# Credit Analysis - Value at Risk & Conditional Value at Risk

Hassan Mir

2024-05-21

In this script, I perform financial analysis and risk assessment on a portfolio of assets. I start with exploratory data analysis and extreme value theory calculations, followed by volatility estimation using GARCH modeling. Finally, I calculate Value at Risk (VaR) through historical simulation to understand the portfolio's risk and return characteristics for informed investment decisions.

```
warnings=FALSE
library('rugarch')
```

```
## Loading required package: parallel
```

```
##
## Attaching package: 'rugarch'
```

```
## The following object is masked from 'package:stats':
##
##     sigma
```

```
library('evir')
library('gmm')
```

```
## Warning: package 'gmm' was built under R version 4.3.3
```

```
## Loading required package: sandwich
```

```
library('ggplot2')
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```
##
## Attaching package: 'ggplot2'
```

```
## The following object is masked from 'package:evir':
##
##     qplot
```

```
library('data.table')
```

```
## Warning: package 'data.table' was built under R version 4.3.2
```

```
library('readxl')
```

```
## Warning: package 'readxl' was built under R version 4.3.3
```

```
library('MASS')
```

```
## Warning: package 'MASS' was built under R version 4.3.3
```

**Calculating Value at Risk using Historical Simulation.**

```
X<-read_excel('C:/Users/dell/Desktop/Sem 2/Mathematical and Quantitative Finance/Labs/FourAssets.xls', 
X<-as.data.frame(X)
alphas<-c(4000000,3000000,1000000,2000000)
```

Creating a simple return function.

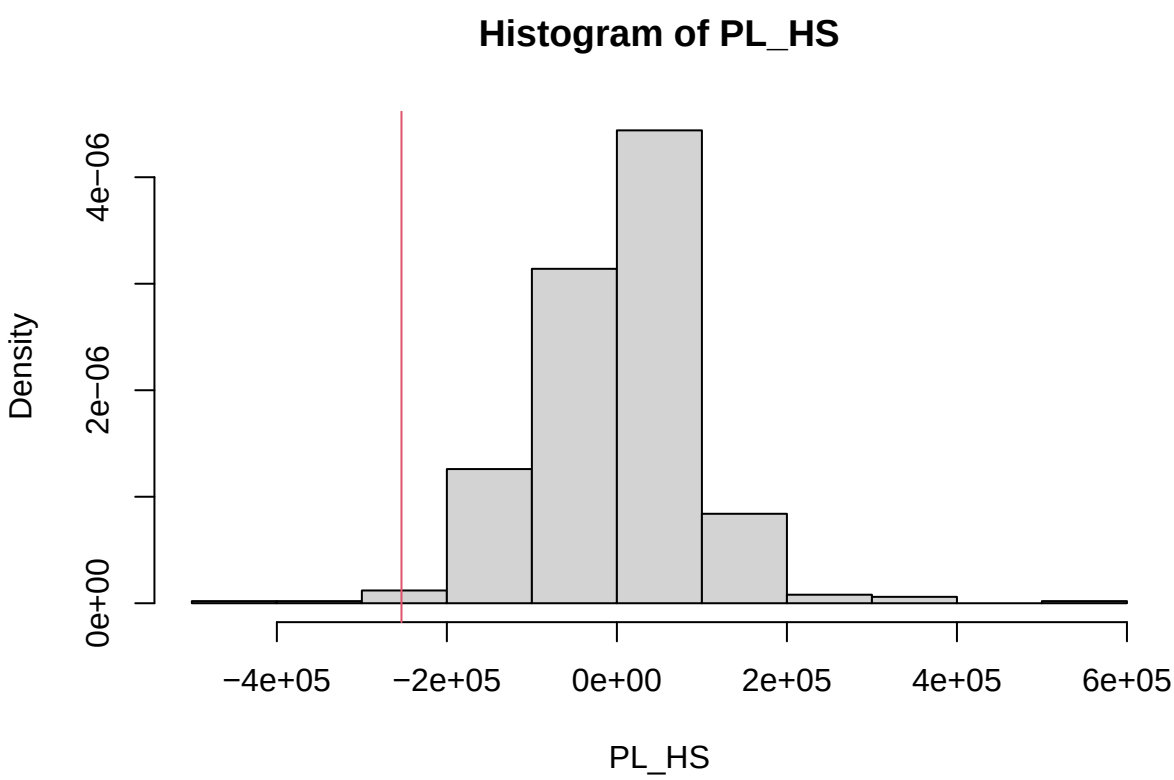
```
SimpleReturn<-function(p) {diff(p)/p[-length(p)]}
```

Calculating simple returns and finding out relevant portfolio profit and losses and then calculating the VaR.

```
SR_HS<-apply(X=X,MARGIN=2,FUN='SimpleReturn')
m<-dim(SR_HS)[1]

ALPHAS_HS<-matrix(rep(x=alphas,times=m),byrow=TRUE,nrow=m)

M_HS<-ALPHAS_HS*SR_HS

M_HS2<-sweep(x=SR_HS,MARGIN=2,STATS=alphas,FUN="*")

PL_HS<-apply(M_HS,MARGIN=1,FUN='sum')

cl<-0.99
tail<-1-cl

VaR_HS<-sort(PL_HS)[floor(tail*length(PL_HS))]
VaR_HS
```

```
## [1] -253385
```

```
hist(PL_HS,freq=FALSE);abline(v=VaR_HS,col=2)
```




