Central European University
Empirical Finance
Final Project

Convergence of Log Returns' Maxima and the Role of Liquidity

by

Chyngyz Davletov

Professor: Rosario Nunzio Mantegna
Teaching Assistant: Luca Marotta

April 2016

Table of Contents

List of Tables

List of Figures

1 Introduction

The purpose of this work is to examine whether the distribution of the normalized maxima of log returns from time series of different stocks' prices converges to the Gumbel distribution when properly normalized and to investigate the role of liquidity in the convergence through analysing stocks with different levels of market capitalization, which is the proxy for liquidity. For these purposes I take three hundred stocks from London Stock Exchange, dividing them into three samples with one hundred stocks in each. After analysing the distribution of the maxima of log returns from each sample, I conclude that convergence to the Gumbel distribution takes place and is the fastest for highly-capitalized stocks. The technical part was done in R studio.

2 Data

As mentioned in the introductory part, three hundred stocks from London Stock Exchange (LSE) were used in this work and divided into three samples by the order of market capitalization. In particular, the sample of highly-capitalized stocks consists of the constituents of FTSE 100 index (Financial Times Stock Exchange 100), which contains one hundred stocks with the highest market capitalization traded on LSE; the sample of middle-capitalized stocks is comprised of one hundred stocks from FTSE 250 index, which contains two hundred and fifty stocks rated from 101 to 350 in terms of market capitalization on the LSE; finally, the sample of low-capitalized stocks consists of one hundred stocks from FTSE Small Cap that contains the stocks with the lowest capitalization traded on LSE. The order of capitalization for highly-capitalized, middle-capitalized, and low-capitalized is of 10 billion, 1 billion, and 100 million correspondingly. The time series for all stocks were obtained from Yahoo Finance.

3 Strategy

In order to implement the analysis I used the following strategy:

1. Get both daily and monthly historical data for each stock
2. Compute log returns for each series
3. Normalize monthly log returns by standard deviation for each 20-day window obtained from daily data
4. Get maxima of the normalized series for each stock
5. Normalize the obtained distribution of the maxima by the appropriate normalizing constants
6. Get the probability density function for each of FTSE 100, FTSE 250, FTSE Small Cap samples

The appropriate normalizing constants are:

$$a_n = (2 \ln n)^{-1/2}$$

and

$$b_n = (2 \ln n)^{1/2} - \frac{\ln(\ln n) + \ln 4\pi}{2(2 \ln n)^{1/2}},$$

for which

$$\frac{Z_n - b_n}{a_n} \xrightarrow{D} \text{Gumbel Distribution},$$

where Z_n is the maximum. Figure 1 shows the appearance of the data.

Date	logreturns
2016-03-24	-0.0443302105
2016-03-23	-0.0553633196
2016-03-22	0.0068915761
2016-03-21	-0.0106798010
2016-03-18	0.0264541560
2016-03-17	0.0933339493
2016-03-16	0.0107127485
2016-03-15	-0.1143465865
2016-03-14	0.0582030366
2016-03-11	0.0251387943
2016-03-10	-0.0532504074
2016-03-09	-0.0011307954
2016-03-08	-0.1681257099
2016-03-07	0.0591927546
2016-03-04	0.1049852107

(a) Daily

Date	logreturns	sd	normalized returns
2016-03-01	0.0419000181	0.020289872	0.461763817
2016-02-01	0.5486660649	0.020910351	5.867212111
2016-01-04	-0.0763067219	0.013823068	-1.234364266
2015-12-01	-0.3109115853	0.017690873	-3.929819907
2015-11-02	-0.2908582295	0.026362449	-2.467065103
2015-10-01	-0.0078360310	0.018464868	-0.094893165
2015-09-01	-0.2964473208	0.021676158	-3.058089756
2015-08-03	-0.0639348471	0.021250443	-0.672751452
2015-07-01	-0.1242276926	0.021211180	-1.309599798
2015-06-01	-0.1106806988	0.031907386	-0.775649774
2015-05-01	-0.0741776761	0.035674759	-0.464940282
2015-04-01	0.0879160469	0.037598930	0.522850672
2015-03-02	-0.1278650720	0.021366967	-1.338116893
2015-02-02	0.0835603719	0.013882240	1.345940371

(b) Monthly

Figure 1: Data for 1 stock: (a) initial daily time series; (b) normalized monthly time series

