STAT509-001-HW3-Xinye X

Q1

Column seven of the data set RecentFord.csv on the book's web site contains Ford daily closing prices, adjusted for splits and dividends, for the years 2009–2013. Repeat Problem 1 using these more recent returns.

(a) For Ford returns, the sample mean = 0.001842, sample median = 0.000815, and standard deviation = 0.02641701

```
fin_dat <- read.csv('/Users/xuxinye/Desktop/Umich classes/STATS 509/My HWs/RecentFord.csv')
ret <- diff(fin_dat$Adj.Close) / fin_dat$Adj.Close[-1258] # relative return, not lgo
summary(ret)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.174825 -0.011542  0.000815  0.001842  0.014167  0.160131
```

```
mu <- mean(ret)
sigma <- sd(ret)
mu; sigma
```

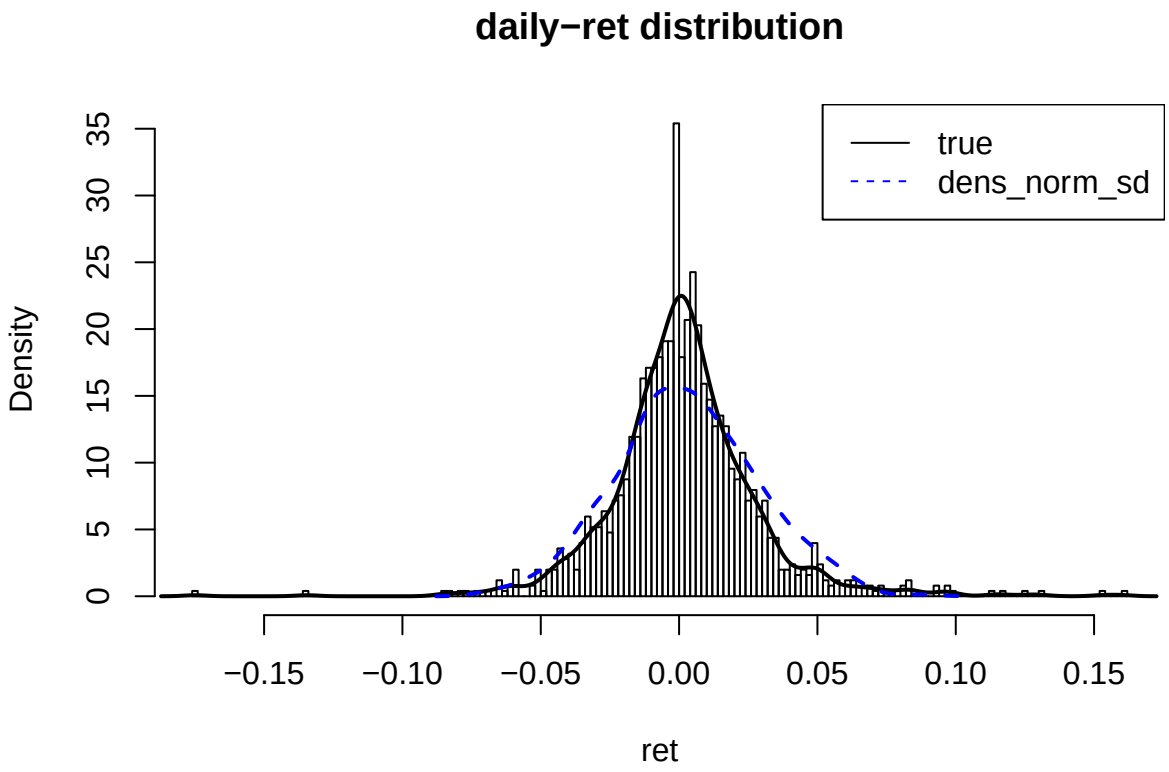
```
## [1] 0.001842084
```

```
## [1] 0.02641701
```

