Risk hw3

Jiaqi Li

April 22, 2019

Cowork:Xiangui Mei, Shuyu Dong, Yuhua Deng

Probelm 1

```
options(warn = -1)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

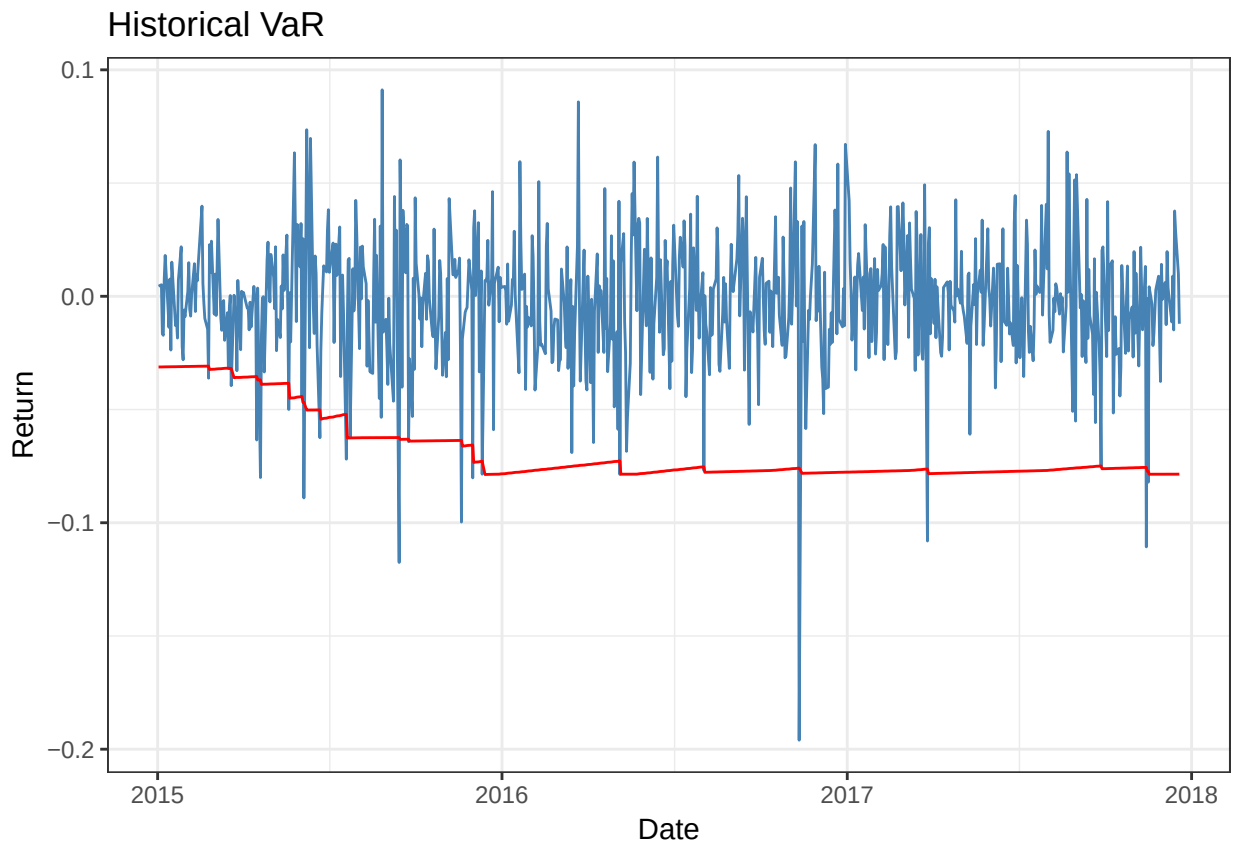
```
library(ggplot2)
library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
#############################################################################
#                             Question 1                                    #
#############################################################################
data = read.csv("hw3_return.csv") %>% as.data.table()
data = data[,Date := as.Date(as.character(Date),format = "%m/%d/%Y")]
data = data[, Year := as.numeric(format(Date, "%Y"))]
setkey(data,Year)
test = data[.(2015:2017)]

n = length(data[.(2014)]$Return)
N = length(data$Return)
c = 0.99
f_VaR = function(c,n,N,data){
  VaR = c()
  for(i in n:N){
    VaR[i-n+1] = -(quantile(data[1:i],1-c,na.rm = T))
  }
  return(VaR)
}
VaR = f_VaR(c,n,N,data$Return)
```