4 Technical Issues

When normalizing the monthly log returns by the standard deviation for each 20-day window, I obtained sometimes infinite values. This was due to the fact that standard deviations for some 20-day windows for several stocks were essentially zero. And mostly those stocks were low-capitalized. This is because low-capitalized stocks are traded a lot less frequently than highly-capitalized because of lower liquidity of the former. The infinite values affected the shape of the density functions of the maxima, thus I decreased the sample size by removing stocks for which the standard deviation was zero. Another issue is associated with the quality of records for low-capitalized stocks. Perhaps because these stocks are not as important as the highly-capitalized stocks, historical prices are recorded in a less accurate manner so that sometimes there were observed mismatches of dimensions (like daily data has more months than monthly data) and the function in R reported an error. As mentioned, mostly this issue is associated with low-capitalized stocks resulting in the final samples that are summarized in Table 1. After removing stocks with "bad" records I ended up with 90, 82, and 62 stocks for each sample. After removing infinite values from theses samples, I ended up with 87, 72, and 43 stocks in each sample. The sample decreases significantly for low-capitalized stocks as expected.

5 Results

This part presents the results that were obtained using the data and following the steps described in the strategy part. Figure 2 presents the densities of the obtained maxima for the three samples

Table 1: Final Samples

Sample	Without "bad" records	Without infinite values
FTSE 100	90	87
FTSE 250	82	72
FTSE Small Cap	62	43

without “bad” records, i.e. samples with 90, 82, and 62 stocks, centred and normalized with proper constants. Figure 3 presents the densities for three samples without infinite values (i.e. 87, 72, and 43). It is apparent that the density shrinks and gets more mass around the mean as the order of market capitalization increases. It is interesting that with highly-capitalized stocks, the density has even more mass around the mean than the Gumbel distribution. What is also worth noting between figures 2 and 3, is that when the infinite values are removed, density increases at the mean, which is particularly pronounced with FTSE Small Cap and FTSE 250 (because the sample size for FTSE 100 decreases insignificantly, its shape is almost unchanged from figure 2 to 3).