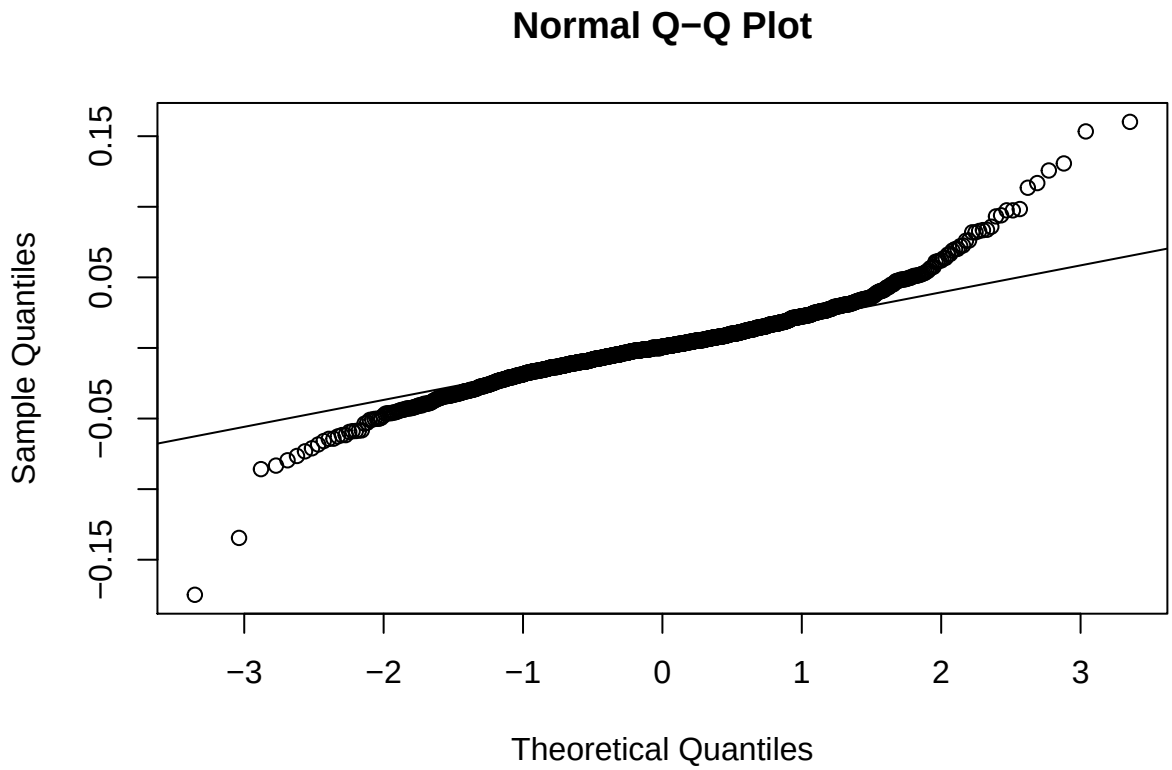
(b) Create a normal plot of the Ford returns. Do the returns look nor- mally distributed? If not, how do they differ from being normally distributed?

With the histigram plot and simulated nomal plot by sampling mean and std dev, it suggests that the peaks of the return are much higher than nomal and it seems true return dist has heavy tail. From the QQ plot, we can also notice that the reuturn plot has heavier tail than normal.

```
set.seed(123)
xnorm <- rnorm(500,mu,sigma)
dens_norm_sd = density(xnorm,kernel=c("gaussian")) # simulated pdf
hist(ret, breaks = 200, main='daily-ret distribution', freq = FALSE)
lines(density(ret),lty=1,lwd=2)
lines(dens_norm_sd,lty=2,lwd=2, col = 'blue')
legend('topright', c('true', 'dens_norm_sd'),lty = c(1, 2), col = c('black', 'blue'))
```




```
qqnorm(ret)
qqline(ret)
```



(c) Test for normality using the Shapiro–Wilk test? What is the p-value? Can you reject the null hypothesis of a normal distribution at 0.01?

The null-hypothesis of Shapiro–Wilk test is that the population is normally distributed. As the p-value < 2.2e-16 is very samll and less than 0.01, then the null hypothesis is rejected.

```
shapiro.test(ret)
```

