

**Vizsga GP modellek 2024. május 15. 10 óra.**

**Zavarkó Máté**

**IN3BLK**

## 1. Feladat:

R kód futtatásával kapott eredmény:

```
> x="in3blk";#neptun kód
> z=charToRaw(iconv(x, "latin1", "UTF-8"))
> for (i in 1:6) v=paste("0x",z,sep="")
> e=strtoi(v)
> ax=e[1];ay=e[2];az=e[3];av=e[4];ss=sum(strtoi(v))+24
> cat("ax=",ax,"\n")
ax= 105
> cat("ay=",ay,"\n")
ay= 110
> cat("az=",az,"\n")
az= 51
> cat("av=",av,"\n")
av= 98
> cat("ss=",ss,"\n")
ss= 603
> ar=c( "FB", "AAPL", "AMZN", "GOOG", "NFLX", "TSLA")
> ai=ss-6*floor(ss/6)
> ev=2022-(ss-10*floor(ss/10))
> cat("ev=",ev,"\n")
ev= 2019
> cat("reszveny=",ar[ai+1],"\n")
reszveny= GOOG
```

Tehát a 2019 évi GOOG részvény adatokra van szükségem.

Az adatok az GOOG.csv fájlban találhatóak.

Napi változások eloszlása:

```
One Sample t-test

data:  daily_changes
