1.Risk Measures

2.Univariate t - VaR and ES

3.EVaR

Risk Management

This is sourced from STAD70 course practice questions and sample R code taught by professor Sotos. If you have any questions/concerns/comments feel free to email me: cristal.wang111@gmail.com (mailto:cristal.wang111@gmail.com).

1.Risk Measures

Data

Get last 23years of daily IBM stock prices

```
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

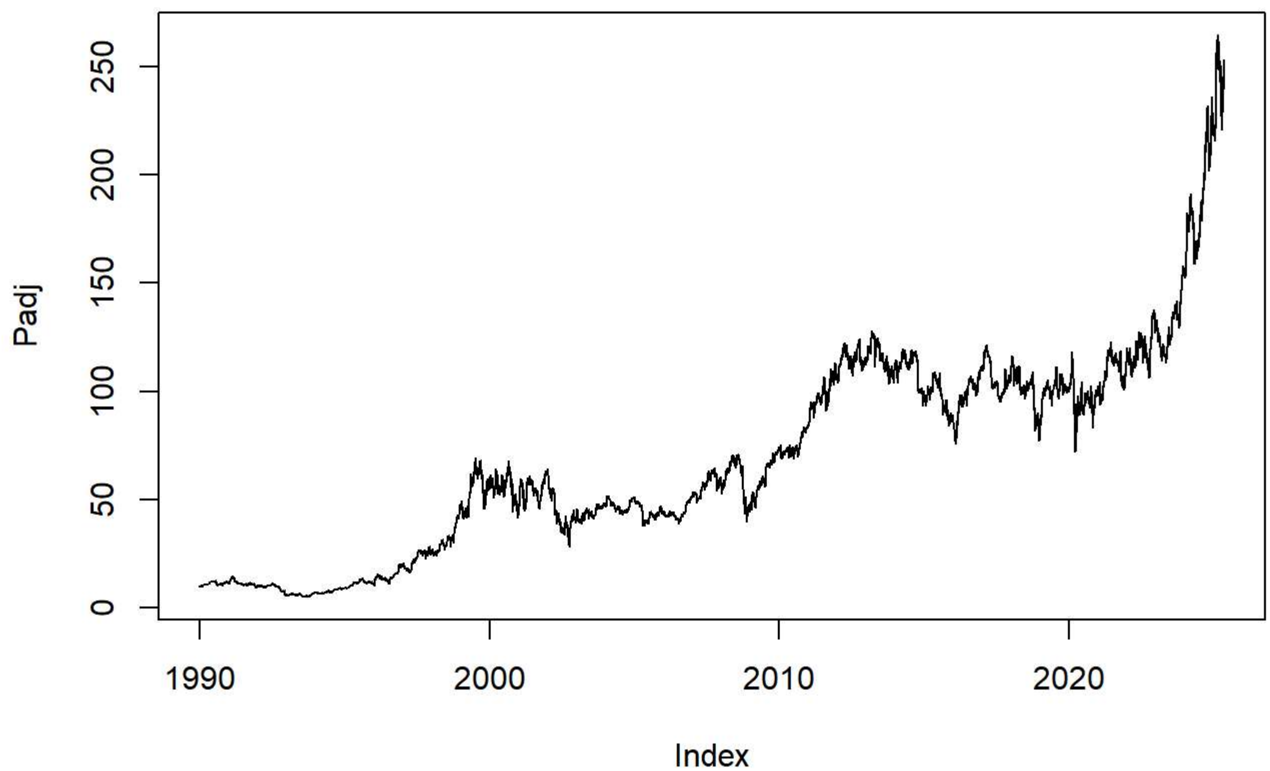
```
library(zoo)
```