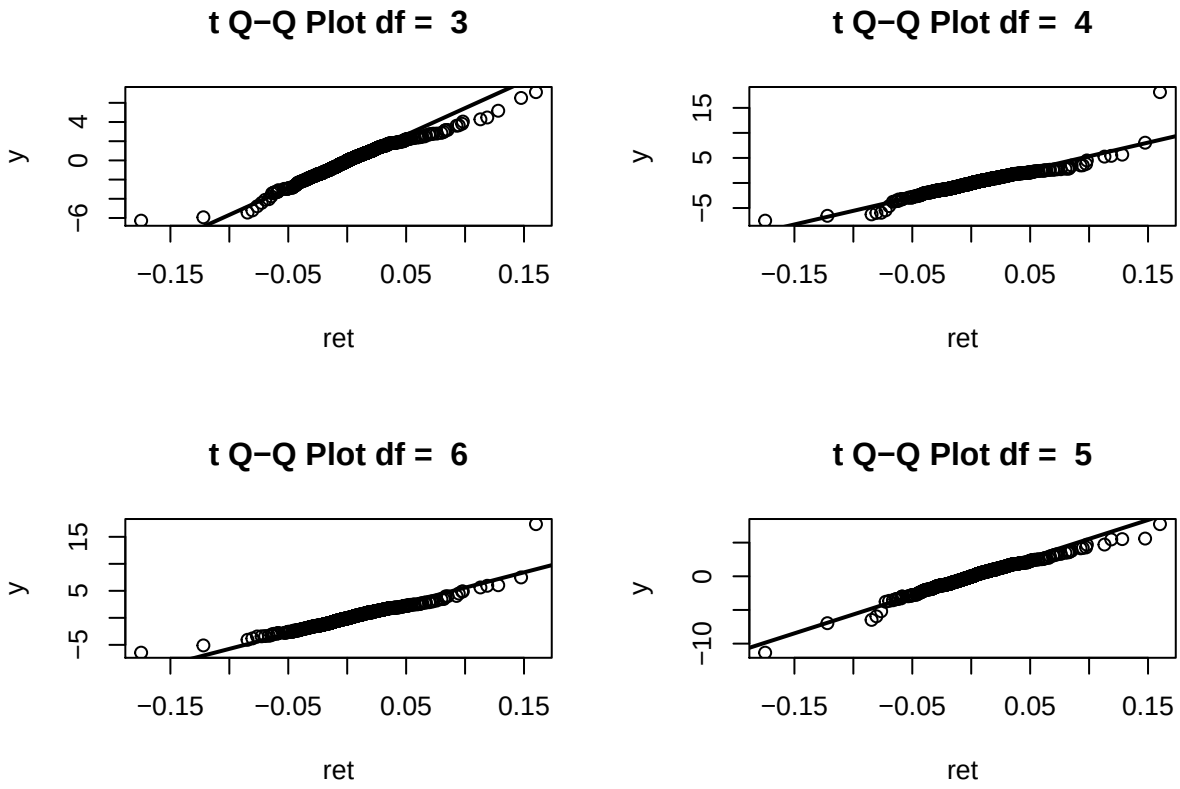
```
##
##  Shapiro-Wilk normality test
##
## data:  ret
## W = 0.93151, p-value < 2.2e-16
```

(d) Create several t-plots of the Ford returns using a number of choices of the degrees of freedom parameter (df). What value of df gives a plot that is as linear as possible? The returns include the return on Black Monday, October 19, 1987. Discuss whether or not to ignore that return when looking for the best choices of df.

From the plot below, df = 3 seems to have a better linear plot. And we also delete the minimal return date, then plot the t qq plot with degree = 4 has a better linear trend. They suggest that with degree of 4, dataset exculuding minimal reurn has more points in the linear line, which mens a better simulation by t-distribution. So, we can ignore the return for the day with large loss.

It happened on 5/12/2009, Ford plans sale of 300 million shares. Based on Ford's closing stock price of 6.08 on Monday, the common stock offering would raise 1.82 billion, although a public offering can cause a stock's price to fall because of shareholder dilution. News: https://dealbook.nytimes.com/2009/05/11/ford-plans-public-offering-of-300-million-shares/

```
cal <- function(x, y) {
  xx = quantile(ret, c(0.25, 0.75))
  yy = quantile(y, c(0.25, 0.75))
  slope = (yy[2] - yy[1]) / (xx[2] - xx[1])
  inter = yy[1] - slope*xx[1]
  return (c(slope,inter))
}
par(mfrow=c(2,2))
degree = c(3,4,6,5,10,20)
for(j in 1:6){
  y <- rt(1000, degree)
  qqplot(ret, y, main = paste('t Q-Q Plot df = ', degree[j]))
  abline(cal(ret, y)[2], cal(ret, y)[1], lwd = 2 )
}
```



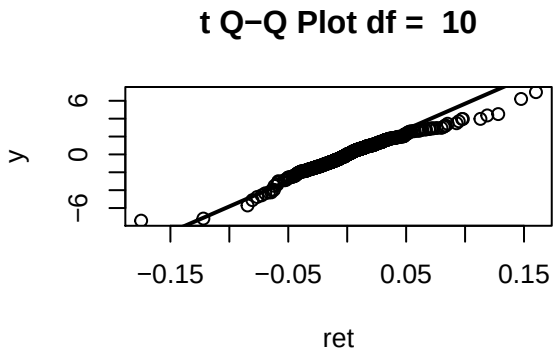

```
# Book page 63
# qqline(y, distribution = function(p) qt(p, df = degree), col = 2)

# find minimal return point
spc <- which(ret == min(ret)) # 89
ret[spc]
```