```
test[, VaR := VaR]
qplot(Date,Return,data = test, geom = "line", color = I("steelblue"),
      main = "Historical VaR") +
  theme_bw() + geom_line(aes(y=-VaR), color = I("red"))
```

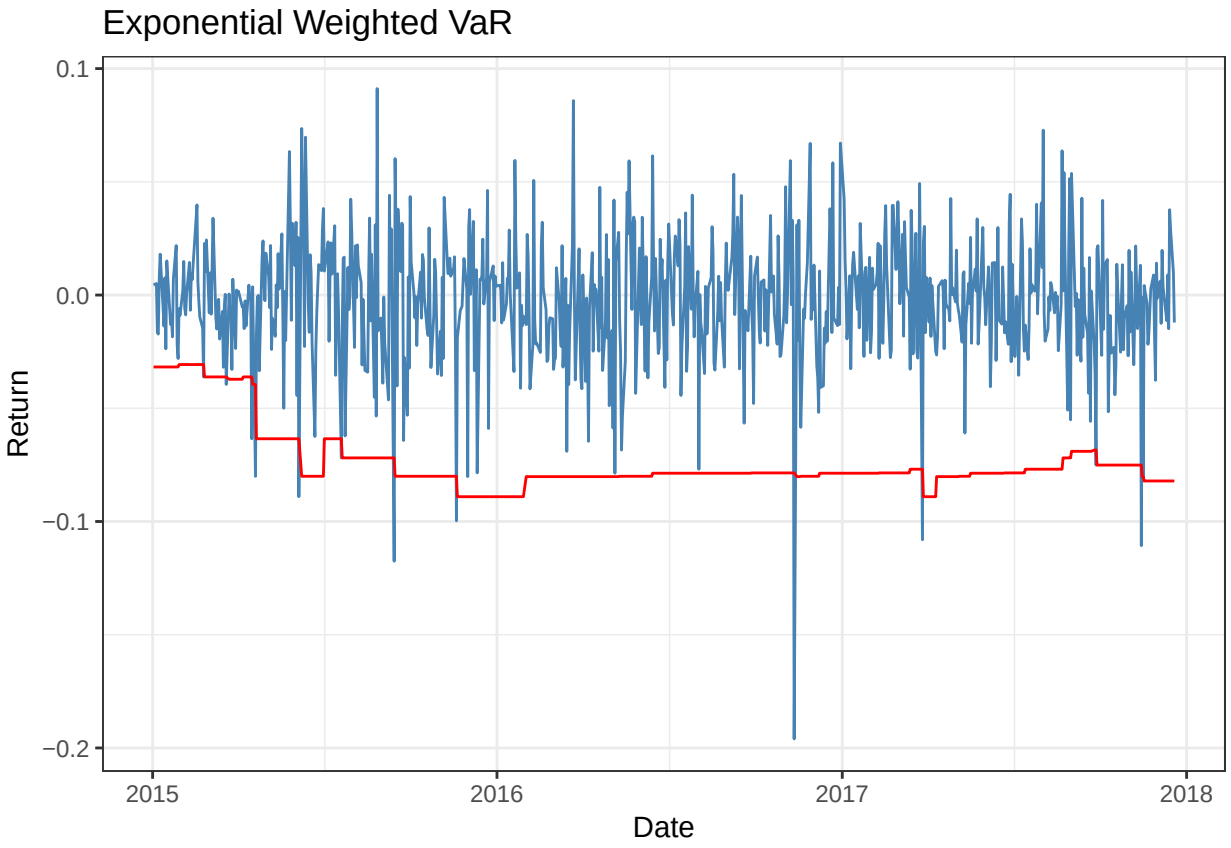



```
exceptions = length(test[Return<(-VaR),]$Return)
exceptions
```

```
## [1] 22
```

```
lambda = 0.995
f_exp_VaR = function(N,c,lambda,data){
  weights = c()
  for(i in 1:N){
    weights[i] = lambda^(N-i)*(1-lambda)/(1-lambda^(N))
  }
  test = as.data.table(cbind(Return = data$Return[1:N],weights))
  setorder(test,Return)
  temp = cumsum(test$weights)
  test[,cum_sum := temp]
  VaR_exp = test[cum_sum > 1-c,]$Return[1]
  return(VaR_exp)
}

VaR_exp = c()
for(i in n:N){
  VaR_exp[i-n+1] = f_exp_VaR(i,c,lambda,data)
```

```
}
test[,VaR_exp := VaR_exp]

qplot(Date,Return,data = test, geom = "line", color = I("steelblue"),
      main = "Exponential Weighted VaR") +
  geom_line(aes(y=VaR_exp), color = I("red")) + theme_bw()
```

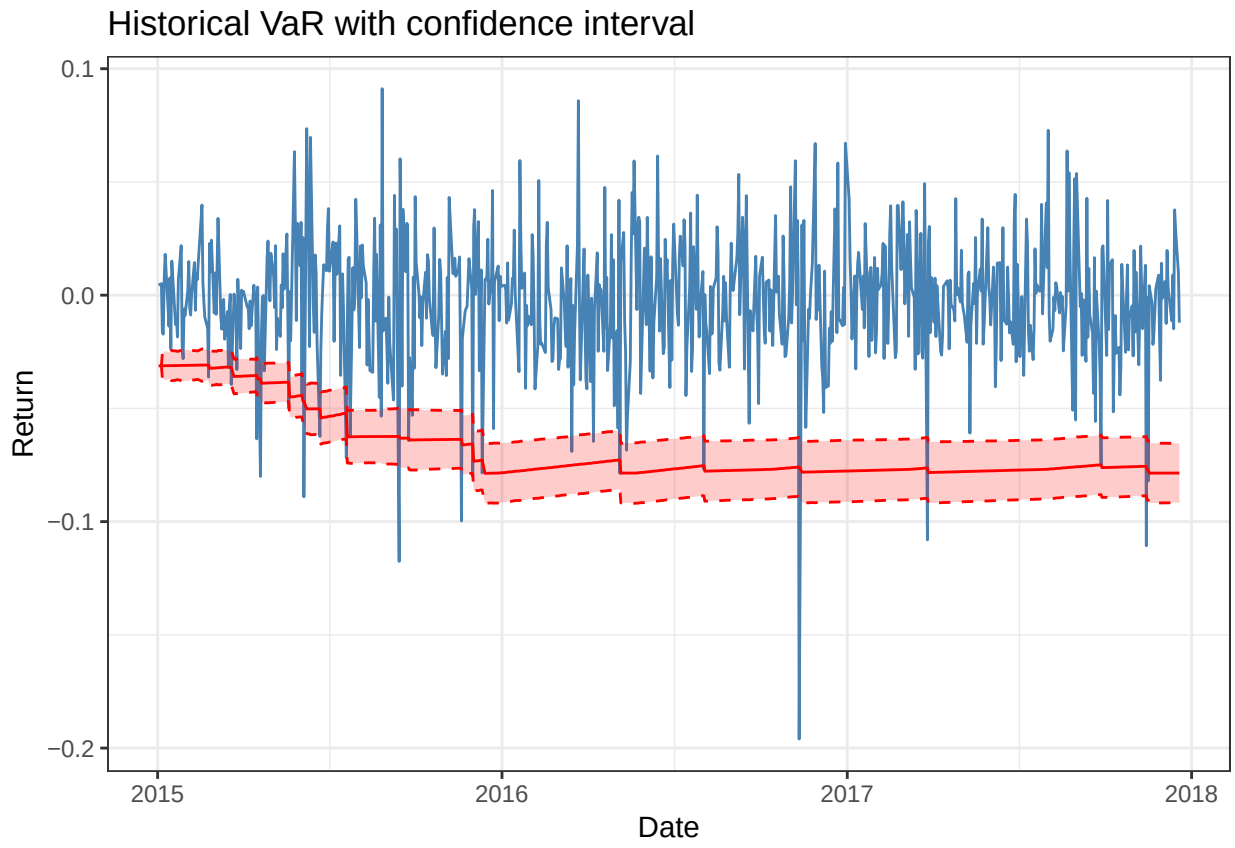