```
##
## Attaching package: 'zoo'
```

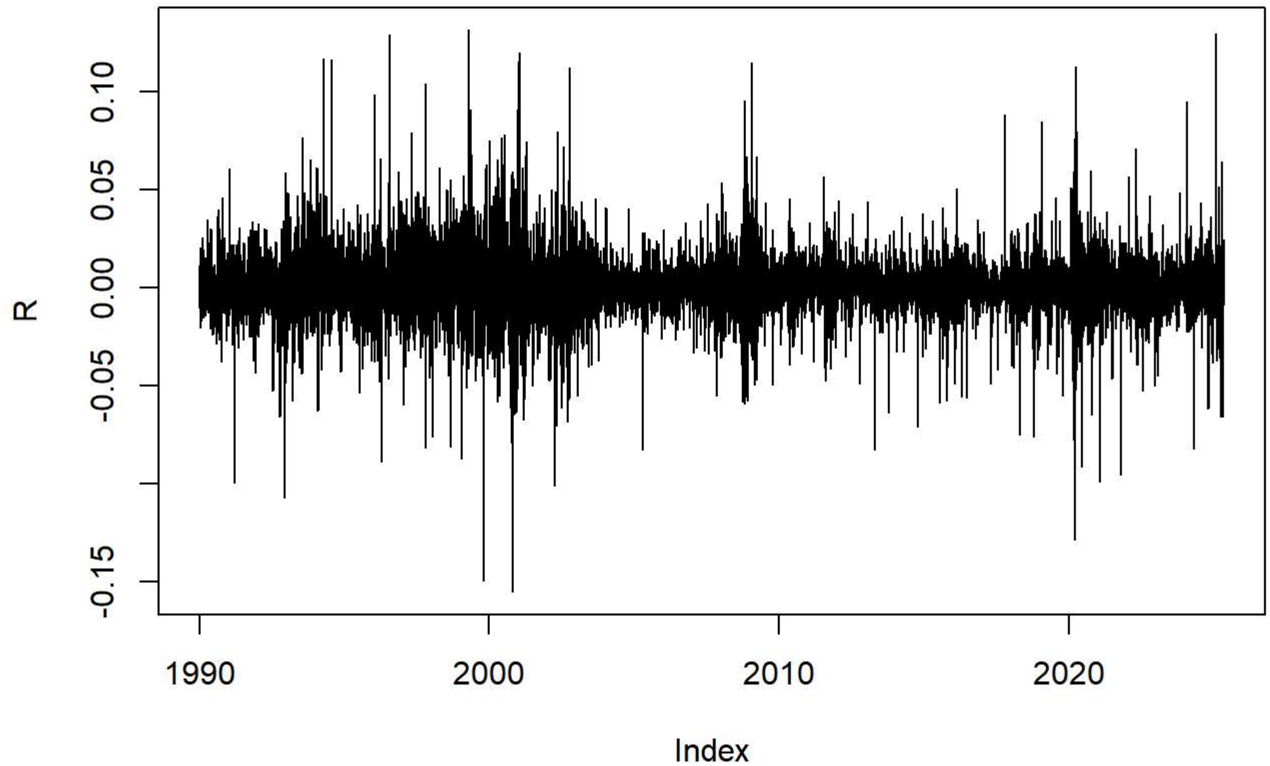
```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
start.date=as.Date("1990-01-01")

# Get last 23years of daily IBM stock prices
Padj=get.hist.quote(instrument = "IBM", start = start.date , quote = "AdjClose", retclass
= "zoo", quiet = T)
plot(Padj)
```




```
# R=diff(Padj)/as.vector(Padj[-length(Padj)]) # net returns
R = diff(Padj)/lag(Padj,k=-1)  # net returns
r=log(1+R) # Log returns. r = log( R + 1 ) <=> R = exp( r ) - 1
plot(R)
```



1.1.Parametric VaR using t-distr

```
library(MASS) # for fitdistr()
parm=fitdistr(R, "t") # get Student's t MLE of (mu,sigma,nu)
mu=parm$est[1]; sig=parm$est[2]; dF=parm$est[3]
alpha=.01

t.quant=qt(alpha,dF,lower.tail = TRUE)
(VaR.t= -mu -sig * t.quant)
```

```
##          m
## 0.04748055
```

```
(CVaR.t= -mu + sig/alpha * dt(t.quant,dF) * (dF+t.quant^2)/(dF-1) )
```

```
##          m
## 0.07250409
```

```
library(MASS) # for fitdistr()
L = -R
parm=fitdistr(L, "t") # get Student's t MLE of (mu,sigma,nu)
mu=parm$est[1]; sig=parm$est[2]; dF=parm$est[3]
alpha=.01