t = 1.0352, df = 249, p-value = 0.3016
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.05240013  0.16851613
sample estimates:
mean of x
 0.058058
```

A görbe 0.3016 p-valueval mutat student eloszlást, de a qq plot alapján látszik hogy ez inkább normális eloszlású lehet.

## 2. Feladat:

(A)

A megadott kódban generált mintarealizációról kellett statisztikai elemzést készíteni.

summary(zn) parancs:

```
> summary(zn)
      V1      V2
Min.   :-13358.807 Min.   :-82941.57
1st Qu.: -115.670 1st Qu.: -293.57
Median :  -2.596  Median :  -28.48
Mean   :   19.265 Mean   : -219.56
3rd Qu.:  105.964 3rd Qu.:  186.51
Max.   :   9277.763 Max.   : 16765.88
```

A summary parancs statisztikai elemzést készít a mintáról.

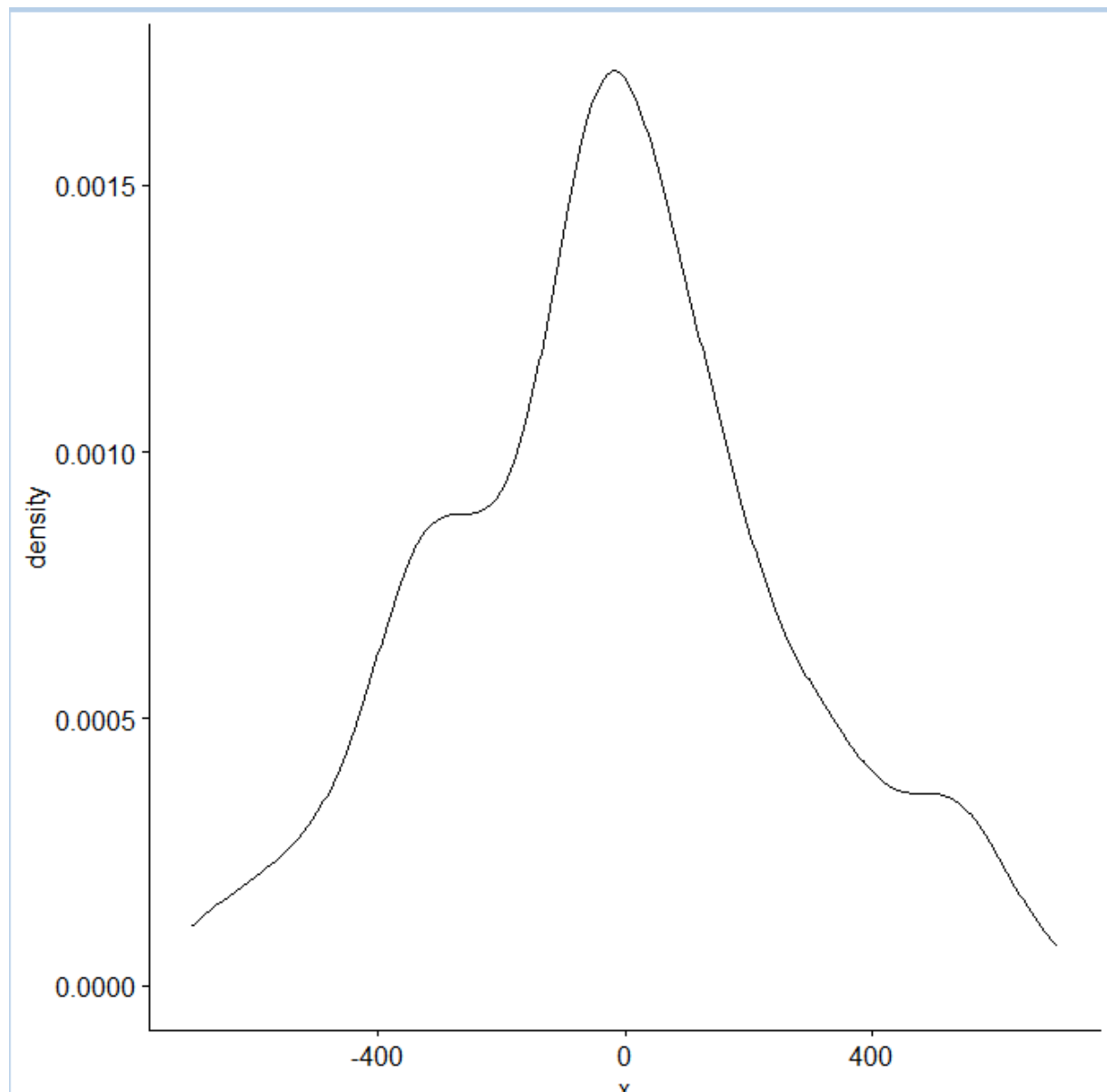
Megtalálható a minimum, első quartilis, medián, átlag, harmadik quartilis, maximum.