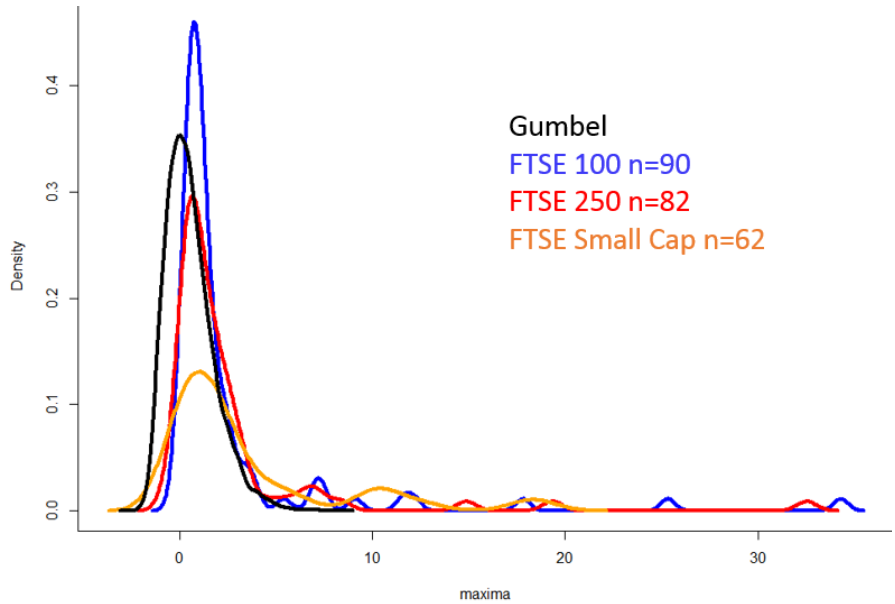

Figure 2: Distribution of Maxima without “Bad” Records

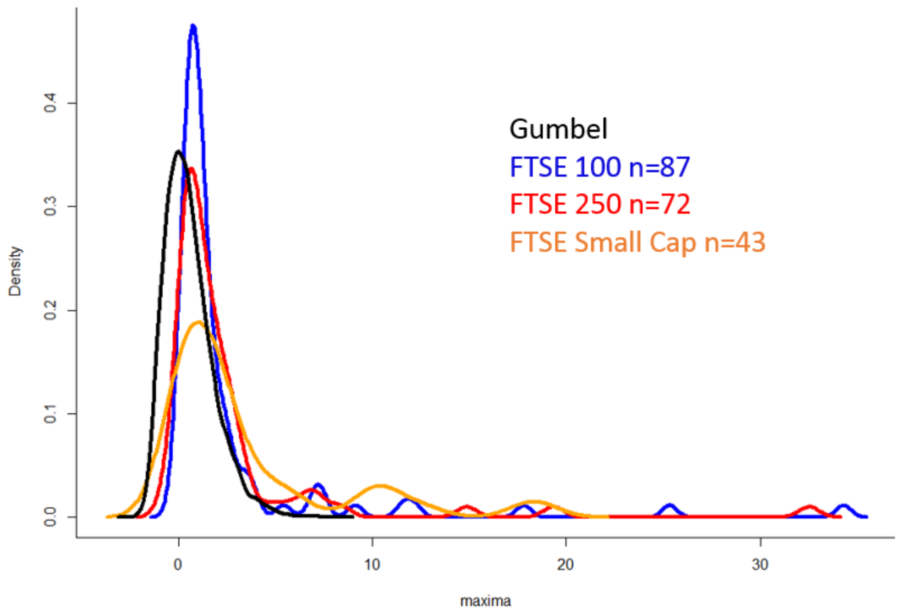

Figure 3: Distribution of Maxima without Infinite Values

So far I have presented the results for the historical prices. Since there is possibly a heterogeneity due to different lengths of time periods of historical data for different stocks, I censor the data from below by: (i) keeping stocks with historical data of more than 20 years; (ii) keeping stocks with historical data of more than 10 years. It should be noted that for the most stocks that are kept time periods overlap. Figure 4 and Figure 5 present the results for 20 and 10 years respectively. Since the samples for middle- and low-capitalized stocks have been already decreased significantly, I present results only for highly-capitalized stocks. Figures report a better convergence in terms of the mass around the mean (compare with the red curve, which is the same as the FTSE 100 in figure 3). In addition there is no significant difference between censoring by 20 years and censoring by 10 years.

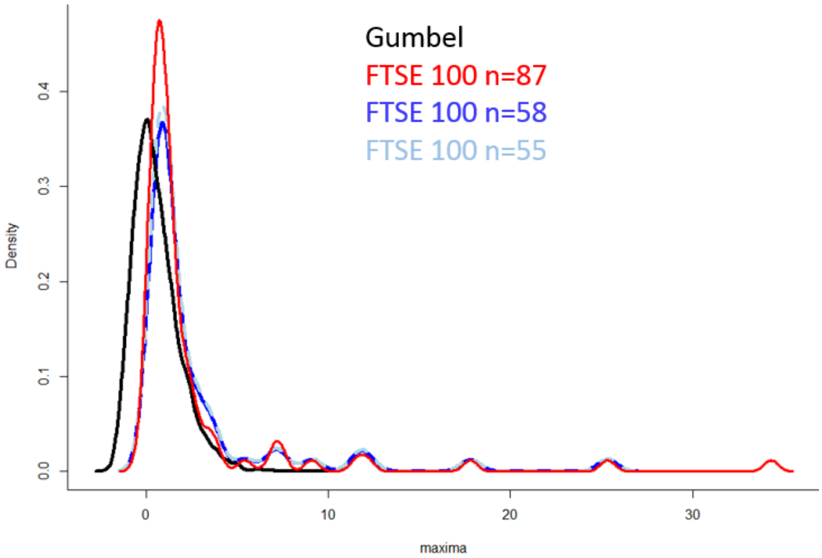

Figure 4: Distribution of Maxima for Stocks with more than 20 Years of Historical Data

