DOCUMENTATIE PROIECT STATISTICA

Alexandrescu Mihaela-Isabelle
Balaban Teodor
Poenaru Gabriela-Catalina

February 3, 2021

1 Exercitiul 1

1. Generați 100 000 de valori dintr-o variabilă aleatoare folosind **metoda transformării inverse** pentru repartițiile definite mai jos:

a) *Repartiția logistică* are densitatea de probabilitate $f(x) = \frac{1}{\beta}\frac{e^{-(x-\mu)/\beta}}{(1+e^{-(x-\mu)/\beta})^2}$ și funcția de repartiție $F(x) = \frac{1}{1+e^{-(x-\mu)/\beta}}$.

b) *Repartiția Cauchy* are densitatea de probabilitate $f(x) = \frac{1}{\pi\sigma}\frac{1}{1+\left(\frac{x-\mu}{\sigma}\right)^2}$ și funcția de repartiție $F(x) = \frac{1}{2}+\frac{1}{\pi}\arctan\left(\frac{x-\mu}{\sigma}\right)$.

Comparați rezultatele obținute cu valorile date de funcțiile **rlogis** și respectiv **rcauchy**(funcțiile de repartiție predefinite în **R** pentru repartițiile logistică și respectiv Cauchy). Ilustrați grafic aceste rezultate.

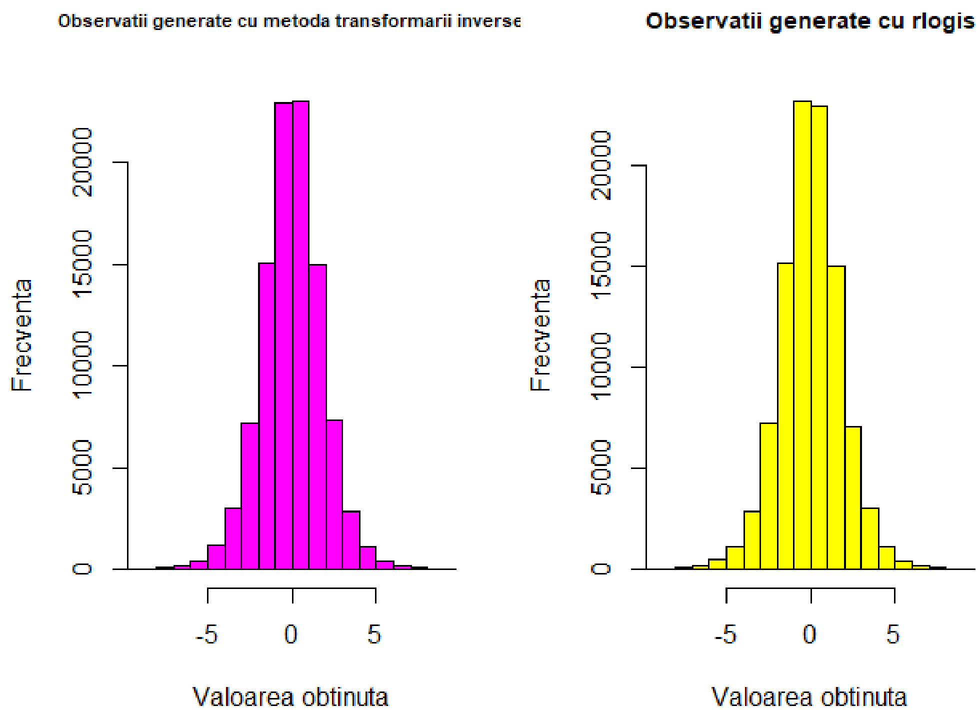
a) Functia de repartitie este continua. Putem calcula inversa, astfel:

$$
\begin{aligned}
&F(x) = u \\
&\frac{1}{1+e^{\frac{-(x-\mu)}{\beta}}} = u| \cdot (1+e^{\frac{-(x-\mu)}{\beta}}) \\
&u(1+e^{\frac{-(x-\mu)}{\beta}}) = 1 \\
&1+e^{\frac{-(x-\mu)}{\beta}} = \frac{1}{u} \\
&e^{\frac{-(x-\mu)}{\beta}} = \frac{1}{u} - 1|ln \\
&\frac{-(x-\mu)}{\beta} = ln(\frac{1}{u} - 1) \\
&x - \mu = -\beta ln(\frac{1-u}{u} \\
&x = -\beta(ln(1-u) - ln(u)) + \mu \\
&x = -\beta ln(1-u) + \beta ln(u) + \mu \\
&F^{-1}(u) = -\beta ln(1-u) + \beta ln(u) + \mu
\end{aligned}
\tag{1}
$$

Rezolvarea exercitiului in R:

Am luat n=100000 (nr de valori dintr-o v.a.). Am creat o functie (valori repLogistica) care are ca date de intrare n (nr de observatii), miu si beta si care returneaza inversa functiei de repartitie prin metoda transformarii inverse. Apoi am realizat histograma data de observatiile generate cu metoda transformarii inverse si histograma data de obsrvatiile generate cu functia predefinita rlogis.

```
n <- 100000

valori_repLogistica <- function(n, miu, beta)
{

  U <- runif(n)  #generam 100000 de observatii dintr-o uniforma
  return (miu-1/beta*log(1-U)+1/beta*log(U))  #returnam inversa functiei de repartitie
}
miu <- 0
beta <- 1
var_1L<-valori_repLogistica(n, miu, beta) #generarea cu ajutorul functiei create de noi
var_2L<-rlogis(n)                         #generarea cu ajutorul functiei prestabilite rlogis

par(mfrow=c(1,2)) #impartim spatiul de plotare in 2

hist(var_1L,main="Observatii generate cu metoda transformarii inverse",xlab="Valoarea obtinuta",
     ylab="Frecventa",xlim=c(-9, 9),col="magenta",cex.main=0.7)

hist(var_2L,main="Observatii generate cu rlogis",xlab="Valoarea obtinuta",
     ylab="Frecventa",xlim=c(-9, 9),col="yellow",cex.main=0.9)
```



Observam ca cele doua histograme sunt aproape identice, deci calculele noastre sunt corecte.

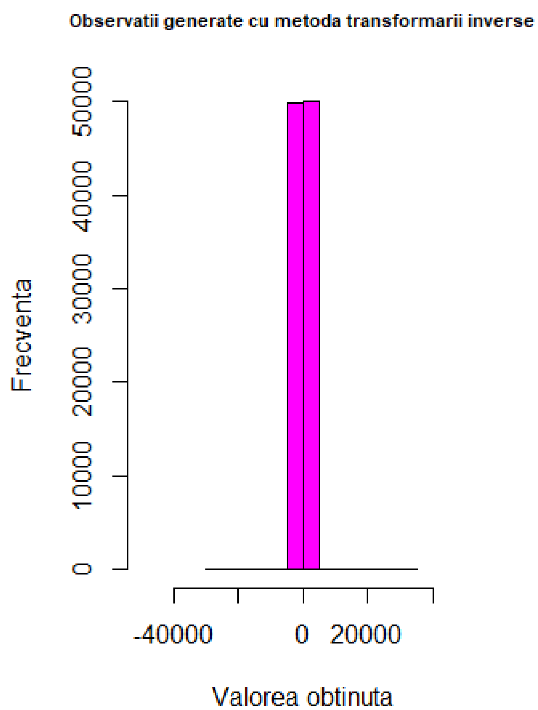
b) Procedam (analog) ca la a) si obtinem:

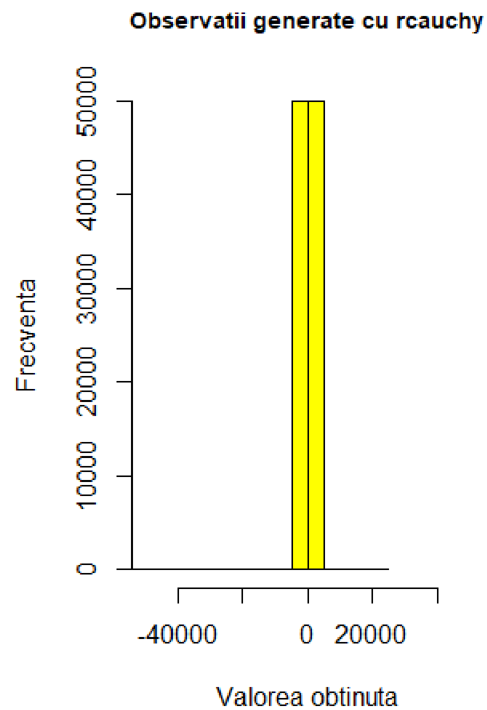
$$\begin{aligned}
&F(x) = u \\
&\frac{1}{2} + \frac{1}{\pi} arctg\left(\frac{x-\mu}{\sigma}\right) = u \\
&\frac{1}{\pi} arctg\left(\frac{x-\mu}{\sigma}\right) = u - \frac{1}{2} \\
&\frac{x-\mu}{\sigma} = tg\left(\pi(u - \frac{1}{2})\right) \\
&x - \mu = \sigma \cdot tg\left(\pi(u - \frac{1}{2})\right) \\
&x = \sigma \cdot tg\left(\pi(u - \frac{1}{2})\right) + \mu \\
&x = \sigma \cdot tg\left(\pi(\frac{2\pi - 1}{2})\right) + \mu \\
&F^{-1}(u) = \sigma \cdot tg\left(\pi(\frac{2\pi - 1}{2})\right) + \mu
\end{aligned} \tag{2}$$

Rezolvarea exercitiului in R:

La b) procedam analog pentru functia de repartitie Cauchy.

```
valori_repCauchy <- function(n, miu, sigma){

  U <- runif(n) #generam 100000 de observatii dintr-o uniforma
  return (miu+sigma*tan(pi*(U-1/2))) #returnam inversa functiei de repartitie
}
miu <- 0
sigma <- 1
var_1C <- valori_repCauchy (n,miu,sigma) #observatii generate cu metoda transformarii inverse
var_2C <- rcauchy(n,0,1) #observatii generate cu ajutorul functiei prestabilite rcauchy


par(mfrow=c(1,2)) #impartim spatiul de plotare in 2



hist(var_1C,main="Observatii generate cu metoda transformarii inverse", xlab="Valorea obtinuta",
     ylab="Frecventa", xlim=c(-50000,50000) ,col='magenta',cex.main=0.7)

hist(var_2C,main="Observatii generate cu rcauchy", xlab="Valorea obtinuta",
     ylab="Frecventa", xlim=c(-50000,50000),col='yellow',cex.main=0.9)
```





2 Exercitiul 2

2. Folosiţi **metoda respingerii** pentru a genera observaţii din densitatea de probabilitate definită prin $f(x) \propto \exp(-(x-2)^2/8)(\sin(2x)^2 - 2\cos(x)^2\sin(3x)^2 + 5)$ parcurgând paşii următori:

(**OBS**: Notaţia "$\propto$" înseamnă că $f(x)$ *este proporţional cu* expresia din dreapta)

a) Reprezentaţi grafic $f(x)$ şi arătaţi că aceasta este mărginită de $Mg(x)$ unde $g(x)$ este densitatea de probabilitate a repartiţiei normale de medie 2 şi dispersie 4. Determinaţi o valoare potrivită pentru constanta M, chiar dacă nu este optimă.
(**Indiciu**: Folosiţi funcţia *optimise* din **R**)

b) Generaţi 100000 de observaţii din densitatea de mai sus folosind metoda respingerii.

c) Deduceţi, pornind de la rata de acceptare a acestui algoritm, o aproximare a *constantei de normalizare* a lui $f(x)$, apoi comparaţi histograma valorilor generate cu reprezentarea grafică a lui $f(x)$ normalizată.