**Calculating Value at Risk using Model Building Approach.**

First I calculate the covariance of SR_HS.

```
S<-cov(SR_HS)
S
```

```
##               V1           V2           V3            V4
## V1  1.229524e-04 7.696591e-05 7.682514e-05 -9.493488e-06
## V2  7.696591e-05 2.013973e-04 1.821076e-04  3.944302e-05
## V3  7.682514e-05 1.821076e-04 1.953506e-04  4.078130e-05
## V4 -9.493488e-06 3.944302e-05 4.078130e-05  1.913129e-04
```

Calculating 1-day 99% VaR.

```
PLvariance<-alphas%*%S%*%matrix(alphas,ncol=1)

VaR_MB<-qnorm(p=tail)*sqrt(PLvariance)

VaR_MB
```

```
##           [,1]
## [1,] -217975.1
```

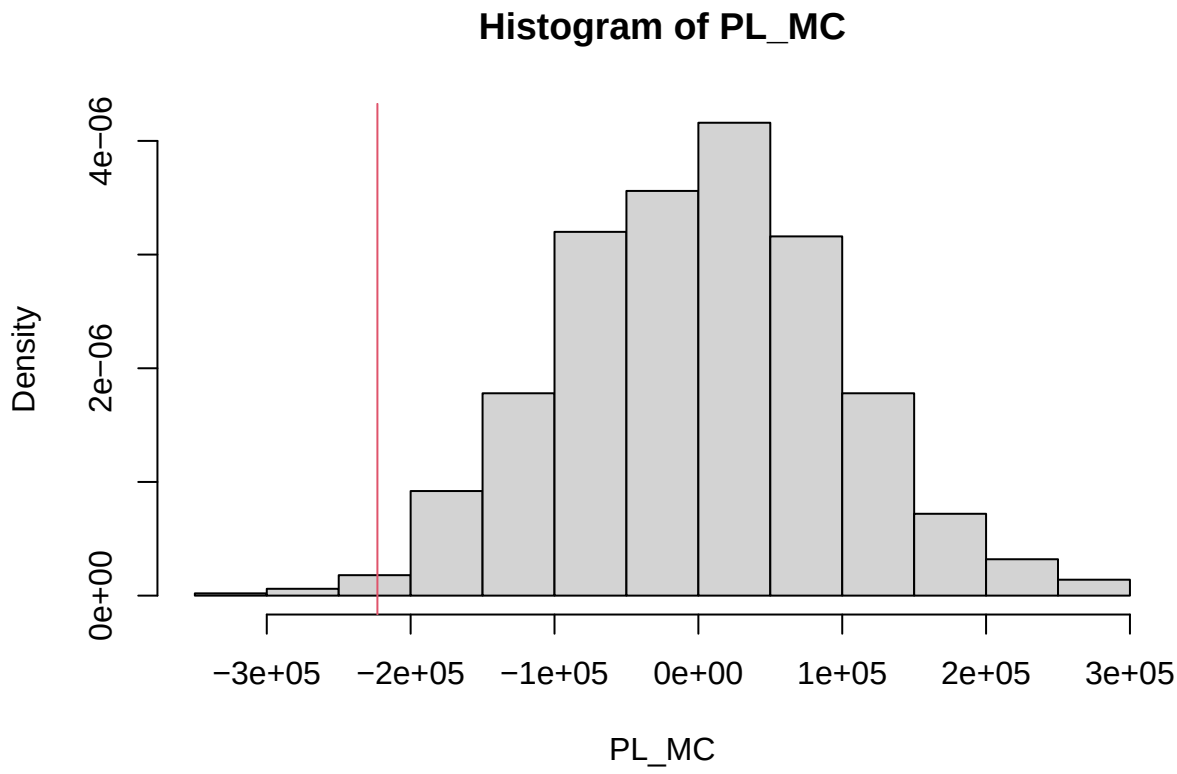
**Calculating Value at Risk using Monte Carlo Approach.**

```
set.seed(1)
```

```
T<-1000
SR_MC<-mvrnorm(n=T,mu=c(0,0,0,0),Sigma=S)

ALPHAS_MC<-matrix(rep(x=alphas,times=T),byrow=TRUE,nrow=T)

M_MC<-ALPHAS_MC*SR_MC

PL_MC<-apply(M_MC,MARGIN=1,FUN='sum')

VaR_MC<-sort(PL_MC)[floor(tail*length(PL_MC))]
VaR_MC
```

```
## [1] -223087.5
```

```
hist(PL_MC,freq=FALSE);abline(v=VaR_MC,col=2)
```




### Calculating Conditional Value at Risk (CVaR)

Finding out expected shortfall using historical simulation.

```
mean(PL_HS[PL_HS<=VaR_HS])
```

```
## [1] -327181.2
```

Finding out expected shortfall using Monte Carlo.

```
mean(PL_MC[PL_MC<=VaR_MC])
```

```
## [1] -256319
```

Finding out expected shortfall using Model Building approach.

```
-sqrt(PLvariance)*exp(-qnorm(p=tail)^2/2)/(sqrt(2*pi)*tail)
```

```
##           [,1]
## [1,] -249726.3
```