```
exceptions = length(test[Return<VaR_exp,]$Return)
exceptions
```

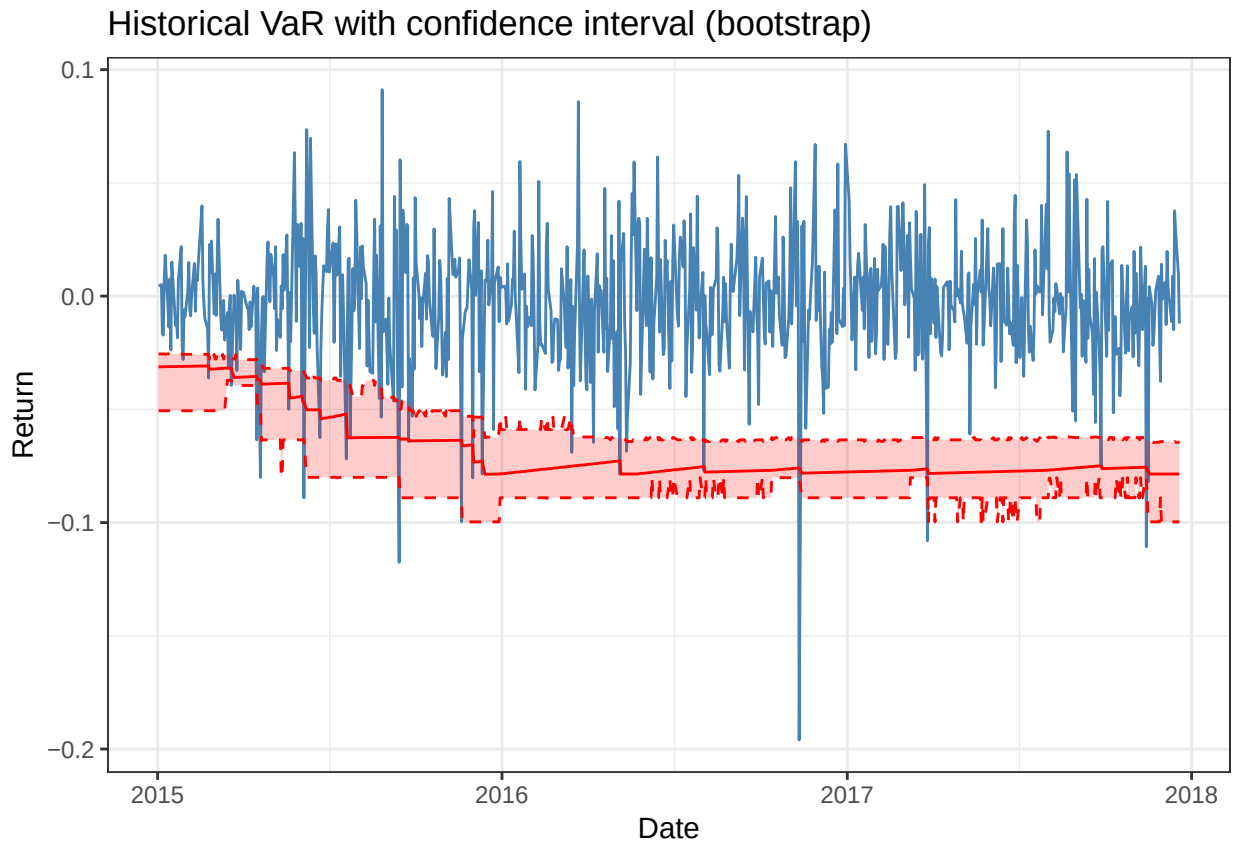
```
## [1] 13
```