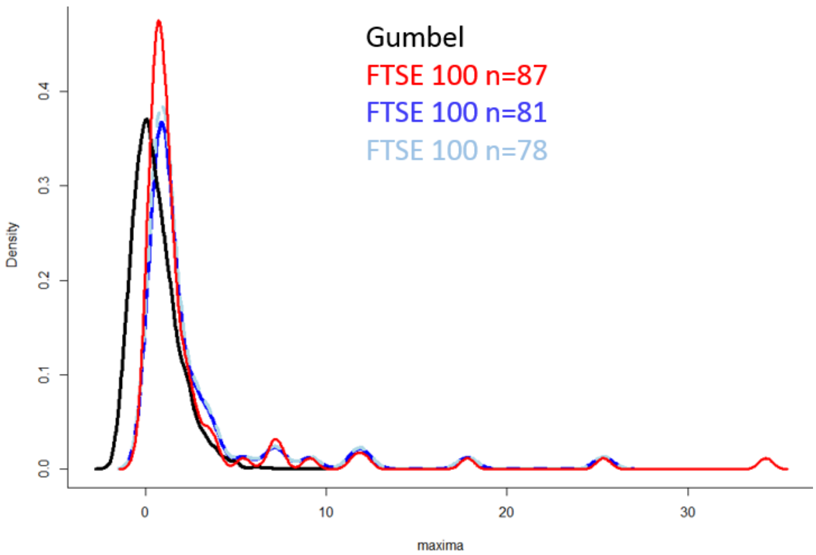

Figure 5: Distribution of Maxima for Stocks with more than 10 Years of Historical Data

6 Conclusion

The aim of this project was to examine whether the maxima of normalized log returns converge to the Gumbel distribution on the set of three hundred stocks from London Stock Exchange and see the role of liquidity in the convergence. During the work I faced several issues which resulted in the reduction of the sample size. Nevertheless, the distribution does seem to converge to the Gumbel, and this convergence is faster for highly-capitalized stocks suggesting the important role of liquidity. The convergence also takes place when the historical periods for the stocks are censored from below.

This work has an important drawback of small sample. In order to deal with this pitfall it is possible to include highly-capitalized stocks from other well-known exchange markets such as NYSE, NASDAQ, Japan Exchange Group, Shanghai Stock Exchange, and others. Also in the context of this topic, it is sufficient to concentrate on the highly-liquid stocks, because they exhibit the fastest convergence. Increasing sample size in this way is expected to deliver more statistically strong results and is the objective that should be pursued in the further exercises on this topic.