t.quant=qt(alpha,dF,lower.tail = FALSE)
(VaR.t= mu+sig * t.quant)
```

```
##          m
## 0.04748055
```

```
(CVaR.t= mu + sig/alpha * dt(t.quant,dF) * (dF+t.quant^2)/(dF-1) )
```

```
##          m
## 0.07250409
```

1.2.Historical VaR

```
alpha=.01

(VaR.hist=-quantile(R,alpha))
```

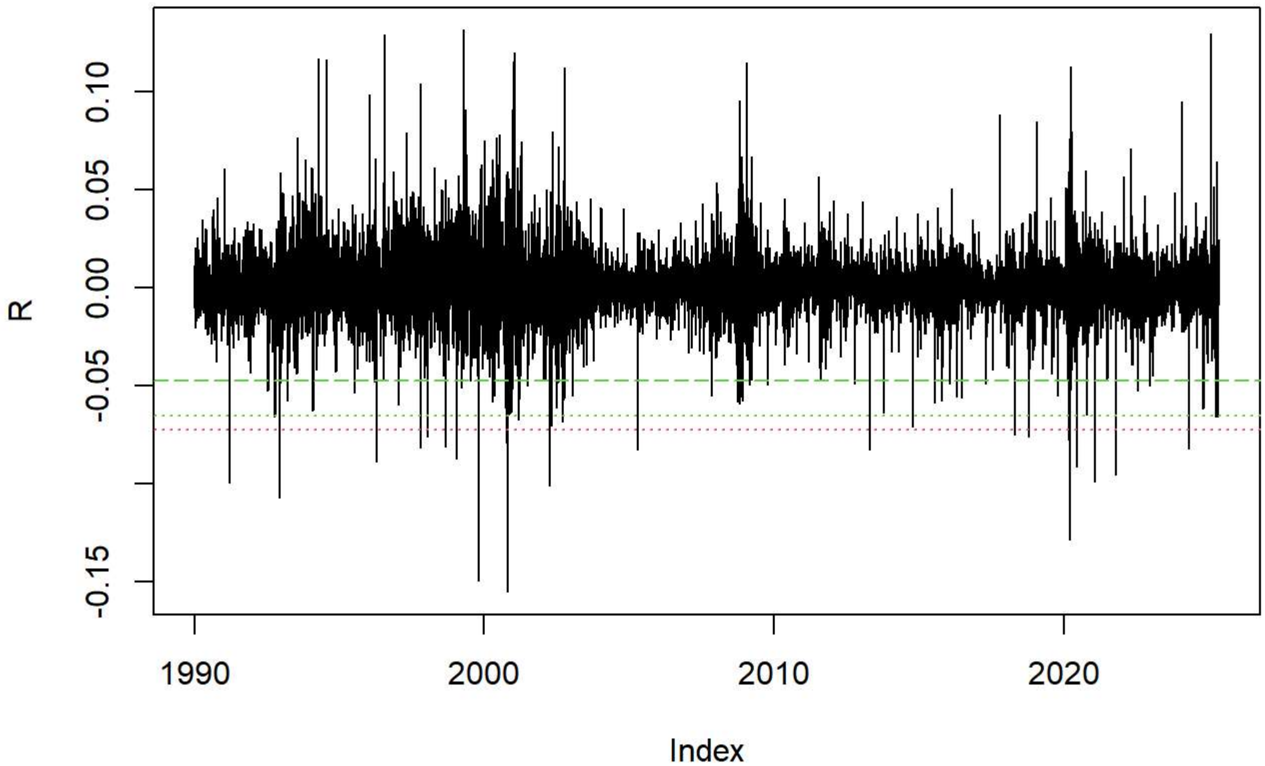
```
##         1%
## 0.04735382
```

