Assignment #7 VaR of a Stock Portfolio

Scott Morgan

Solve textbook exercises 5-6 on page 578. (Exercise 5 is worth 30 points, while 6 is worth 20 points). ***Please note a typo in question 6 b) on page 578: prices[,500] should be prices[500,] instead.***

5. Suppose the risk measure R is VaR(α) for some α. Let P1 and P2 be two portfolios whose returns have a joint normal distribution with means μ1 and μ2, standard deviations σ1 and σ2, and correlation ρ. Suppose the initial investments are S1 and S2. Show that R(P1+P2) ≤ R(P1)+R(P2) under joint normality.

If returns R_1 and R_2 are joint normal, the profit or loss of P_1 + P_2 portfolio S_1R_1 + S_2+R_2 is also normally distributed. As mean is additive and standard deviation is subadditive:

$\mu_{X+Y} = \mu_X + \mu_Y$ and $\sigma_{X+Y} \leq \sigma_X + \sigma_Y$.

We begin this proof by noting the VaR(α) for a given portfolio i is as follows (in this case i = 1 and 2):

$VaR(P_i) = -S_i\mu_i - S_i\sigma_i z_\alpha$

The VaR(α) for P_1 + P_2 is therefore:

$$
\begin{aligned}
VaR(P_1 + P_2, \alpha) &= -(S_1\mu_1 + S_2\mu_2) - \sqrt{S_1^2\sigma_1^2 + S_2^2\sigma_2^2 + 2\rho S_1 S_2 \sigma_1\sigma_2 z_\alpha} \\
&\leq -(S_1\mu_1 + S_2\sigma_2) - \sqrt{S_1^2\sigma_1^2 + S_2^2\sigma_2^2 + 2S_1 S_2 \sigma_1\sigma_2 z_\alpha} \\
&= VaR(P_1, \alpha) + VaR(P_2, \alpha)
\end{aligned}
$$

This is because $\rho \leq 1$ and $-z_\alpha > 0$. If (R_1,R_2) were not joint normal we would not be able to determine the distribution of S_1R_1 + S_2+R_2 as z_α could be different in $VaR(P_1 + P_2)$ and $VaR(P_i)$.

6. This problem uses daily stock price data in the file Stock_Bond.csv on the book's website. In this exercise, use only the first 500 prices on each stock. The following R code reads the data and extracts the first 500 prices for five stocks. "AC" in the variables' names means "adjusted closing" price.

```
library(MASS)
library(fEcofin)
library(readr)
library(PerformanceAnalytics)
library(tvarPackage)

Stock_Bond <- read_csv("C:/Users/Scott/Desktop/451/Module7/Stock_Bond.csv")
dat = Stock_Bond
str(dat)
prices = as.matrix(dat[1:500, c(3, 5, 7, 9, 11)])
options(scipen=999) #eliminate scienfitic notation
```

a) What are the sample mean vector and sample covariance matrix of the 499 returns on these stocks?

The sample mean vectors are of the 499 returns are:

```
GM_AC         F_AC         UTX_AC        CAT_AC        MRK_AC
0.00098012820 0.00155666434 0.00005400236 0.00123163873 0.00085686353
```

The sample covariance matrix of the 499 returns is:

```
              GM_AC        F_AC       UTX_AC       CAT_AC       MRK_AC
GM_AC  0.0004081672 0.0002913652 0.0001877597 0.0002314422 0.0001858492
F_AC   0.0002913652 0.0004493332 0.0001973972 0.0002771111 0.0002188128
UTX_AC 0.0001877597 0.0001973972 0.0003744431 0.0002231245 0.0001698379
CAT_AC 0.0002314422 0.0002771111 0.0002231245 0.0005037975 0.0002291694
MRK_AC 0.0001858492 0.0002188128 0.0001698379 0.0002291694 0.0003123000
```

```
#Calculate returns
n_prices = dim(prices)[1]
rets = ( prices[2:n_prices,] - prices[1:(n_prices-1),] ) / prices[1:(n_prices-1),]

#Calculate average and covariance of returns
stock_means = apply( rets, 2, mean )
stock_covs = cov(rets)

stock_means
stock_covs
```

(b) How many shares of each stock should one buy to invest $50 million in an equally weighted portfolio? Use the prices at the end of the series, e.g., prices[500,].