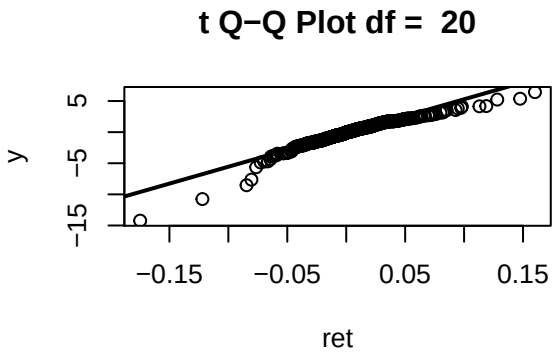
```
## [1] -0.1748252
```

```
fin_dat[(spc-1):(spc+1), ]
```

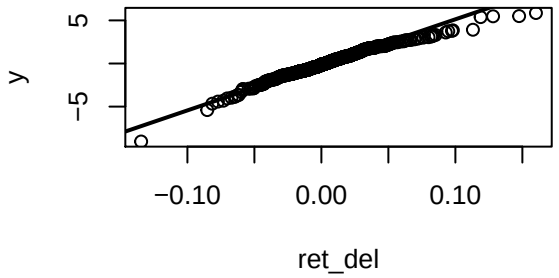
```
##         Date Open High  Low Close    Volume Adj.Close
## 88  5/8/2009 6.19 6.30 6.03  6.24  63884500      5.87
## 89 5/11/2009 6.09 6.26 5.88  6.08  64849100      5.72
## 90 5/12/2009 5.74 5.77 5.01  5.01 217074100      4.72
```

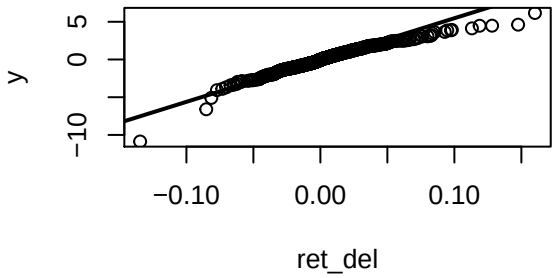
```
ret_del <- ret[-which(ret == min(ret))]

# delete the min return date, dataset: ret_del
par(mfrow=c(2,2))
```

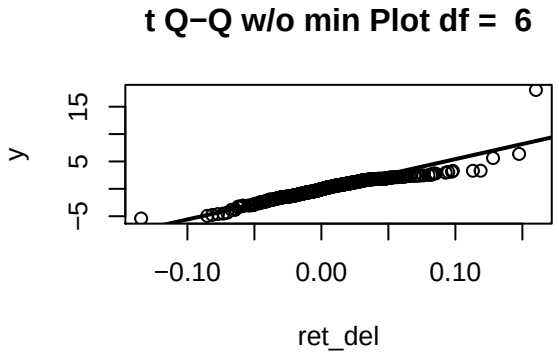


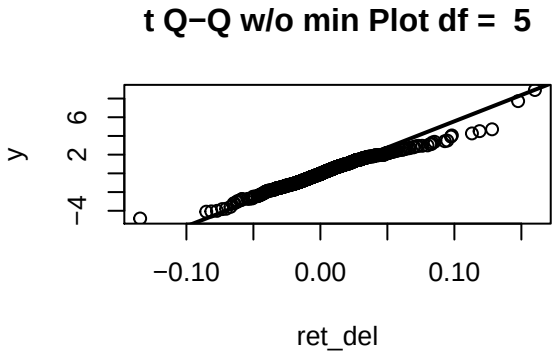


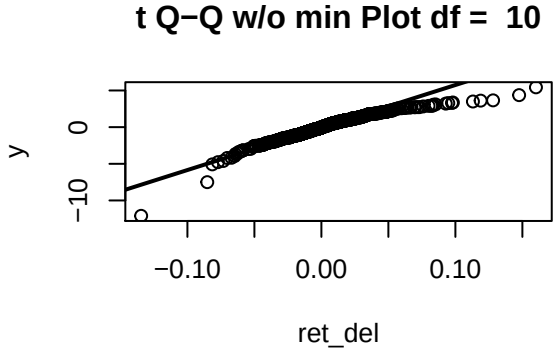

```
degree = c(3,4,6,5,10,20)
for(j in 1:6){
  y <- rt(1000, degree)
  qqplot(ret_del, y, main = paste('t Q-Q w/o min Plot df = ', degree[j]))
  abline(cal(ret, y)[2], cal(ret, y)[1], lwd = 2 )
}
```