```
(CVaR.hist=-mean(R[which(R<(-VaR.hist))]))
```

```
## [1] 0.06550621
```

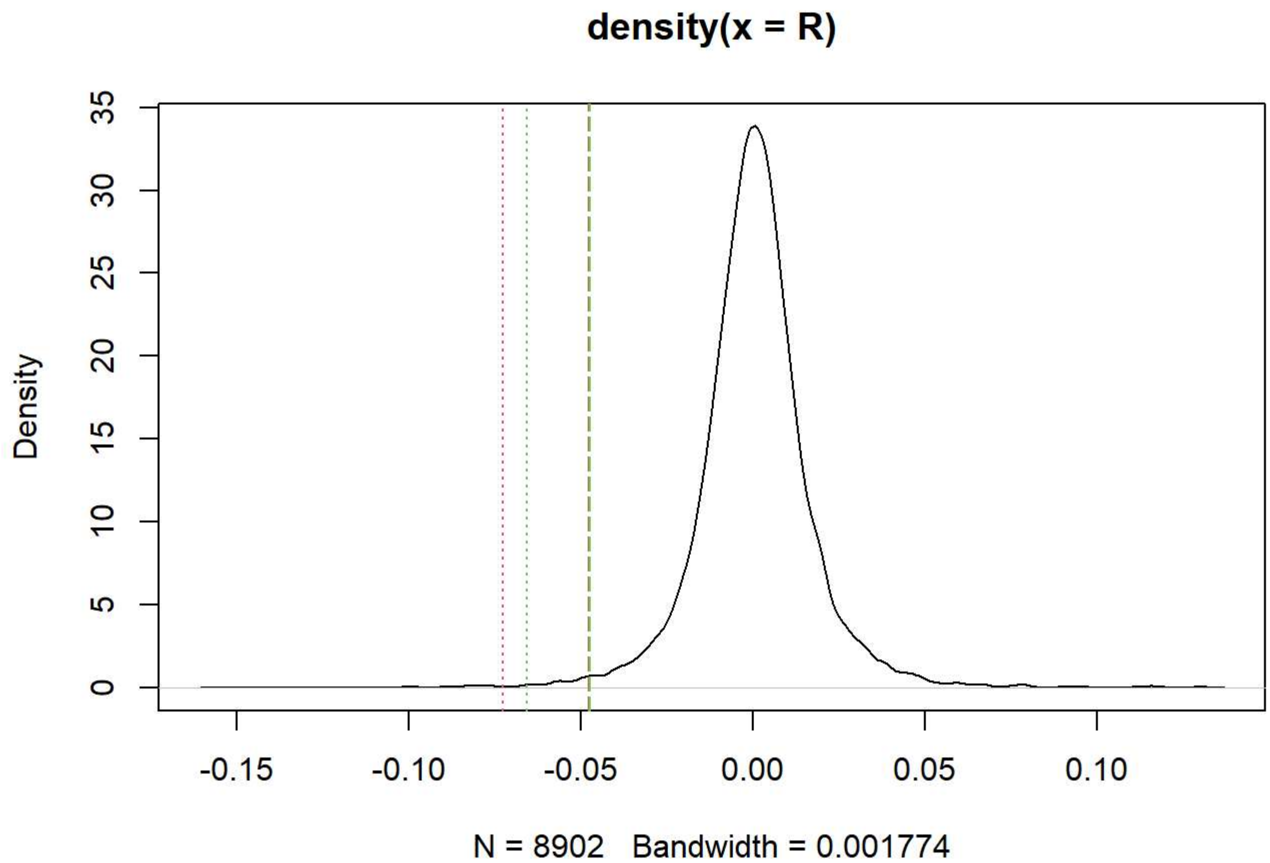
```
plot(R);
abline(h=-VaR.t, col=2, lty=5, lwd=1)
abline(h=-CVaR.t, col=2, lty=3, lwd=1)

abline(h=-VaR.hist, col=3, lty=5, lwd=1)
abline(h=-CVaR.hist, col=3, lty=3, lwd=1)
```




```
plot(density(R))
abline(v=-VaR.t, col=2, lty=5, lwd=1)
abline(v=-CVaR.t, col=2, lty=3, lwd=1)

abline(v=-VaR.hist, col=3, lty=5, lwd=1)
abline(v=-CVaR.hist, col=3, lty=3, lwd=1)
```



If use Loss directly

```
alpha=.01
L=-R
(VaR.hist=quantile(L,1-alpha))
```

```
##        99%
## 0.04735382
```

```
(CVaR.hist=mean(L[which(L>(VaR.hist))]))
```

```
## [1] 0.06550621
```

1.2.1.VaR historical

Calculate the historical Value-at-Risk (VaR) at 95% confidence level. According to this VaR, if you invest $1,000 in stock, the value of your investment after 1 day would exceed which amount with 95% confidence? (Hint: the historical VaR is minus the 5% sample quantile of the returns; use the quantile function in R.

```
CI = 0.95 ; a = 1-CI # CI=1-a
VaR = - quantile( as.numeric(R), probs = a)
# with 95% confidence, investment exceeds:
1000 * (1-VaR)
```

```
##       5%
## 975.1913
```

1.2.2.CVaR historical

Calculate the historical Conditional Value-at-Risk (CVaR). According to this CVaR, if you invest $1,000 in stock, what would be your expected 1-day loss, conditional on the 5% worst cases?