Appendix

```
setwd("C:/Users/User/Desktop/Чина/CEU/2nd Year/2.Winter2016/Empirical Finance/Project/Data/FTSE100")

#temp = list.files(pattern="*.csv")
#myfiles = lapply(temp, read.delim)

#temp = list.files(pattern="*.csv")
#for (i in 1:length(temp)) assign(temp[i], read.csv(temp[i]))

temp <- list.files(pattern="*.csv")
list2env(lapply(setNames(temp, make.names(gsub("*.csv$", "", temp))), read.csv), envir = .GlobalEnv)

#### normalized monthly returns function ####

# x is daily data
# y is monthly data

nr <- function(x,y){
  x["logreturns"] <- numeric()
  x1 <- rev(diff(rev(log(x[,7]))))
  x <- x[-nrow(x),]
  x[,8] <- x1
  x["Date1"] <- character()
  x[,9] <- substr(x[,1],1,7)
  x

  y["logreturns"] <- numeric()
  y1 <- rev(diff(rev(log(y[,7]))))
  y <- y[-nrow(y),]
  y[,8] <- y1
  y[,"sd"] <- numeric()
  y2 <- tapply(x[,8],x[,9],sd)
  y[,"sd"] <- y2[-length(y2)]
  y[,"normalized returns"] <- numeric()
  co <- 1/sqrt(2*log(nrow(y)))
  de <- sqrt(2*log(nrow(y)))-((log(4*pi)+log(log(nrow(y))))/(2*sqrt(2*log(nrow(y)))))
  y[,10] <- ((y[,8]/(y[,9]*sqrt(20)))-de)*co
  as.matrix(y[,10])
}
###

a = c(
  max(nr(AAL.L.d,AAL.L.m)),max(nr(ADM.L.d,ADM.L.m)),max(nr(AHT.L.d,AHT.L.m)),max(nr(ANTO.L.d,ANTO.L.m)),max(nr(ARM.L.d,ARM.L.m)),
  max(nr(AV.L.d,AV.L.m)),max(nr(AZN.L.d,AZN.L.m)),max(nr(BA.L.d,BA.L.m)),max(nr(BAB.L.d,BAB.L.m)),max(nr(BARC.L.d,BARC.L.m)),
  max(nr(BATS.L.d,BATS.L.m)),max(nr(BDEV.L.d,BDEV.L.m)),max(nr(BKG.L.d,BKG.L.m)),max(nr(BLND.L.d,BLND.L.m)),max(nr(BLT.L.d,BLT.L.m)),
  max(nr(BNZL.L.d,BNZL.L.m)),max(nr(BP.L.d,BP.L.m)),max(nr(BRBY.L.d,BRBY.L.m)),max(nr(BT.A.L.d,BT.A.L.m)),max(nr(CCH.L.d,CCH.L.m)),
  max(nr(CCL.d,CCL.m)),max(nr(CNA.L.d,CNA.L.m)),max(nr(CPG.L.d,CPG.L.m)),max(nr(CPI.L.d,CPI.L.m)),max(nr(CRH.L.d,CRH.L.m)),
  max(nr(DC.L.d,DC.L.m)),max(nr(DCC.L.d,DCC.L.m)),max(nr(DGE.L.d,DGE.L.m)),max(nr(DLG.L.d,DLG.L.m)),max(nr(EXPN.L.d,EXPN.L.m)),
  max(nr(EZJ.L.d,EZJ.L.m)),max(nr(FRES.L.d,FRES.L.m)),max(nr(GKN.L.d,GKN.L.m)),max(nr(GLEN.L.d,GLEN.L.m)),max(nr(GSK.L.d,GSK.L.m)),
  max(nr(HL.L.d,HL.L.m)),max(nr(HMSO.L.d,HMSO.L.m)),max(nr(HSBA.L.d,HSBA.L.m)),max(nr(IAG.L.d,IAG.L.m)),#max(nr(IHG.L.d,IHG.L.m)),
  max(nr(III.L.d,III.L.m)),max(nr(IMB.L.d,IMB.L.m)),max(nr(INF.L.d,INF.L.m)),#max(nr(INTU.L.d,INTU.L.m)),
  max(nr(ISAT.L.d,ISAT.L.m)),
  max(nr(ITRK.L.d,ITRK.L.m)),max(nr(ITV.L.d,ITV.L.m)),max(nr(JMAT.L.d,JMAT.L.m)),max(nr(KGF.L.d,KGF.L.m)),max(nr(LAND.L.d,LAND.L.m)),
  max(nr(LGEN.L.d,LGEN.L.m)),max(nr(LLOY.L.d,LLOY.L.m)),max(nr(LSE.L.d,LSE.L.m)),max(nr(MERL.L.d,MERL.L.m)),max(nr(MKS.L.d,MKS.L.m)),
  #max(nr(MNDI.L.d,MNDI.L.m)),