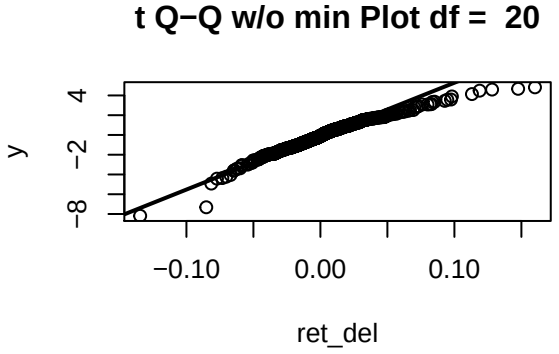












(e) Find the standard error of the sample median using formula (4.3) with the sample median as the estimate of

$$F1 * (0.5)$$

and a KDE to estimate f. Is the standard error of the sample median larger or smaller than the standard error of the sample mean?

standard error of the sample median = 0.0006269494, standard error of the sample mean = sd/ sqr(n) = 0.0007451024

```
# standard error of the sample median
med = median(ret)
n = length(ret) # # a large number, such as probabilty dateset length
d = density(ret)
KDE = approx(d$x, d$y, med, method = "linear")
q = 0.5
sqrt((q * (1-q)) / (n * KDE$y^2 ))   # formula 4.3
```

```
## [1] 0.0006269494
```

```
# standard error of the sample mean
sd(ret)/sqrt(length(ret))
```

```
## [1] 0.0007451024
```

Q2

(a) For the Ford daily returns in Problem 1., fit a Generalized Pareto Distribution (GPD for short) to the upper tail of the negative log returns and give detailed plots of the fit (along the lines of what is generated in class) and discuss your results.

Tail estimates for log loss returns with thresholds 0.02, the plot suggests that shape estimate has smallest value when reachs 600. Then, we beginning from threshold .02 for the loss, and find the estimates of shape (xi) = 0.07434931 and the scale = 0.01620256 From the last tail plot, it indicates a good approximation by using GPD although it seems there are two outliers.

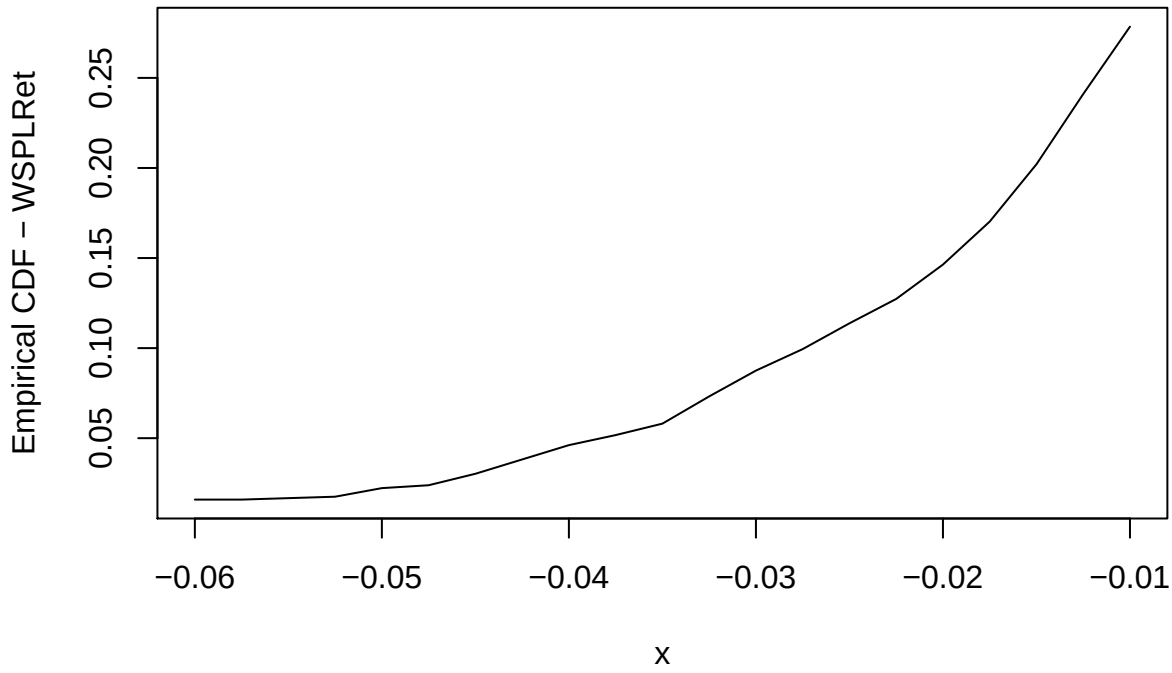
```
WSPLRet = diff(log(fin_dat$Adj.Close))
library(POT)
library(evir)
```

```
## Warning: package 'evir' was built under R version 3.4.4
```