```
CVaR = -mean( as.numeric(R)[ R < -VaR ] )
# expected loss in 5% worst case
1000*CVaR
```

```
## [1] 39.41906
```

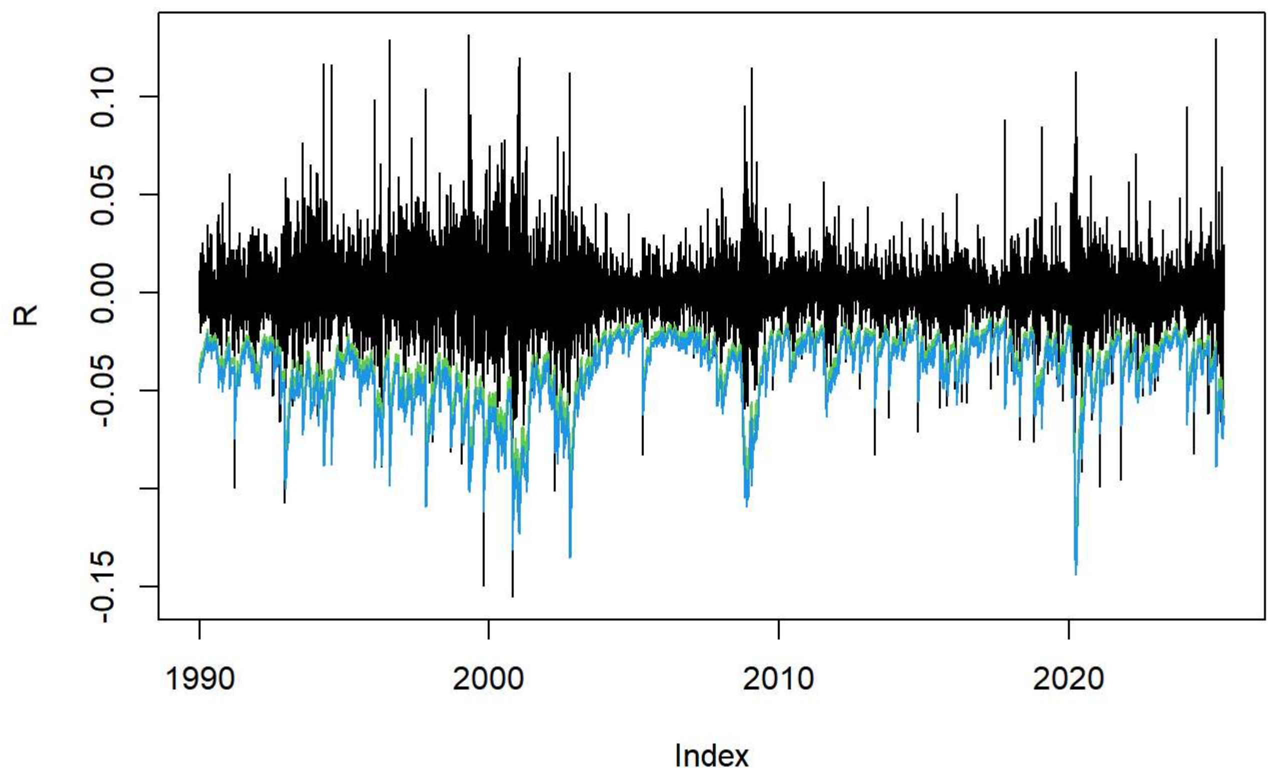
1.3.Time Series - RiskMetrics Model

Simple time series model for return volatility (P29 -30 lecture W6)

```
V2=var(R); lambda=.94
n=length(R); alpha=.01

for(i in 2:n){
  V2[i]=lambda*V2[i-1]+(1-lambda)*R[i-1]^2
}
V2=zoo(V2,index(R)) # fitted RiskMetrics volatilities square
VaRa = sqrt(V2)*qnorm(1-alpha)
CVaRa = sqrt(V2)*dnorm(qnorm(alpha))/alpha

plot(R)
lines(-VaRa, col=3) # lines(sqrt(V2)*qnorm(alpha), col=2)
lines(-CVaRa, col=4)
```



1.4.Time Series - ARCH/GARCH models

Garch(p,q) models.