```
#########################################################################
#                              Question 2                               #
#########################################################################

mu = c();std = c()
for(i in 1:length(test$Return)){
  mu[i] = mean(test$Return[1:i])
  std[i] = sd(test$Return[1:i])
}
x = mu-2.33*std
fx = dnorm(x,mean = mu, sd = std)
StdDev = (1/fx)*sqrt((0.99*0.01)/n)
CI_left = -test$VaR-abs(qnorm(0.025,0,1))*StdDev
CI_right = -test$VaR+abs(qnorm(0.025,0,1))*StdDev
qplot(Date,Return,data = test, geom = "line", color = I("steelblue"),
```

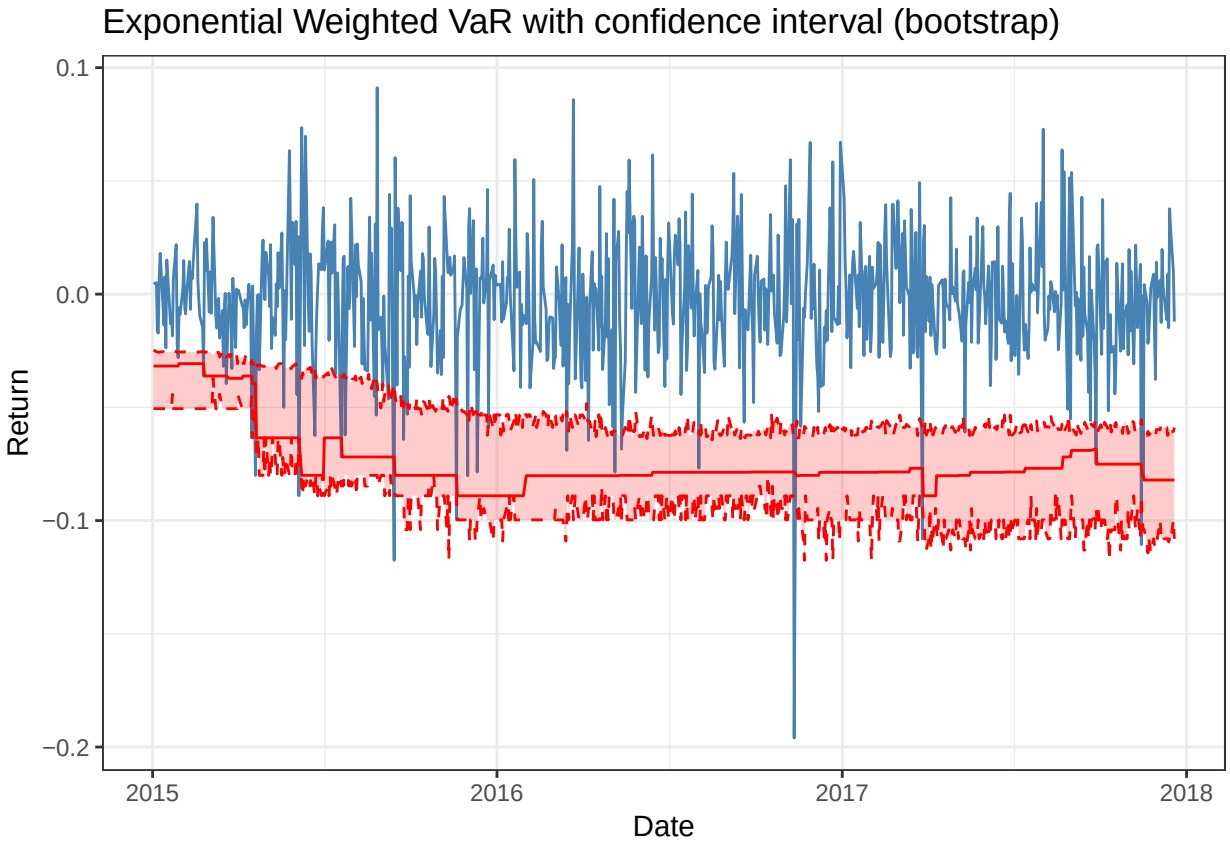
```
      main = "Historical VaR with confidence interval") +
  theme_bw() + geom_line(aes(y=-VaR), color = I("red"))+
  geom_line(aes(y=CI_left), color = I("red"),linetype = 2) +
  geom_line(aes(y=CI_right), color = I("red"),linetype = 2) +
  geom_ribbon(aes(ymin=CI_left,ymax=CI_right), fill="red", alpha="0.2")
```



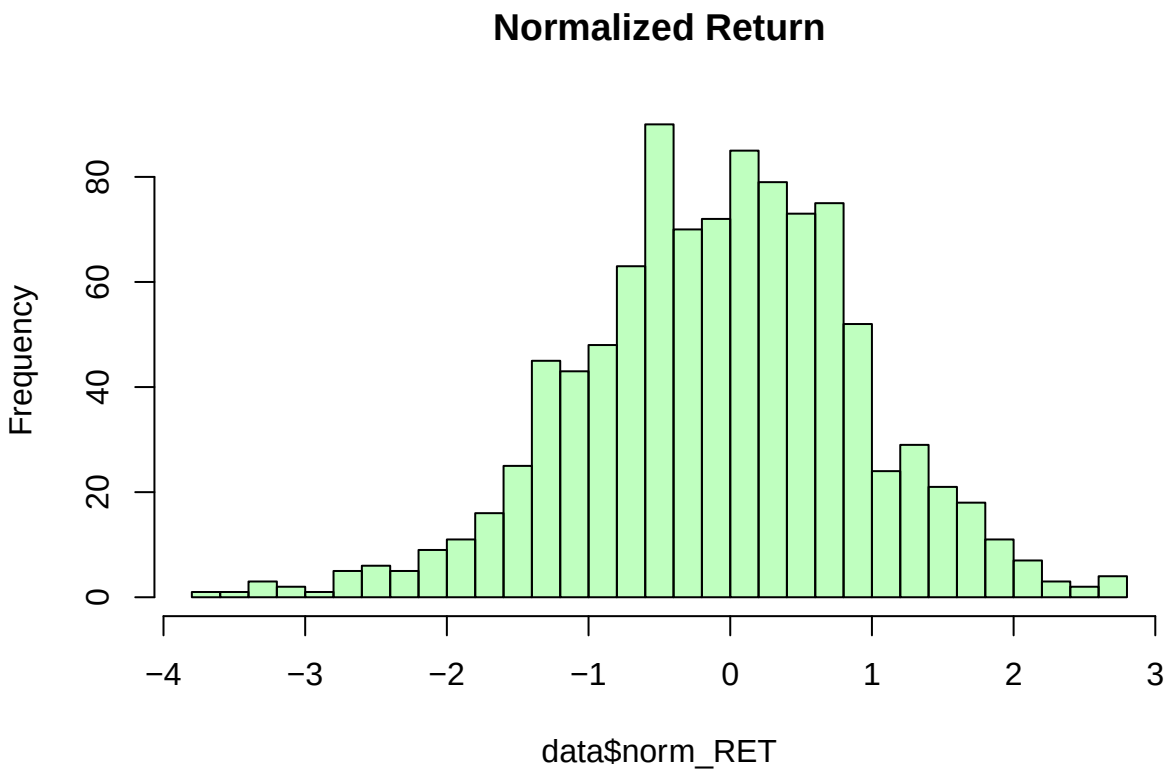

