CS")

dvar <- matrix(nrow = length(dates),ncol=ncol(cmp_ret))
for(count in 1:length(dates)){
  dvar[count,] <- betas_cmp*var_port_08[count,]
}
dvar <- xts(dvar,dates)
colnames(dvar) <- c("GS","UBS","JPM","Citi","Barclays","MS","DB","Bank Of America","BNP","CS")
kable(round(abs(dvar[nrow(dvar),]),4),caption="DVAR using Beta logic")

#CVAR
cvar <- matrix(nrow = length(dates),ncol=ncol(cmp_ret))
for(count in 1:length(dates)){
  cvar[count,] <- betas_cmp*var_port_08[count,]*weights*totalAmt*1000000
}

```

```

}
cvar <- xts(cvar,dates)
colnames(cvar) <- c("GS","UBS","JPM","Citi","Barclays","MS","DB","Bank Of America","BNP","CS")
kable(t(abs(cvar[nrow(cvar),])),caption = "CVaR based on Beta logic")

#Sum of component VaR
abs(xts(apply(cvar,1,sum),index(cvar))[nrow(var_port_08),])

#Portfolio VaR
abs(var_port_08[nrow(var_port_08)]*15000000)

cmp_ret_2008 <- cmp_ret[index(cmp_ret) >= '2008-01-01',]
profits <- apply(cmp_ret_2008,2,mean)*weights*15000000*nrow(cmp_ret_2008)
var_dollar_port <- var[nrow(var),]*weights*15000000
raroc <- profits/abs(var_dollar_port)
kable(round(t(raroc),4),caption="RAROC for banks")

```