a) Stim ca f(x) este proportionala cu h(x), deci f(x)=c*h(x). Presupunem ca c=1, deci f(x)=h(x). Adica graficele coincid.

```
#stim ca functia f este proportonala cu h(asadar, f=c*h)
#luam c=1
#definim functia f
f <- function(x){
  return (exp(-(x-2)^2/8)*((sin(2*x))^2-2*(cos(x))^2*(sin(3*x))^2+5))
}
```

Densitatea de probabilitate a normalei este:

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{\frac{-(x-\mu)^2}{2\sigma^2}} \tag{3}$$

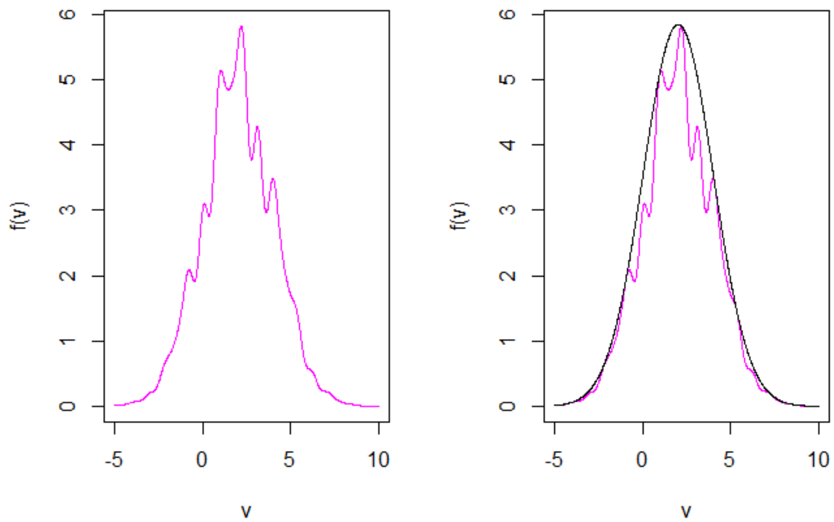
Cum media=2 si dispersia=4 ($\sigma = 2$), avem:

$$g(x) = \frac{1}{2\sqrt{2\pi}} \cdot e^{\frac{-(x-2)^2}{8}} \tag{4}$$

Pentru metoda respingerii ne uitam la raportul $\frac{h}{g}$ i.e. $2\sqrt{2\pi} \cdot + (sin^2(2x) - 2cos^2(x) \cdot sin^2(3x) + 5)$. Apoi cu functia optimise aflam M (valoarea maxima a acestui raport).

```
par(mfrow=c(1,2))
v <- seq(-5,10, 0.01) #discretizam intervalul -5,10 pentru a plota
plot(v, f(v), type="l", col = "magenta") #trasam graficul functiei f

#ne uitam la densitatea normalei de medie 2 si dispersie 4 notata cu g
#g=(exp(-(x-2)^2/8))/(2*sqrt(2*pi))

#definesc raportul f/g ca ulterior sa il maximizam pt a gasi constanta M
raport <- function(x)
{
  return ( 2*sqrt(2*pi) *((sin(2*x))^2-2*(cos(x))^2*(sin(3*x))^2+5))
}

#aflu valoarea maxima a acestui raport
M <- optimise(raport, c(0,2), maximum = TRUE)


v <- seq(-5,10, 0.01) #discretizam intervalul -5,10 pentru a plota
plot(v, f(v), type="l", col = "magenta") #trasam graficul functiei f
lines(v, dnorm(v,2,2)*M[[2]], col = "black") #suprapunem graficul g*M
```