```
bootstrap_VaR_H = c()
bootstrap_VaR_L = c()
for(i in 1:(N-n)){
  bootstrap_VaR = c()
  for(j in 1:1000){
    temp_bootstrap = sample(data$Return[1:(n+i-1)],
                            size = n+i-1, replace = T)
    bootstrap_VaR[j] = quantile(temp_bootstrap,0.01,type = 1)
  }
  bootstrap_VaR_H[i] = quantile(bootstrap_VaR,0.975)
  bootstrap_VaR_L[i] = quantile(bootstrap_VaR,0.025)
}

qplot(Date,Return,data = test, geom = "line", color = I("steelblue"),
      main = "Historical VaR with confidence interval (bootstrap)") +
  theme_bw() + geom_line(aes(y=-VaR), color = I("red"))+
  geom_line(aes(y=bootstrap_VaR_L), color = I("red"),linetype = 2) +
  geom_line(aes(y=bootstrap_VaR_H), color = I("red"),linetype = 2) +
  geom_ribbon(aes(ymin=bootstrap_VaR_L,ymax=bootstrap_VaR_H), fill="red", alpha="0.2")
```



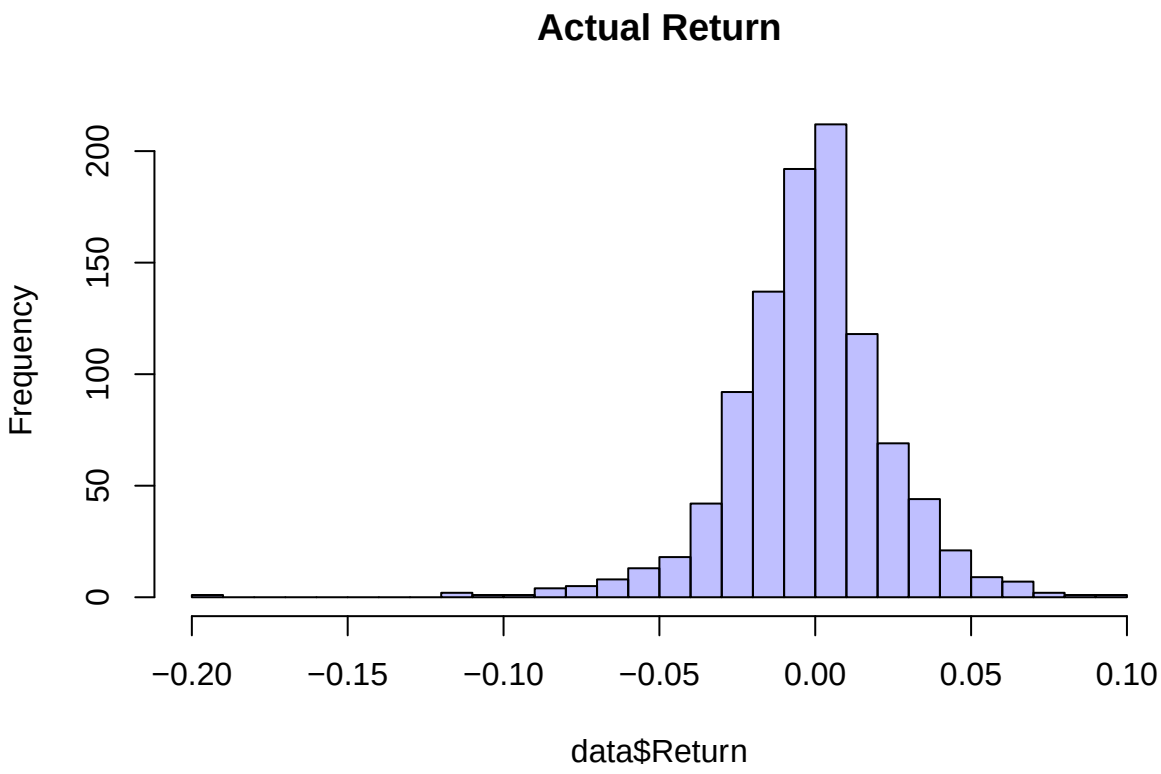