  max(nr(MRW.L.d,MRW.L.m)),max(nr(NG.L.d,NG.L.m)),max(nr(OML.L.d,OML.L.m)),max(nr(PFG.L.d,PFG.L.m)),
  max(nr(PPB.L.d,PPB.L.m)),max(nr(PRU.L.d,PRU.L.m)),max(nr(PSN.L.d,PSN.L.m)),max(nr(PSON.L.d,PSON.L.m)),max(nr(RB.L.d,RB.L.m)),
  max(nr(RBS.L.d,RBS.L.m)),max(nr(RDSA.L.d,RDSA.L.m)),max(nr(RDSB.L.d,RDSB.L.m)),max(nr(REL.L.d,REL.L.m)),max(nr(REX.L.d,REX.L.m)),
  max(nr(RIO.L.d,RIO.L.m)),max(nr(RMG.L.d,RMG.L.m)),max(nr(RR.L.d,RR.L.m)),max(nr(RRS.L.d,RRS.L.m)),max(nr(RSA.L.d,RSA.L.m)),
  max(nr(SAB.L.d,SAB.L.m)),max(nr(SBRY.L.d,SBRY.L.m)),max(nr(SDR.L.d,SDR.L.m)),max(nr(SGE.L.d,SGE.L.m)),max(nr(SHP.L.d,SHP.L.m)),
  max(nr(SKY.L.d,SKY.L.m)),max(nr(SL.L.d,SL.L.m)),max(nr(SN.L.d,SN.L.m)),max(nr(SSE.L.d,SSE.L.m)),max(nr(STAN.L.d,STAN.L.m)),
  max(nr(STJ.L.d,STJ.L.m)),max(nr(SVT.L.d,SVT.L.m)),max(nr(TPK.L.d,TPK.L.m)),max(nr(TSCO.L.d,TSCO.L.m)),max(nr(ULVR.L.d,ULVR.L.m)),

max(nr(UU.L.d,UU.L.m)),max(nr(VOD.L.d,VOD.L.m)),max(nr(WOS.L.d,WOS.L.m)),max(nr(WPG.L.d,WPG.L.m)),max(nr(WPP.L.d,WPP.L.m)),max(nr(WTB.L.d,WTB.L.m))
)

b <- as.matrix(a)
b <- b[!is.na(b)]
b <- as.matrix(b)
#plot(density(b))

c <- b[!is.infinite(b)]
c <- as.matrix(c)
#plot(density(c))

setwd("C:/Users/User/Desktop/Чина/CEU/2nd Year/2.Winter2016/Empirical Finance/Project/Data/FTSE250")
temp <- list.files(pattern="*.csv")
list2env(lapply(setNames(temp, make.names(gsub("*.csv$", "", temp))), read.csv), envir = .GlobalEnv)

a2 = c(
  max(nr(CLDNd,CLDNm)),max(nr(CLId,CLIm)),#max(nr(CLLNd,CLLNm)),
  max(nr(CNEd,CNEm)),max(nr(COBd,COBm)),
  max(nr(CRDAd,CRDAm)),max(nr(CRSTd,CRSTm)),max(nr(CWCd,CWCm)),max(nr(CWDd,CWDm)),max(nr(CWKd,CWKm)),
  max(nr(DCGd,DCGm)),max(nr(DEBd,DEBm)),max(nr(DJANd,DJANm)),max(nr(DLNd,DLNm)),max(nr(DNLMd,DNLMm)),
  max(nr(DOMd,DOMm)),max(nr(DPHd,DPHm)),max(nr(DPLMd,DPLMm)),max(nr(DRXd,DRXm)),max(nr(DTYd,DTYm)),
  max(nr(ECMd,ECMm)),#max(nr(EDINd,EDINm)),
  max(nr(ELMd,ELMm)),#max(nr(ELTAd,ELTAm)),
  max(nr(EMGd,EMGm)),
  max(nr(ERMd,ERMm)),max(nr(ESNTd,ESNTm)),max(nr(ESURd,ESURm)),max(nr(ETOd,ETOm)),max(nr(EVRd,EVRm)),
  max(nr(FCPTd,FCPTm)),max(nr(FCSSd,FCSSm)),max(nr(FDSAd,FDSAm)),#max(nr(FEVd,FEVm)),
  max(nr(FGPd,FGPm)),
  #max(nr(FGTd,FGTm)),
  #max(nr(FRCLd,FRCLm)),
  max(nr(GCPd,GCPm)),max(nr(GFRDd,GFRDm)),max(nr(GFSd,GFSm)),
  max(nr(GFTUd,GFTUm)),max(nr(GNCd,GNCm)),max(nr(GNKd,GNKm)),max(nr(GNSd,GNSm)),max(nr(GOGd,GOGm)),
  max(nr(GPORd,GPORm)),max(nr(GRGd,GRGm)),max(nr(GRId,GRIm)),max(nr(GSSd,GSSm)),max(nr(HASd,HASm)),
  max(nr(HFDd,HFDm)),max(nr(HGGd,HGGm)),max(nr(HICLd,HICLm)),max(nr(HIKd,HIKm)),max(nr(HLMAd,HLMAm)),
  max(nr(HMSFd,HMSFm)),max(nr(HOMEd,HOMEm)),max(nr(HSTNd,HSTNm)),max(nr(HSVd,HSVm)),max(nr(HSXd,HSXm)),