An investor should by the following for an equally weighted $50 million portfolio:

- 594,530 GM shares
- 2,232,143 F shares
- 2,923,977 UTX shares
- 1,821,494 CAT shares
- 1,754,386 MRK shares

These allocations are assuming the purchase of partial or fractional shares is not allowed and not taking transactions costs into account.

```
S = 50000000
n_stocks = dim(rets)[2]
shares<-(S/n_stocks)/prices[500,]
round(shares,0)

# GM_AC    F_AC  UTX_AC  CAT_AC  MRK_AC
# 594530 2232143 2923977 1821494 1754386
```

(c) What is the one-day VaR(0.1) for this equally weighted portfolio? Use a parametric VaR assuming normality.

The one day VaR (0.1) of the equally weighted portfolio is $984,181.50.

```
# Compute the mean return and standard deviation of the portfolio
port_mean_ret = sum((1/5)*stock_means) #average return of stocks = means times wgt
w = rep(1/5, n_stocks) #weights in percentage

#Compute variance and standard deviation of the portfolio
portfolio_var = t(w) %*% stock_covs %*% w #portfolio covariance
port_std=sqrt(portfolio_var) #Portfolio standard deviation

#Compute 1-day VaR(0.1)----------------------------------------

#Method 1: 1-day VaR(0.1) Calculations
var1day_v1 =-S*qnorm(0.1, mean = port_mean_ret, sd = port_std)
var1day_v1
#984181.5

#Method 1: 1-day VaR(0.1) Check
InverseCum1<-qnorm(0.1, mean = port_mean_ret, sd = port_std)
InverseCum1
#[1] -0.01968363

#Method 2: 1-day VaR(0.1) Calculations
require(tvarPackage)
var1day_v2=-S*VaR_norm(0.1,port_mean_ret,port_std)
var1day_v2
#984181.5

#Method 2: 1-day VaR(0.1) Check
InverseCum2=VaR_norm(0.1,port_mean_ret,port_std)
InverseCum2
#-0.01968363
```

(d) What is the five-day Var(0.1) for this portfolio? Use a parametric VaR assuming normality. You can assume that the daily returns are uncorrelated.

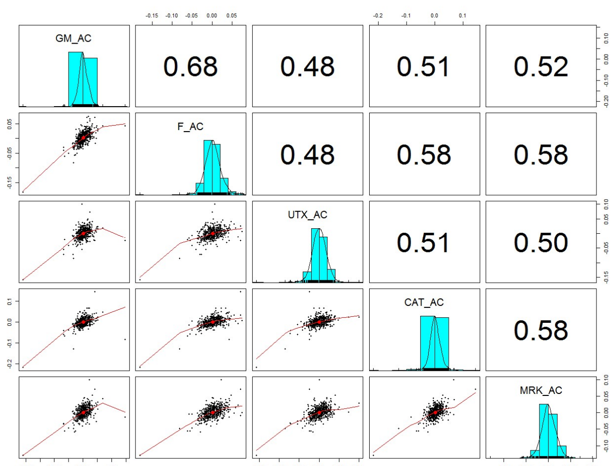
The five day VaR(0.1) of the equally weighted portfolio where the daily returns are uncorrelated is $1,062,887.00. It should be noted that this calculation assumes that the returns are normally distributed. Equation 19.38 of Ruppert and Matteson (2015) provides:

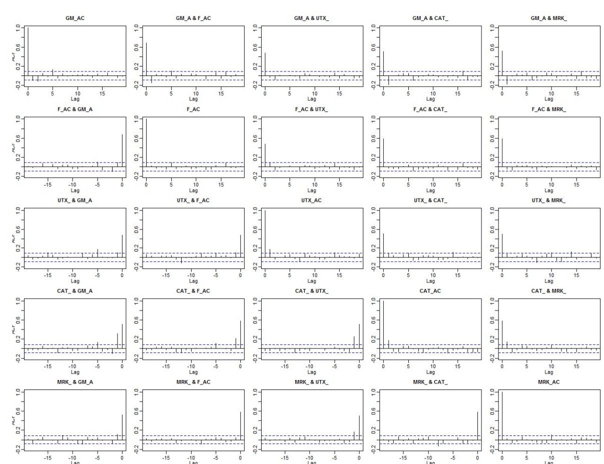
$$\mathrm{VaR}_P^{\mathrm{M\ days}} = -S \times \left\{ M\widehat{\mu}_P^{1\ \mathrm{day}} + \sqrt{M}\Phi^{-1}(\alpha)\widehat{\sigma}_P^{1\ \mathrm{day}} \right\}$$

Where the horizon M is allowed to change from daily to weekly without re-estimating the mean and standard deviation with weekly instead of monthly returns. Instead, one simply uses 19.38 with M = 5.

One concern is that if the distributions are not normal and positive autocorrelation of the daily returns exists, then 19.38 underestimates the M-day VaR. Using the *acf* and *pairs.panel* functions we observe possible single name positive autocorrelation and non-normal distributions.

Note: The five-day VaR of the correlated returns is $2,071,364 (not using the diag function).





```
#Compute variance and standard deviation of the portfolio over 5 days. Page 572
M=5
portfolio_mean_5d = M * sum( (1/5) * stock_means)
portfolio_var_5d = M * sum( (1/5)^2 * diag(stock_covs) ) # uncorrelated returns

#Compute 5-day VaR(0.1)

#Method 1: 5-day VaR(0.1) Calculations
var5day_v1  =-S*qnorm(0.1, mean = portfolio_mean_5d, sd = sqrt(portfolio_var_5d))
var5day_v1
# 1062887

#Method 1: 5-day VaR(0.1) Check
InverseCum1day5<-qnorm(0.1, mean = portfolio_mean_5d, sd = sqrt(portfolio_var_5d))
InverseCum1day5
```

```
#-0.02125774

#Method 2: 5-day VaR(0.1) Calculations
var5day_v2=-S*VaR_norm(0.1,portfolio_mean_5d,sqrt(portfolio_var_5d))
var5day_v2
#1062887

#Method 2: 5-day VaR(0.1) Check
InverseCum2day5<-VaR_norm(0.1,portfolio_mean_5d,sqrt(portfolio_var_5d))
InverseCum2day5
#-0.02125774

#plots
acf(rets)
pairs.panels(rets)
```

PROPOSED EXTRA-CREDIT FOR QUESTION 6 – PARTS A,B,D – 10 POINTS
Using the PerformanceAnalytics package presents an alternative method for calculating these metrics and many other metrics. First, the data needs to be put into xts format:

```
library(MASS)
library(tvarPackage)
library(xts)
library(PerformanceAnalytics)
library(readr)
library(lubridate)
mydata <- read_csv("C:/Users/SMorgan/Desktop/451/Module7_Current/Stock_Bond.csv")
mydata$Date<-parse_date_time(x = mydata$Date,
                orders = c("d m y", "d B Y", "m/d/y"),
                locale = "eng")

mydata = as.data.frame(mydata[1:500, c(1,3, 5, 7, 9, 11)])
rownames(mydata) <- mydata[,1]
mydata <- mydata[, -grep("Date", colnames(mydata))]
mydata <- as.xts(mydata)
returns<-CalculateReturns(mydata,method="simple")
returns<-as.xts(returns)
returns<-na.omit(returns)
S = 50000000
```