Eloszlás megállapítása:

```
library(ggpubr)
ggdensity(zn[,1]) + xlim(-700, 700)
ggdensity(zn[,2]) + xlim(-700, 700)
```

Peremvizsgálat:

```
#peremvizsgálat az eloszlás megerősítéséhez:
cov_zn= cov(zn)
mean_zn = colMeans(zn)
```



(B)

**A kétdimenziós eloszlás:** két változó közötti kapcsolatot (például kovarianciát) ír le. Ez a kapcsolat lehetővé teszi az eloszlás leírását egy bizonyos módon. Az ilyen eloszlások segítenek megérteni, milyen összefüggés lehet a változók között. Egyes esetekben a normál eloszlás jó közelítés lehet, de ez nem mindig igaz. Az eloszlás pontos jellege további elemzést igényelhet, például a korrelációs együttható meghatározásával vagy a marginális eloszlások vizsgálatával.

### 3. Feladat:

Feladat paramétereinek meghatározása

```
n <- 800  
rho <- 0.6  
sigma1 <- 2  
sigma2 <- 3
```

Szórásmátrix létrehozása

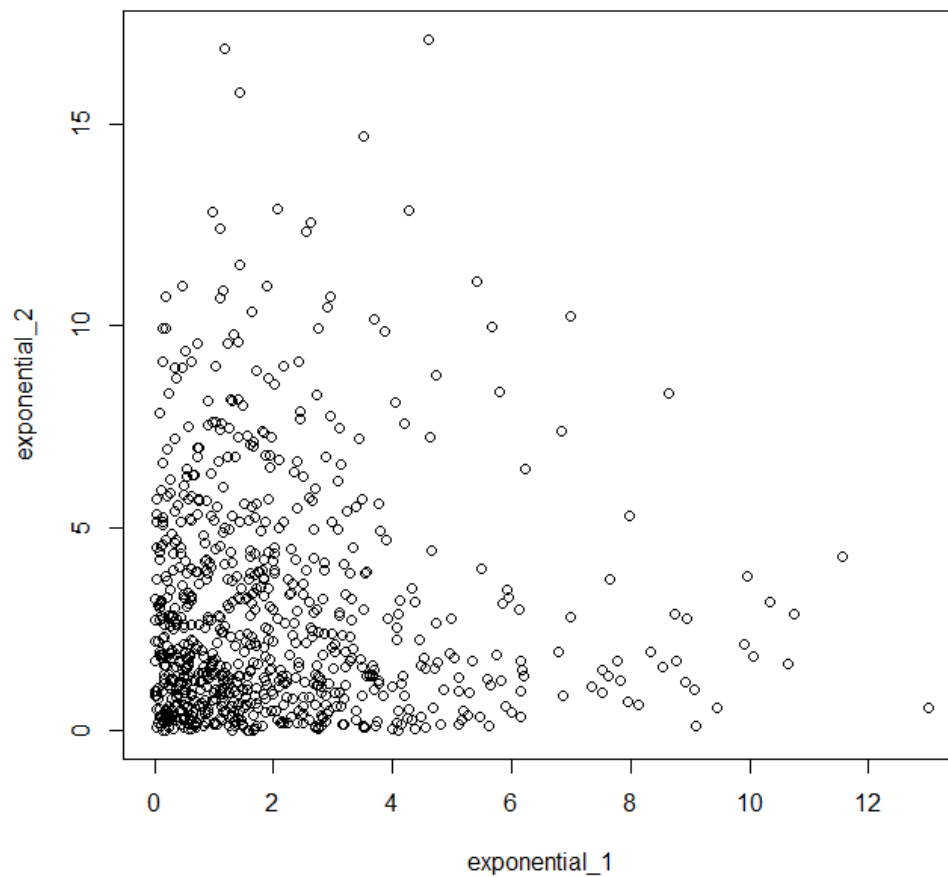
```
Sigma <- matrix(c(sigma1^2, rho * sigma1 * sigma2, rho * sigma1 * sigma2, sigma2^2), nrow = 2)
```

Exponenciális eloszlású minta generálása

```
# Exponenciális eloszlású minta generálása  
exponential_1 <- rexp(n, rate = 1/sigma1)  
exponential_2 <- rexp(n, rate = 1/sigma2)
```

Létrehozom a kétdimenziós mátrixot, illetve az ábrázolás

```
realizacio <- cbind(exponential_1, exponential_2)  
plot(realizacio )
```



## 4. Feladat

Seed beállítása és könyvtár betöltése

A `set.seed` függvénnyel egy véletlenszám-generátor kezdőértékét állítjuk be, hogy az eredmények reprodukálhatóak legyenek. A seed értéke  $ss + 139$ .

```
set.seed(ss+139)

library(stabledist)
```

Stabil eloszlású minta generálása

```
# Stabil eloszlású generátor
realizacio <- rstable(n, alpha = 1 + (ax + az) / (ax + ay + az), beta = 0, gamma = 100, delta = 11)
```

Realizáció ábrázolása

```
par(mfrow=c(2,2))

hist(realizacio, main="Histogram")

plot(density(realizacio), main="Sűrűségfüggvény")
```

t-próba végrehajtása

```
t_test <- t.test(realizacio)

print(t_test)
#a t_test XYZ valószínűségű student eloszlást mutat
```

Ferdeség és csúcsosság (skewness és kurtosis) kiszámítása

```
library(moments)

skewness_value <- skewness(realizacio)
print(skewness_value)

kurtosis_value <- kurtosis(realizacio)
print(kurtosis_value)

#a lapultságból és a ferdeségből tudjuk hogy ez nem normál eloszlás
```