  #max(nr(HVPEd,HVPEm)),
  max(nr(HWDNd,HWDNm)),max(nr(IAPd,IAPm)),max(nr(ICPd,ICPm)),max(nr(IGGd,IGGm)),
  max(nr(IMId,IMIm)),max(nr(INCHd,INCHm)),max(nr(INPPd,INPPm)),max(nr(INVPd,INVPm)),max(nr(IPFd,IPFm)),
  max(nr(IPOd,IPOm)),max(nr(IRVd,IRVm)),#max(nr(JAMd,JAMm)),
  max(nr(JDd,JDm)),max(nr(JDWd,JDWm)),
  max(nr(JLIFd,JLIFm)),max(nr(JLTd,JLTm)),#max(nr(JMGd,JMGm)),
  max(nr(JRGd,JRGm)),max(nr(JUPd,JUPm)),
  max(nr(KAZd,KAZm)),max(nr(KIEd,KIEm)),max(nr(KLRd,KLRm)),max(nr(LADd,LADm)),max(nr(LMPd,LMPm)),
  max(nr(LOOKd,LOOKm)),max(nr(LRDd,LRDm)),max(nr(LREd,LREm)),#max(nr(MABd,MABm)),
```

```
  max(nr(MARSd,MARSm)),
  max(nr(MCROd,MCROm)),max(nr(MGAMd,MGAMm)),max(nr(MGGTd,MGGTm)),max(nr(MLCd,MLCm)),#max(nr(MNKSd,MNKSm)),
  max(nr(MONYd,MONYm)),max(nr(MPId,MPIm)),#max(nr(MRCd,MRCm)),
  max(nr(MROd,MROm)),max(nr(MSLHd,MSLHm))
)

b2 <- as.matrix(a2)
b2 <- b2[!is.na(b2)]
b2 <- as.matrix(b2)
#plot(density(b2))

c2 <- b2[!is.infinite(b2)]
c2 <- as.matrix(c2)
#plot(density(c2))

setwd("C:/Users/User/Desktop/Чина/CEU/2nd Year/2.Winter2016/Empirical Finance/Project/Data/FTSESmallCap")
temp <- list.files(pattern="*.csv")
list2env(lapply(setNames(temp, make.names(gsub("*.csv$", "", temp))), read.csv), envir = .GlobalEnv)

# STOCK EEE is 888 - i changed it from 888 to EEE.
a3 = c(
  #max(nr(AASd,AASm)),
  #max(nr(ABDd,ABDm)),
  max(nr(AEPd,AEPm)),#max(nr(AGITd,AGITm)),
  max(nr(AGRd,AGRm)),max(nr(ALYd,ALYm)),#max(nr(ANWd,ANWm)),
  max(nr(APFd,APFm)),max(nr(AQPd,AQPm)),
  max(nr(ARWd,ARWm)),max(nr(ATSd,ATSm)),#max(nr(AUKTd,AUKTm)),
  max(nr(AVONd,AVONm)),max(nr(B32d,B32m)),
  max(nr(BACTd,BACTm)),max(nr(BBGId,BBGIm)),#max(nr(BEEd,BEEm)),max(nr(BGFDd,BGFDm)),max(nr(BGSd,BGSm)),
  max(nr(BHYd,BHYm)),#max(nr(BIOGd,BIOGm)),
  max(nr(BKTd,BKTm)),max(nr(BMSd,BMSm)),max(nr(BMYd,BMYm)),
  max(nr(BPId,BPIm)),max(nr(BRAMd,BRAMm)),max(nr(BRFId,BRFIm)),max(nr(BRGEd,BRGEm)),#max(nr(BRSCd,BRSCm)),
  max(nr(BRWMd,BRWMm)),max(nr(BSIFd,BSIFm)),#max(nr(BUTd,BUTm)),
  max(nr(BVCd,BVCm)),max(nr(CALd,CALm)),
  max(nr(CARd,CARm)),#max(nr(CDId,CDIm)),max(nr(CGTd,CGTm)),
  max(nr(CHGd,CHGm)),max(nr(CIUd,CIUm)),
  max(nr(CKNd,CKNm)),max(nr(CLIGd,CLIGm)),max(nr(CMSd,CMSm)),max(nr(CNCTd,CNCTm)),max(nr(COSTd,COSTm)),
  max(nr(CPRd,CPRm)),max(nr(CSNd,CSNm)),max(nr(CSRTd,CSRTm)),max(nr(CTRd,CTRm)),max(nr(CVCd,CVCm)),
  max(nr(CYNd,CYNm)),max(nr(DABd,DABm)),#max(nr(DIAd,DIAm)),
  max(nr(DRTYd,DRTYm)),max(nr(DVOd,DVOm)),
  max(nr(ECWOd,ECWOm)),max(nr(EEEd,EEEm)),
  #max(nr(EFMd,EFMm)),
  max(nr(ENQd,ENQm)),#max(nr(EWId,EWIm)),
  max(nr(EXId,EXIm)),
  #max(nr(FASd,FASm)),