```
bootstrap_expVaR_H = c()
bootstrap_expVaR_L = c()
for(i in 1:(N-n)){
  bootstrap_expVaR = c()
  for(j in 1:100){
    temp_bootstrap_exp = sample(data$Return[1:(n+i-1)],
                                size = n+i-1, replace = T)
    weights = lambda^(n+i-1-c(1:(n+i-1)))*(1-lambda)/(1-lambda^(n+i-1))
    temp_exp = cbind(temp_bootstrap_exp,weights) %>% as.data.table()
    setorder(temp_exp,temp_bootstrap_exp)
    x = cumsum(temp_exp$weights)
    temp_exp[,cum_sum := x]
    bootstrap_expVaR[j] = temp_exp[cum_sum>1-c,]$temp_bootstrap_exp[1]
  }
  bootstrap_expVaR_H[i] = quantile(bootstrap_expVaR,0.975)
  bootstrap_expVaR_L[i] = quantile(bootstrap_expVaR,0.025)
}

qplot(Date,Return,data = test, geom = "line", color = I("steelblue"),
      main = "Exponential Weighted VaR with confidence interval (bootstrap)") +
  theme_bw() + geom_line(aes(y=VaR_exp), color = I("red"))+
  geom_line(aes(y=bootstrap_expVaR_L), color = I("red"),linetype = 2) +
  geom_line(aes(y=bootstrap_expVaR_H), color = I("red"),linetype = 2) +
  geom_ribbon(aes(ymin=bootstrap_expVaR_L,ymax=bootstrap_expVaR_H),
              fill="red", alpha="0.2")
```



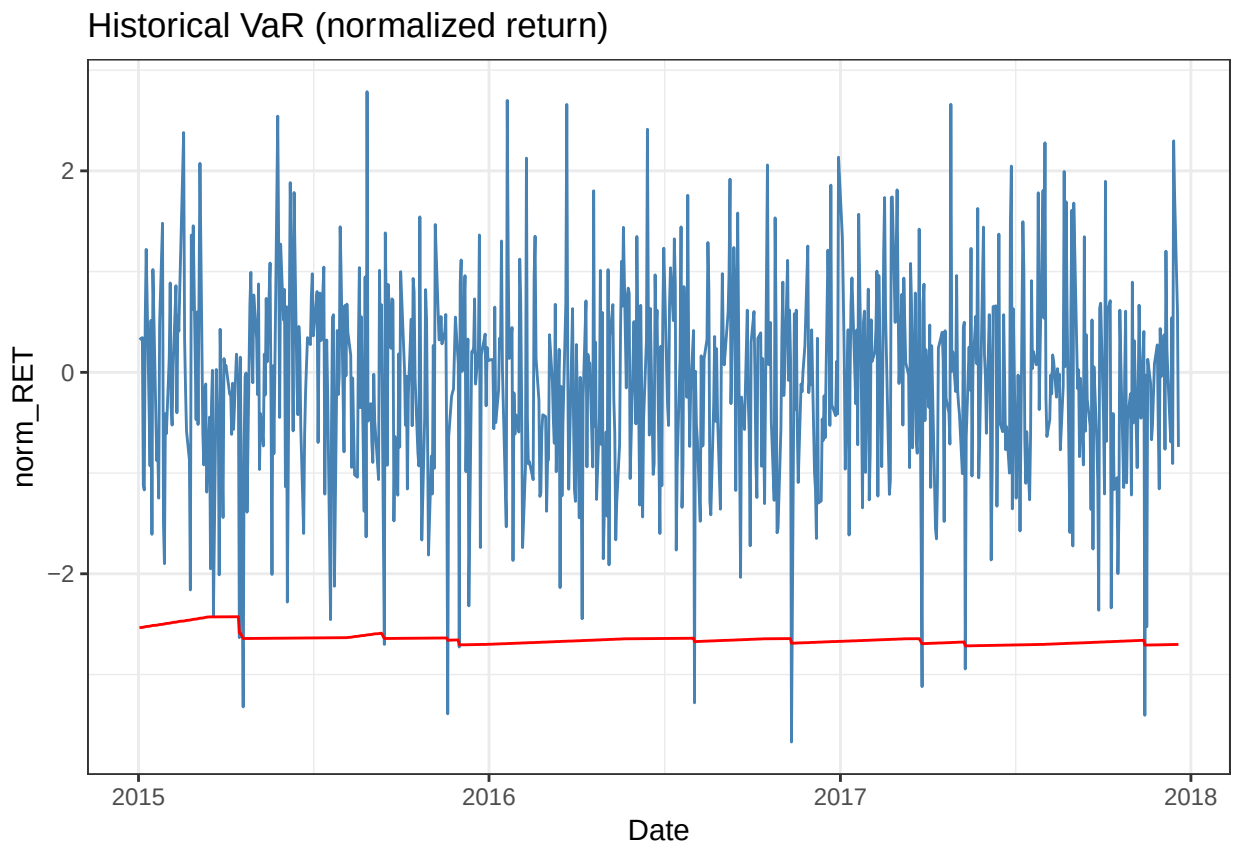

```
#########################################################################
#                            Question 3                                 #
#########################################################################
data[, `:=` (Year = as.numeric(format(Date,"%Y")), Month = as.numeric(format(Date,"%m")))]
data[,volatility := sd(Return), by = list(Year,Month)]
data[,volatility_shift := shift(volatility)]
data[,norm_RET := Return/volatility_shift]
hist(data$norm_RET,breaks = 30,col=rgb(0,1,0,1/4),
     main = "Normalized Return")
```