## 5. Feladat

```
library(readr)
data <- read_csv("C:/Users/au087447/Desktop/Zavarkó_Máté_IN3BLK/GOOG.csv")
library(car)
# logreturn létrehozása
logreturn = c()

# A záró értékek kinyerése
close <- data$Close

# logreturn értékek számítása
for (i in 1:length(close)-1) {
  logreturn[i] = log(close[i+1]/close[i])
}

logreturn = abs(logreturn)

# Eloszlás
library(ggpubr)
ggdensity(logreturn)
qqPlot(logreturn)

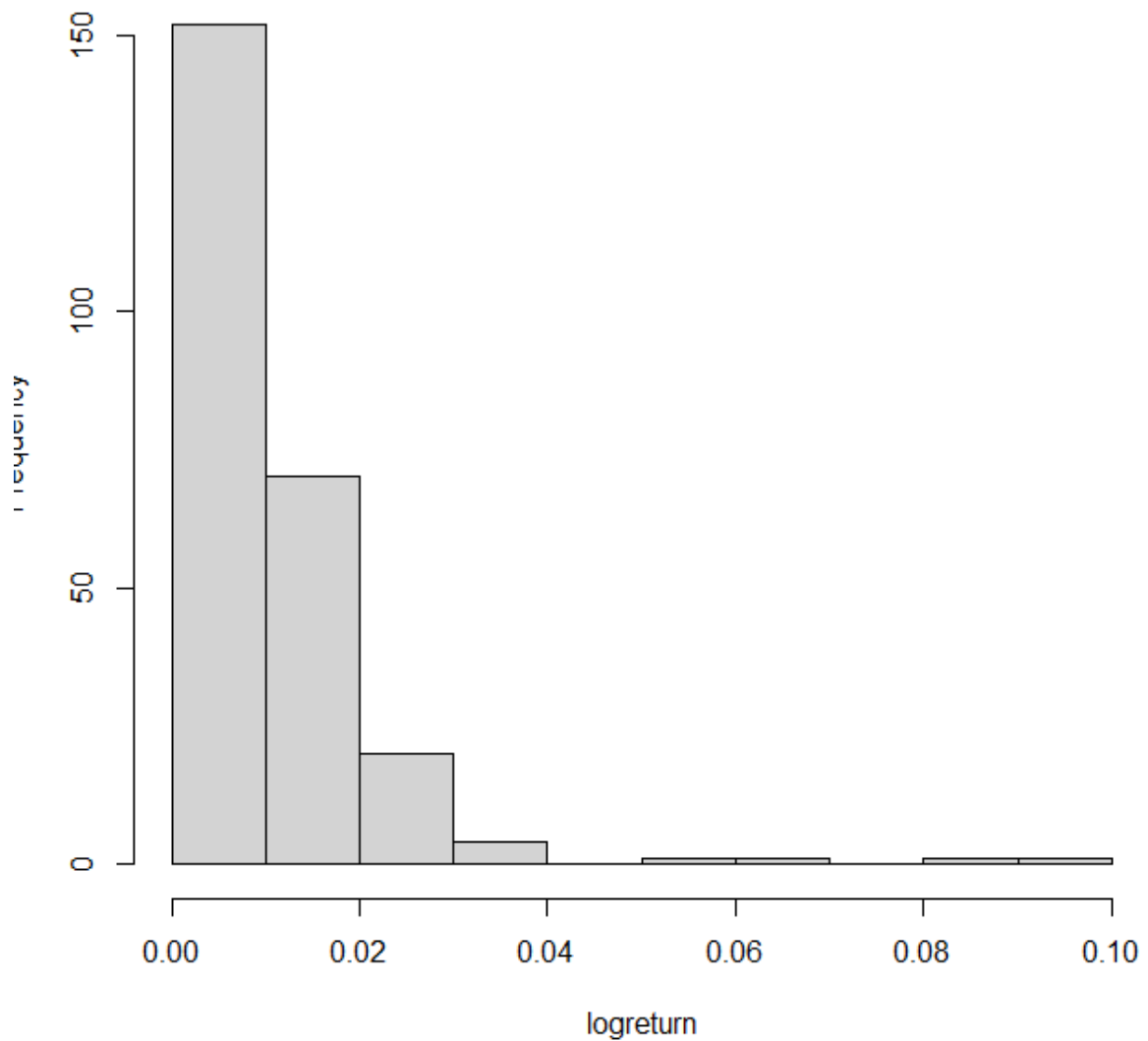
# Khi-négyzet teszt
chisq.test(logreturn)

ggdensity(logreturn)
#a sűrűség függvény alapján látszik a poisson eloszlás

hist(logreturn, main="Histogram")

t_test <- t.test(daily_changes)
print(t_test)
#a görbe 1 p-valueval mutat student eloszlást, de a qq plot alapján látszik hogy ez inkább normális eloszlású lehet
```

Histogram



Itt a read -el beolvasom a GOOG.csv file. Egy for ciklus segítségével feltöltöm a logreturn

tömböt.