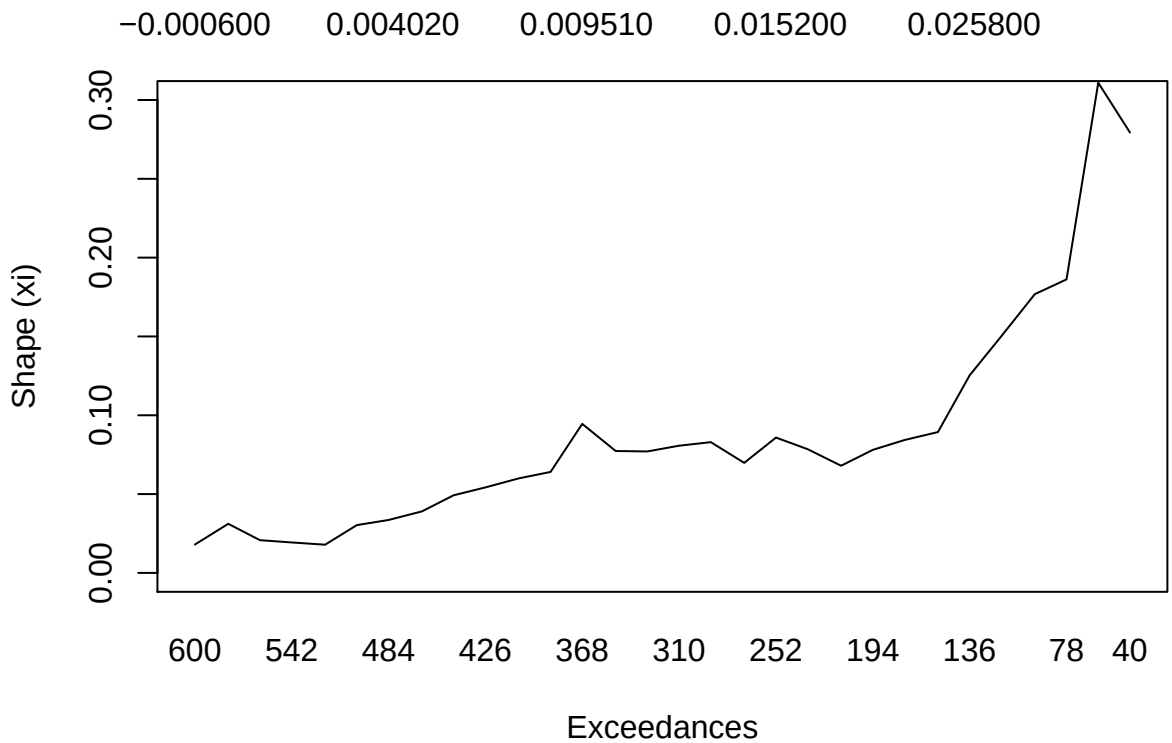
```
##
## Attaching package: 'evir'
```

```
## The following objects are masked from 'package:POT':
##
##     dgpd, pgpd, qgpd, rgpd
```