```
hist(data$Return,breaks = 30,col=rgb(0,0,1,1/4),
     main = "Actual Return")
```



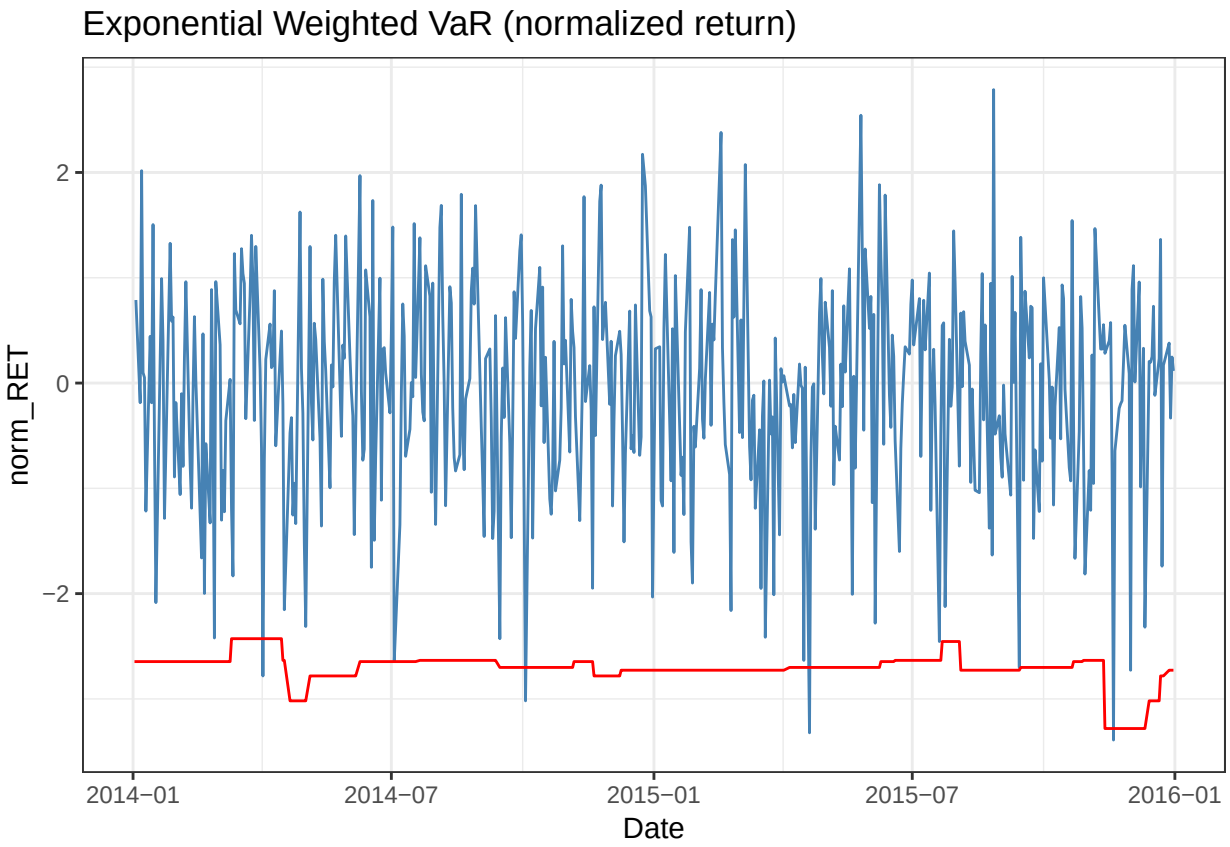

```
#########################################################################
#                            Question 4                                 #
#########################################################################
VaR_new = f_VaR(c,n,N,data$norm_RET)[-1]
setkey(data,Year)
test = na.omit(data[.(2015:2017)])
test[, VaR_new := VaR_new]
qplot(Date,norm_RET,data = test, geom = "line", color = I("steelblue"),
      main = "Historical VaR (normalized return)") +
  theme_bw() + geom_line(aes(y=-VaR_new), color = I("red"))
```




```
exceptions = length(test[norm_RET<(-VaR_new),]$Return)
exceptions
```

```
## [1] 10
```

```
test = data[.(2014:2015)]
f_exp_VaR = function(N,c,lambda,data){
  weights = c()
  for(i in 1:N){
    weights[i] = lambda^(N-i)*(1-lambda)/(1-lambda^(N))
  }
  test = as.data.table(cbind(Return = data$norm_RET[1:N],weights))
  setorder(test,Return)
  temp = cumsum(test$weights)
  test[,cum_sum := temp]
  VaR_exp = test[cum_sum > 1-c,]$Return[1]
  return(VaR_exp)
}
VaR_exp = c()
for(i in n:N){
  VaR_exp[i-n+1] = f_exp_VaR(i,c,lambda,data)
}
test[,VaR_exp := VaR_exp]

qplot(Date,norm_RET,data = test, geom = "line", color = I("steelblue"),
      main = "Exponential Weighted VaR (normalized return)") +
  geom_line(aes(y=test$VaR_exp), color = I("red")) + theme_bw()
```