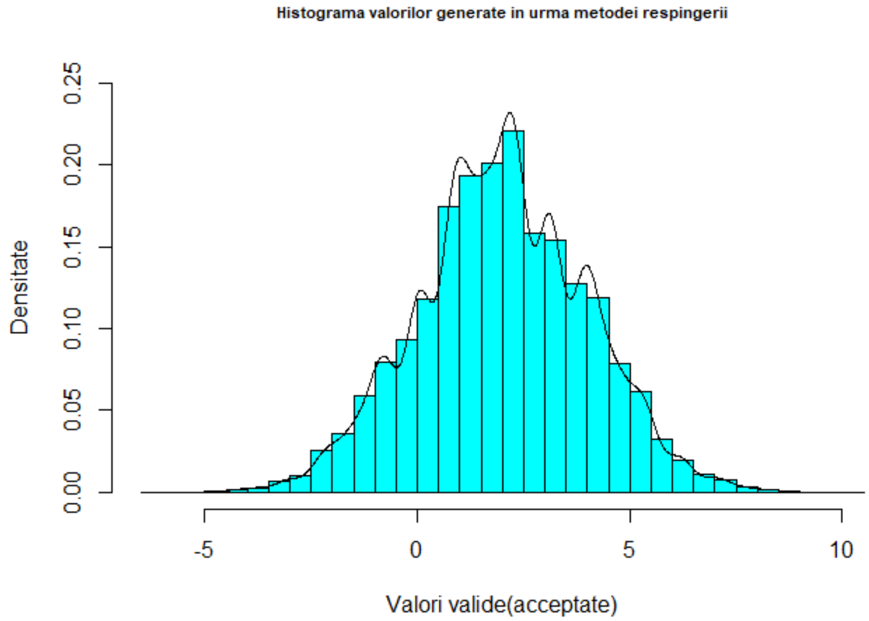
Trasam in acelasi grafic atat f, cat si $g \cdot M$ si putem observa ca normala margineste functia f.



b) Pentru b) aplicam metoda respingerii.

```
#generam 100000 observatii cu metoda respingerii
n <- 100000  #numar observatii
val_obtinuta <- c() #definim un vector gol in care stocam valorile gasite
k<- 0 #numar observatii valide
for(i in 1:n)
{

  u <- runif(1) #generam o observatie din uniforma
  x <- rnorm(1,2,2) #generam o observatie din normala de medie 2 si dispersie 4
  if( u <= h(x)/(M[[2]]*dnorm(x,2,2))) #conditia din algoritmul metodei respingerii
    {
    k <- k + 1 #contorizam numarul observatiilor valide pentru a calcula constanta c
    val_obtinuta[k] <- x #punem in vector valoarea x
    }
}
```

c) Pentru c contorizam "pasii de acceptare" ai algoritmului pentru ca stim din seminar ca acestia formeaza o variabila aleatoare repartizata geometric, de medie c.
Astfel, reusim sa obtinem constanta c (ea fiind media unei repartitii geometrice).
Realizam histograma valorilor generate, peste care trasam graficul functiei f(x) normalizate.



Histograma valorilor generate in urma metodei respingerii
Densitate
0.00
0.05
0.10
0.15
0.20
0.25
-5
0
5
10
Valori valide(acceptate)

3 Exercitiul 3

3. **Metoda Monte Carlo pentru aproximarea unor integrale**

Punctul de plecare al metodei Monte Carlo pentru aproximarea unei integrale este nevoia de a evalua expresia $\mathrm{E}_f(h(X)) = \int_{\chi} h(x)f(x)dx$, *unde* χ *reprezintă mulţimea de valori a variabile aleatoare X(care este, de obicei, suportul densităţii* f *).*

Principiul metodei Monte Carlo este de a aproxima expresia de mai sus cu media de selecţie $\overline{h_n} = \frac{1}{n}\sum_{j=1}^{n} h(x_j)$ *pornind de la un eşantion* $(X_1, X_2, \dots X_n)$ *din densitatea* f, *întrucât aceasta converge a.s. către* $\mathrm{E}_f(h(X))$, *conform legii numerelor mari. Mai mult, atunci când* $h^2(X)$ *are medie finită viteza de convergenţă a lui* $\overline{h_n}$ *poate fi determinată întrucât convergenţa este de ordin* $O(\sqrt{n})$ *iar varianţa aproximării este* $\mathrm{var}(\overline{h_n}) = \frac{1}{n}\int_{\chi}(h(x) - \mathrm{E}_f(h(X)))^2 f(x)dx$, *cantitate care poate fi de asemenea aproximată prin* $v_n = \frac{1}{n^2}\sum_{j=1}^{n}(h(x_j) - \overline{h_n})^2$.