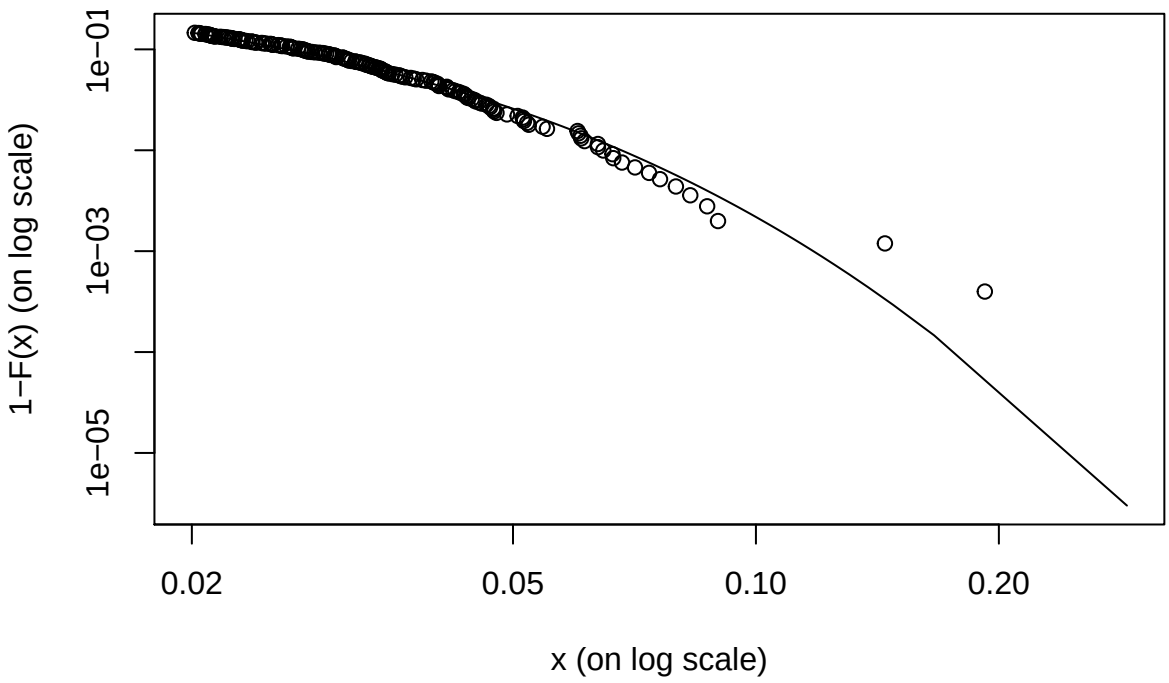
```
eecdf = ecdf(WSPLRet)
uv = seq(from = -.06,to = -.01, by = .0025)
plot(uv,eecdf(uv),type='l',xlab='x',ylab='Empirical CDF - WSPLRet')
```




```
shape(-WSPLRet, models = 30, start = 40, end = 600, reverse = TRUE, ci=FALSE,auto.scale='FALSE', ylim=c
```




```
gpd_est = gpd(-WSPLRet, thresh=.02, method = c("ml"), information = c("observed"))
xi = gpd_est$par.ests[1]
scale = gpd_est$par.ests[2]
tp = tailplot(gpd_est)
```



(b) Compute the relative VaR (expressed in units of the current price) at the level = .002 utilizing for the case of fitting a normal distribution to the returns (not using any POT).

Witht the level = .002, we fitting the normal distribution by return dataset's mean and mad, because this normal distrbution Then, the rel-VaR = 0.07419033 * Unit price

```
set.seed(123)
VaRt_nor <- - qnorm(0.002,mean(ret),sd(ret)) # atready based on relative return
VaRt_nor
```

```
## [1] 0.07419033
```

(c) Compute VaR for 1 million dollars invested in Ford stock at the level = .002 utilizing the GPD distributional fits generated in part (a). Discuss the comparison of this result with result from part (b).

After transformation to relative return, for the negative log-returns, the rel-VaR is 0.1073185, which is bigger than the normal estimated one VaRt_nor = 0.07419033 So VaR for 1 million dollars utilizing the GPD = 10731.85, while fits generated in part (a) = 74190.33, which is quite smaller than GPD's one. Because the GDP has a heavier tail, so the quantile is large than normal in the tail parts.

```
m = 0.02
level = 0.002
alphat = 1-level/eecdf(-m)
q_1= qgpd( alphat , xi = gpd_est$par.ests[1], mu = m, beta = gpd_est$par.ests[2])
VaRt = exp(q_1)-1
VaRt
```

```
##      beta
## 0.1073185
```

```
100000 * VaRt # VaR by GPD
```

```
##     beta
## 10731.85
```

```r
1000000*VaRt_nor # VaR by normal
```