#PART A) Note the Arthmetic Mean, Variance and Standard Deviation match above.

```
returns.stats<-table.Stats(returns)
returns.stats
```

```
# GM_AC     F_AC    UTX_AC   CAT_AC   MRK_AC
# Observations    499.0000 499.0000 499.0000 499.0000 499.0000
# NAs               0.0000   0.0000   0.0000   0.0000   0.0000
# Minimum          -0.2102  -0.1821  -0.1568  -0.2168  -0.1298
# Quartile 1       -0.0085  -0.0104  -0.0102  -0.0090  -0.0085
# Median            0.0000   0.0023   0.0000   0.0000   0.0000
# Arithmetic Mean   0.0010   0.0016   0.0001   0.0012   0.0009
# Geometric Mean    0.0008   0.0013  -0.0001   0.0010   0.0007
# Quartile 3        0.0105   0.0132   0.0108   0.0116   0.0105
# Maximum           0.1470   0.0755   0.0992   0.1447   0.0990
# SE Mean           0.0009   0.0009   0.0009   0.0010   0.0008
# LCL Mean (0.95)  -0.0008  -0.0003  -0.0016  -0.0007  -0.0007
# UCL Mean (0.95)   0.0028   0.0034   0.0018   0.0032   0.0024
# Variance          0.0004   0.0004   0.0004   0.0005   0.0003
# Stdev             0.0202   0.0212   0.0194   0.0224   0.0177
# Skewness         -1.5298  -1.1735  -1.0687  -1.6583  -0.4804
# Kurtosis         28.7579  11.6725  10.6174  22.6308   7.9693
```

PART B) – Nothing to be done. The following VaR function in Part C and Part D assumes equal weighting.

PART C) Response matches above.

```
x<-VaR(returns, p=0.90, method="gaussian", portfolio_method="component")
day1_var<-x$VaR
y<-day1_var * S
y
# [1] 984181.5
```

PART D) Weekly returns calculated from daily returns. The VaR function does not allow me to specify uncorrelated returns, therefore this response is different from above at $2,070,447 versus $1,062,887 using uncorrelated returns. Interesting that uncorrelated returns have that much of an impact on the VaR; this speaks to the value of long/short or more market neutral strategies.

```
#Assumes equal weights
df<-mydata
df.new = df[seq(1, nrow(df), 5), ]
returns.new<-CalculateReturns(df.new,method="simple")
returns.new<-as.xts(returns.new)
returns.new<-na.omit(returns.new)

#PerformanceAnalytics // VaR function Assumes equal weights
w<-VaR(returns.new, p=0.90, method="gaussian", portfolio_method="component")
w$VaR*S

table.Autocorrelation(returns.new)#Some positive autocorrelation

#              GM_AC    F_AC  UTX_AC  CAT_AC  MRK_AC
# rho1         0.1903 -0.0470  0.0194 -0.0281 -0.1306
# rho2         0.0479 -0.0058  0.0821 -0.0396 -0.0168
# rho3         0.0550 -0.0605  0.2164 -0.1647 -0.0290
# rho4         0.0649  0.2422 -0.0294 -0.1418  0.1431
# rho5         0.0370  0.0493  0.0606  0.0343  0.0280
# rho6         0.1561  0.0912  0.0852 -0.0404  0.0119
# Q(6) p-value 0.2809  0.2425  0.3328  0.4832  0.6617

table.Distributions(returns.new) #some non-normality
#                          GM_AC   F_AC  UTX_AC  CAT_AC MRK_AC
# Monthly Std Dev         0.0368 0.0454  0.0468  0.0512 0.0396
# Skewness               -0.7383 0.4790 -1.2795 -0.9950 0.0189
# Kurtosis                4.8924 5.3955  9.5516  8.6684 3.3367
# Excess kurtosis         1.8924 2.3955  6.5516  5.6684 0.3367
# Sample skewness        -0.7613 0.4939 -1.3192 -1.0259 0.0195
# Sample excess kurtosis  2.0547 2.5842  6.9581  6.0286 0.4175
```

REFERENCES

Ruppert, David, and David S. Matteson. *Statistics and Data Analysis for Financial Engineering: with R Examples*. Springer, 2015.