```
library(fGarch)
```

```
## NOTE: Packages 'fBasics', 'timeDate', and 'timeSeries' are no longer
## attached to the search() path when 'fGarch' is attached.
##
## If needed attach them yourself in your R script by e.g.,
##         require("timeSeries")
```

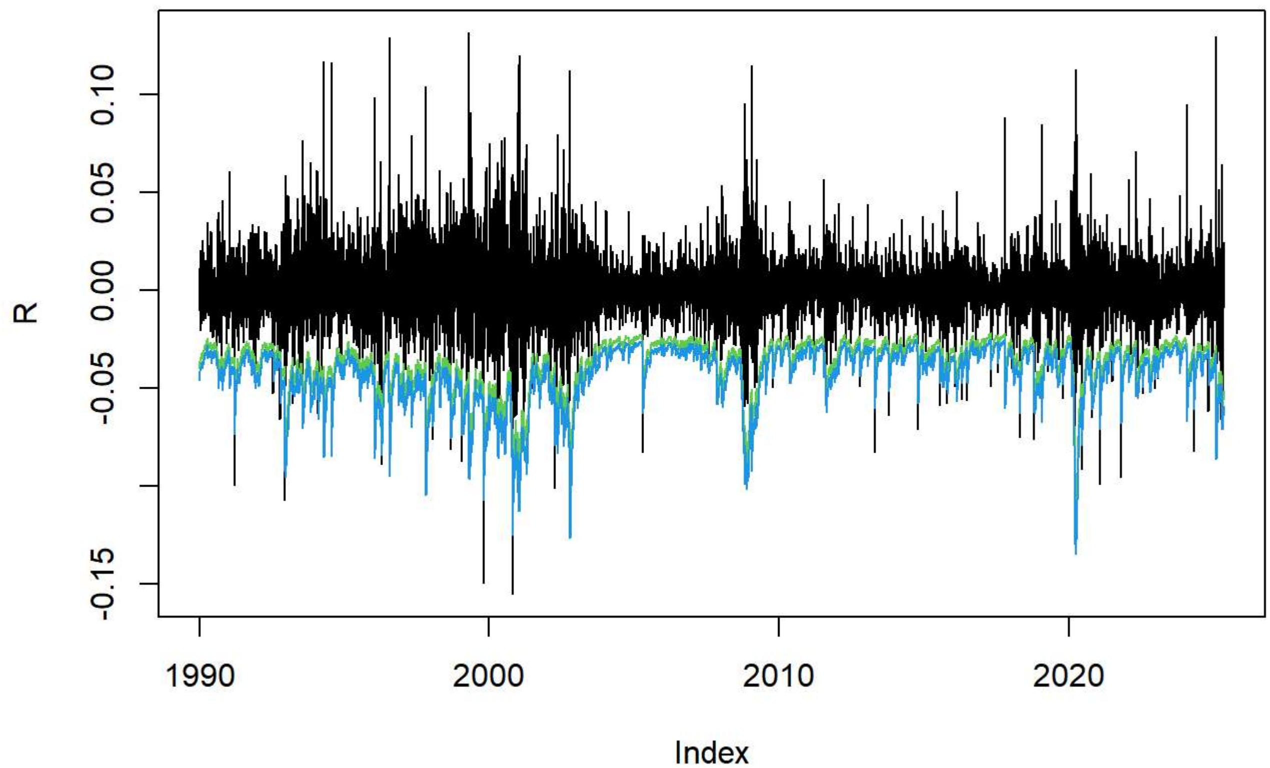
```
V2=var(R);alpha=.01

fit=garchFit( ~garch(1,1), data = R, trace = F) # fit GARCH(1,1) model with Normal errors
summary(fit)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(1, 1), data = R, trace = F)
##
## Mean and Variance Equation:
##  data ~ garch(1, 1)
## <environment: 0x000001b5d47299a0>
##  [data = R]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         mu       omega      alpha1       beta1
## 5.9011e-04  4.7098e-06  5.4721e-02  9.2988e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     5.901e-04   1.499e-04    3.937 8.26e-05 ***
## omega  4.710e-06   9.305e-07    5.062 4.16e-07 ***
## alpha1 5.472e-02   8.117e-03    6.742 1.56e-11 ***
## beta1  9.299e-01   1.054e-02   88.185  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  24467.05    normalized:  2.748489
##
## Description:
##  Fri May  9 17:40:29 2025 by user: crist
##
##
## Standardised Residuals Tests:
##                                   Statistic   p-Value
##  Jarque-Bera Test   R    Chi^2  24308.063979 0.0000000
##  Shapiro-Wilk Test  R    W                NA        NA
##  Ljung-Box Test     R    Q(10)      9.187842 0.5143745
##  Ljung-Box Test     R    Q(15)      9.641501 0.8416643
##  Ljung-Box Test     R    Q(20)     18.249479 0.5709778
##  Ljung-Box Test     R^2  Q(10)      3.872669 0.9529083
##  Ljung-Box Test     R^2  Q(15)      6.326868 0.9737186
##  Ljung-Box Test     R^2  Q(20)     10.516516 0.9578109
##  LM Arch Test       R    TR^2       4.732990 0.9663079
##
## Information Criterion Statistics:
```