```
exceptions = length(test[norm_RET<VaR_exp,]$Return)
exceptions
```

```
## [1] 4
```

```
#############################################################################
#                              Question 5                                   #
#############################################################################
```

Based on question 1, since the number of exceptions of the exponential weighted VaR is less than that of the historical VaR, we should choose to use exponential weighted VaR instead of the historical VaR. Also, the normalized returns looks much better than the actual returns because the shape looks normal, and the number of exceptions of both historical VaR and exponential weighted VaR are both reduced significantly. Thus, we should normalized the returns first before analyzing the VaR.

Probelm 2

1

Assume that there are at most 2 people borned in same year, then there are 2 people borned in the first year, 2 people borned in the second year, and 2 people borned in the third year. Then, there are still 2 people left, which means it is impossible to have at most 2 people borned in the same year if there are 8 people borned in a 3-year time period. Thus, at least 3 of the 8 people are born within the same 1-year period.

2

Based on the question, the 5-day 98% VaR is $-\sigma_5 \times \Phi^{-1}(0.02) = 10$, $\sigma_5 = \frac{-10}{-2.053749} = 4.87$ million. Then, compute 10-day 99% VaR: $-\sigma_{10} \times \Phi^{-1}(0.01) = -\sqrt{2}\sigma_5 \times \Phi^{-1}(0.01) = -\sqrt{2} \times 4.87 \times (-2.33) = 16.047$ million.

3

Assume we want 99% VaR and there are 20 trading days evry month.
P(observing more than 1 exception) $= 1 - P$(observing no exception + observing 1 exception)
P(observing more than 1 exception) $= 1 - \sum_{k=0}^{1} \frac{20!}{k!(20-k)!}(1-c)^k c^{n-k}$ where $c = 0.99$

```
1-pbinom(1,20,1-c)
```

```
## [1] 0.01685934
```

Thus, the probability is 0.01685934.

4

```
library(quantmod)
```

```
## Loading required package: xts
```

```
## Loading required package: zoo
```

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

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

```
##
## Attaching package: 'xts'
```

```
## The following objects are masked from 'package:data.table':
##
##     first, last
```

```
## The following objects are masked from 'package:dplyr':
##
##     first, last
```

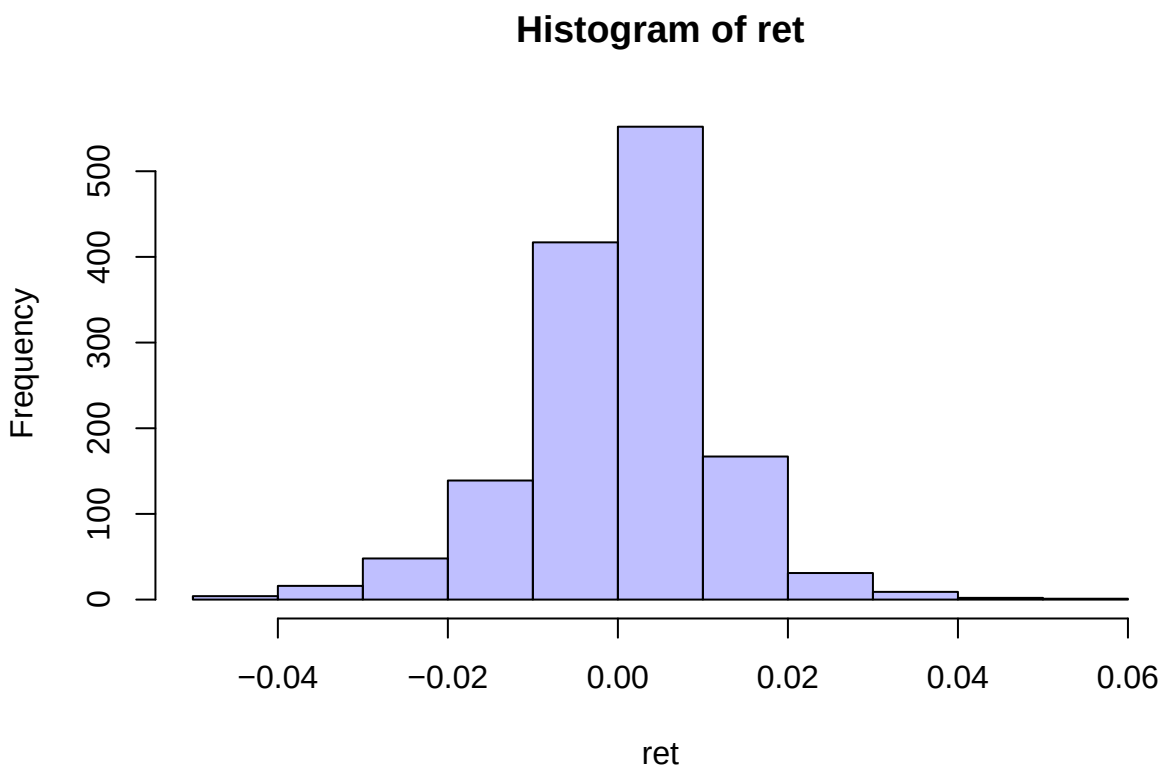
```
## Loading required package: TTR
```

```
## Version 0.4-0 included new data defaults. See ?getSymbols.
```

```
getSymbols("IPO")
```

```
## 'getSymbols' currently uses auto.assign=TRUE by default, but will
## use auto.assign=FALSE in 0.5-0. You will still be able to use
## 'loadSymbols' to automatically load data. getOption("getSymbols.env")
## and getOption("getSymbols.auto.assign") will still be checked for
## alternate defaults.
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.warning4.0"=FALSE). See ?getSymbols for details.
##
## WARNING: There have been significant changes to Yahoo Finance data.
## Please see the Warning section of '?getSymbols.yahoo' for details.
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.yahoo.warning"=FALSE).
## [1] "IPO"
```

```
ret = dailyReturn(IPO)
hist(ret,col = rgb(0,0,1,1/4))
```




```
VaR = quantile(ret,1-c)
VaR*sqrt(252)
```

```
##         1%
## -0.5031152
```

I used a ETF, which symbol is “IPO”, the annualized VaR is 0.5.
Compared to this “IPO”, Uber should have a slightly lower VaR.