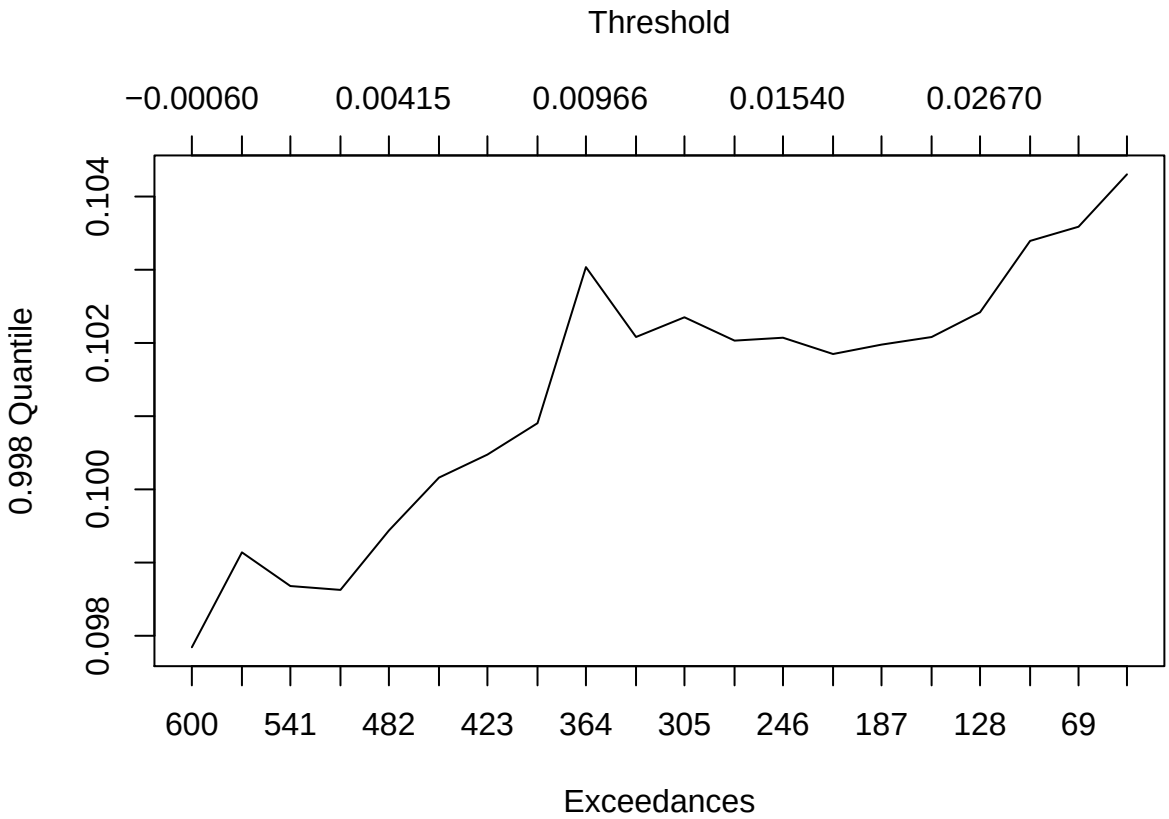
```
## [1] 74190.33
```

```r
# another method:
# a=quant(-WSPLRet, p = 0.998, models = 20, start = 600, end = 40, reverse =TRUE, ci = FALSE, auto.scal
# VaRt = -exp(-a[2,7])-1
# # a[2,7] is qest with threshold 0.01731645, which is in the most stable range
# #The plot shows the estimated 0.998 quantile for the log return as a function of the thresholds is ve
```

(d) Utilizing 'quant" command in evir package, give a discussion on the stability of the VaR as function of threshold.

The plot shows the estimated 0.998 quantile for the log return as a function of the thresholds is very stable over the range 0.0096 to 0.0267, and quantile range is from 0.098 to 0.104, which is a quite small interval. So we can regard it as a stable result. Based on this result, we have confidence that the Generalized Pareto distribution is a reasonably solid and stable way to model the tail-distribution and estimate relative VaR at level = 0.002.

```r
a=quant(-WSPLRet, p = 0.998, models = 20, start = 600, end = 40, reverse =TRUE, ci = FALSE, auto.scale =
```



(e) Compute expected shortfall for part (c) using Monte Carlo methods and the estimated GPD distribution and relative VaR in (c).

By using monte carlo, we generate random numerbs of estimated gpd. Then for all loss which is larger than VaRt, we calculate the mean of them. Then, the expected shortfall = 0.1287589, which is larger than VaRt = 0.07419033 So it is reasonable. Then, expected shortfall for 1 million portfolio is 128899.9

```r
m = 0.02 # threshold, not original location parameter.
r_gpd <- -rgpd(1000000, xi = gpd_est$par.ests[1], mu = 0.02,beta = gpd_est$par.ests[2])
# need to useing -rgpd at this mometn, because the gpd above is estimated by loss fun
```

```r
returns <- exp(r_gpd) - 1 # change log return to original return
ES <- -mean(returns[returns < -VaRt])
1000000 * ES
```

```
## [1] 129226.9
```

```r
ES
```

```
## [1] 0.1292269
```