```
##       AIC       BIC       SIC      HQIC
## -5.496080 -5.492892 -5.496080 -5.494994
```

```
V=zoo(fit@sigma.t,index(R)) # fitted GARCH volatilities, sigma
VaRa = V*qnorm(1-alpha)
CVaRa = V*dnorm(qnorm(alpha))/alpha
plot(R)
lines(sqrt(V2)*qnorm(alpha), col=2)
lines(-VaRa, col=3) #
lines(-CVaRa, col=4) #
```

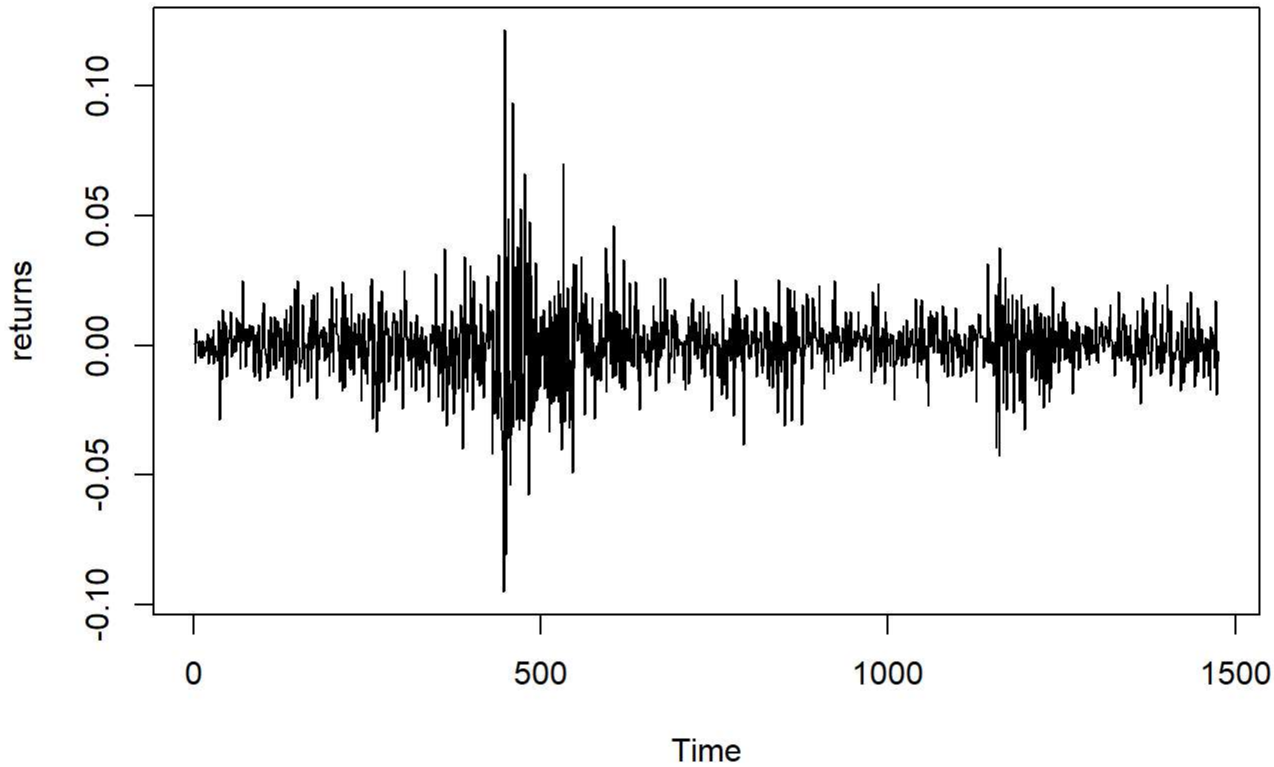


2.Univariate t - VaR and ES

In this section we will compare VaR and ES parametric (unconditional) estimates with those from using ARMA+GARCH (conditional) models. Consider the daily returns for Coca-Cola stock from January 2007 to November 2012.

Consider the daily returns for Coca-Cola stock from January 2007 to November 2012.

```
CokePepsi = read.table("CokePepsi.csv", header=T)
price = CokePepsi[,1]
returns = diff(price)/lag(price)[-1] # Net Return
ts.plot(returns)
```



Assume that the returns are iid and follow a t-distribution. Run the following commands to get parameter estimates in R.