  max(nr(FAXd,FAXm)),max(nr(FDLd,FDLm)),max(nr(FENRd,FENRm)),#max(nr(FJVd,FJVm)),
  max(nr(FLYBd,FLYBm)),max(nr(FOURd,FOURm)),max(nr(FSTAd,FSTAm)),#max(nr(FSVd,FSVm)),
  max(nr(FXPOd,FXPOm)),
  #max(nr(GAWd,GAWm)),
  max(nr(GDWNd,GDWNm)),max(nr(GEMDd,GEMDm)),max(nr(GLEd,GLEm)),#max(nr(GPEd,GPEm)),
  max(nr(HDYd,HDYm)),#max(nr(HGTd,HGTm)),
  max(nr(HLCLd,HLCLm)),max(nr(HNTd,HNTm)),max(nr(HOCd,HOCm)),
  max(nr(HRGd,HRGm)),#max(nr(HRId,HRIm)),
  max(nr(HSDd,HSDm)),max(nr(IEMd,IEMm)),#max(nr(IMGd,IMGm)),
  max(nr(ITEd,ITEm)),#max(nr(JAId,JAIm)),max(nr(JESCd,JESCm)),max(nr(JETGd,JETGm)),
  max(nr(JFJd,JFJm)),
  #max(nr(JIId,JIIm)),
  max(nr(JPRd,JPRm)),#max(nr(JRSd,JRSm)),
  max(nr(KCOMd,KCOMm)),max(nr(KMRd,KMRm)),
  max(nr(LAMd,LAMm)),max(nr(LSLd,LSLm)),max(nr(MCBd,MCBm)),max(nr(MMCd,MMCm))#,max(nr(THRGd,THRGm))
)

b3 <- as.matrix(a3)
b3 <- b3[!is.na(b3)]
b3 <- as.matrix(b3)
#plot(density(b3))

c3 <- b3[!is.infinite(b3)]
c3 <- as.matrix(c3)
#plot(density(c3))

library(evd)
d <- dgumbel(1:100,loc=0,scale=1)
gum <- rgumbel(10000,loc=0,scale=1)
g <- density(gum)



#co1 <- (2*log(90))^(-0.5)
#de1 <- sqrt(2*log(90))-((log(4*pi)+log(log(90)))/(2*((2*log(90))^(0.5))))

#co2 <- (2*log(87))^(-0.5)
#de2 <- sqrt(2*log(87))-((log(4*pi)+log(log(87)))/(2*((2*log(87))^(0.5))))

#co3 <- (2*log(82))^(-0.5)
#de3 <- sqrt(2*log(82))-((log(4*pi)+log(log(82)))/(2*((2*log(82))^(0.5))))

#co4 <- (2*log(72))^(-0.5)
#de4 <- sqrt(2*log(72))-((log(4*pi)+log(log(72)))/(2*((2*log(72))^(0.5))))

#co5 <- (2*log(62))^(-0.5)
#de5 <- sqrt(2*log(62))-((log(4*pi)+log(log(62)))/(2*((2*log(62))^(0.5))))

#co6 <- (2*log(43))^(-0.5)
#de6 <- sqrt(2*log(43))-((log(4*pi)+log(log(43)))/(2*((2*log(43))^(0.5))))

d1 <- density(b)
#d1 <- density((b-de1)*co1)
d2 <- density(c)
#d2 <- density((c-de2)/(co2^(-1)))
d3 <- density(b2)
#d3 <- density((b2-de3)/(co3^(-1)))
d4 <- density(c2)
#d4 <- density((c2-de4)/(co4^(-1)))
d5 <- density(b3)
#d5 <- density((b3-de5)/(co5^(-1)))
```

```
d6 <- density(c3)
#d6 <- density((c3-de6)/(co6^(-1)))


plot(range(g$x,d1$x, d3$x, d5$x), range(g$y, d1$y, d3$y, d5$y), type = "n", xlab = "maxima",
     ylab = "Density", main = "Density of maxima of normalized log returns")
lines(d1, col = "blue",lwd=4)
lines(d3, col = "red",lwd=4)
lines(d5, col = "orange",lwd=4)
lines(g, col = "black",lwd=4)

plot(range(g$x,d2$x, d4$x, d6$x), range(g$y, d2$y, d4$y, d6$y), type = "n", xlab = "maxima",
     ylab = "Density", main = "Density of maxima of normalized log returns")
lines(d2, col = "blue",lwd=4)
lines(d4, col = "red",lwd=4)
lines(d6, col = "orange",lwd=4)
lines(g, col = "black",lwd=4)
```