```
S = 4000
alpha = 0.05
library(MASS)
res = fitdistr(returns,'t')
```

```
## Warning in dt((x - m)/s, df, log = TRUE): NaNs produced
```

```
## Warning in log(s): NaNs produced
## Warning in log(s): NaNs produced
## Warning in log(s): NaNs produced
## Warning in log(s): NaNs produced
## Warning in log(s): NaNs produced
## Warning in log(s): NaNs produced
```

```
## Warning in dt((x - m)/s, df, log = TRUE): NaNs produced
```

```
## Warning in log(s): NaNs produced
## Warning in log(s): NaNs produced
## Warning in log(s): NaNs produced
## Warning in log(s): NaNs produced
## Warning in log(s): NaNs produced
## Warning in log(s): NaNs produced
```

```
mu_return = res$estimate['m']
lambda = res$estimate['s'] # Sigma
nu = res$estimate['df']
qt(alpha, df=nu)
```

```
## [1] -2.292268
```

```
dt(qt(alpha, df=nu), df=nu)
```

```
## [1] 0.04799629
```

For an investment of $4,000, what are estimates of $VaR^t(0.05)$ and $ES^t(0.05)$?

$VaR^t(0.05)$ is 75.31 and $ES^t(0.05)$ is 122.1. See the output below.

```
mu_loss=-mu_return

VaR =S*( mu_loss + lambda*qt(1-alpha,df=nu))
options(digits=4);VaR
```

```
##     m
## 75.31
```

```
den = dt(qt(alpha, df=nu), df=nu)
ES = S * (mu_loss + lambda*(den/alpha)* (nu+qt(alpha, df=nu)^2 )/(nu-1))
ES
```

```
##     m
## 122.1
```

3.EVaR

Consider the example with the two risky zero-coupon bonds priced at $95 per $100 face value, where each has 4% default probability independently of the other.

3.1.Entropic Value-at-Risk (EVaR) by optimize()

Calculate the $\alpha = 5\%$ Entropic Value-at-Risk (EVaR) for one of these bonds. Use numeric minimization: `optimize()` in R, to find EVaR.

The (marginal) loss distribution of each bond ($L_{1/2}$) is given by the PMF

$$p_L(\ell) = \mathbb{P}(L = \ell) = \begin{cases} 0.04, & \ell = 95 - 0 = 95 \\ 0.96, & \ell = 95 - 100 = -5 \end{cases}$$

with MGF

$$M_L(z) = \mathbb{E}[e^{zL}] = 0.04e^{95z} + 0.96e^{-5z}$$

The EVaR at α is given by

$$EVaR_\alpha = \inf_{z>0} \left\{ \frac{\ln(M_L(z)/\alpha)}{z} \right\} = \inf_{z>0} \left\{ \frac{\ln\left((0.04e^{95z} + 0.96e^{-5z})/0.05\right)}{z} \right\}$$

```r
fn = function(z){ log((0.04*exp(95*z)+0.96*exp(-5*z))/0.05)/z }
optimise(fn, c(0,1))
```

```
## $minimum
## [1] 0.0669
##
## $objective
## [1] 92.1
```

Running this minimization w.r.t. z in R, we get that the minimum is $EVaR_{0.05}(L) = 92.10402$, occurring at $z = 0.06690106$.

3.2.Subadditive Verify

Calculate the EVaR of a portfolio of two of these bonds, and show that it is subadditive.

The loss distribution for the sum of the two bonds $(L_1 + L_2)$ is

$$p_{L_1+L_2}(\ell) = \mathbb{P}(L_1 + L_2 = \ell) = \begin{cases} (0.04)^2 = 0.0016, & \ell = 95 + 95 = 190 \\ 2(0.96)(0.04) = 0.0768, & \ell = 95 - 5 = 90 \\ (0.96)^2 = 0.9216, & \ell = -5 - 5 = -10 \end{cases}$$

with MGF

$$M_{L_1+L_2}(z) = \mathbb{E}[e^{z(L_1+L_2)}] = 0.0016e^{190z} + 0.0768e^{90z} + 0.9216e^{-10z}$$

Running this minimization w.r.t. z in R:

```r
fn=function(z){log((0.0016*exp(190*z)+0.0768*exp(90*z)+0.9216*exp(-10*z))/0.05)/z}
optimise(fn, c(0,1) )
```

```
## $minimum
## [1] 0.03842
##
## $objective
## [1] 122
```

we get that the minimum is $EVaR_{0.05}(L_1 + L_2) = 122.0294$, occurring at $z = 0.03841828$.

Note that this EVaR is subadditive, since

$$\mathrm{EVaR}_{0.05}(L_1 + L_2) \leq \mathrm{EVaR}_{0.05}(L_1) + \mathrm{EVaR}_{0.05}(L_2)$$

$$\Leftrightarrow 122.0294 \leq 2 \times 92.10402 = 184.208$$