Mai precis, datorită teoremei limită centrală, pentru un n *suficient de mare expresia*

$\frac{\overline{h_n} - \mathrm{E}_f(h(X))}{\sqrt{v_n}}$ *poate fi aproximată cu o normală standard, ceea ce conduce la posibilitatea construirii unui test de convergenţă şi a unor margini pentru aproximarea lui* $\mathrm{E}_f(h(X))$.

Cerinţă:

Pentru funcţia $h(x) = (1 - x^2)^{\frac{3}{2}}$ construiţi o aproximare a integralei acesteia pe intervalul [0,1] după cum urmează:

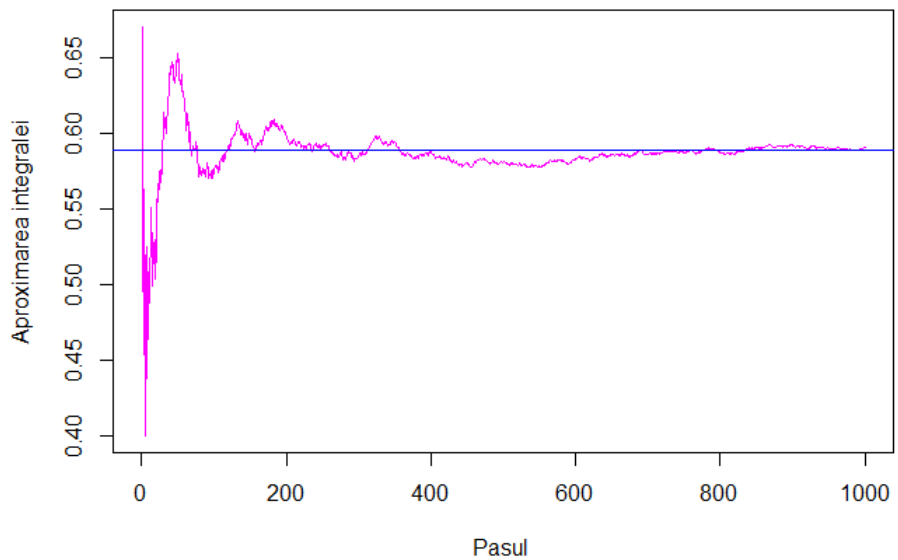
Valoarea integralei poate fi văzută ca fiind media funcţiei $h(X)$ unde X este repartizată uniform pe [0,1]. Urmărind algoritmul dat de metoda Monte Carlo construiţi programul **R** care determină aproximarea acestei integrale. Comparaţi rezultatul obţinut cu cel analitic. Ataşaţi reprezentările grafice pe care le consideraţi utile pentru a putea observa eficienţa metodei.

```r
n = 1000 #nr observatiilor

#functia h, a carei integrala vrem sa o aproximam
h = function(x) {
  return ( (1-x^2)^(3/2) )
}

s = 0    #initializam suma pt aproximarea integralei
int = c()

for( i in 1:n) {
  x = runif(1,0,1)            #generam cate o observatie din uniforma
  s = s + h(x)
  int[i] = s/i                #aici calculam integrala la fiecare iteratie(pentru fiecare observatie)
}
MonteCarlo = s/n              #aproximarea finala a integralei

plot(int, type = "l",col = "magenta",xlab = " Pasul  ",ylab = "Aproximarea integralei")
abline(h=0.589, col='blue')     #valoarea integralei calculata de mana
```

Calculam integrala:

$$\begin{aligned}
&I = \int_0^1 \left(1-x^2\right)^{\frac{3}{2}} dx \\
&\text{Facem S.V. x=sin(y)} \\
&I = \int_0^{\frac{\pi}{2}} cos(y)^4 dy \\
&= \int_0^{\frac{\pi}{2}} \frac{1+cos(2y)}{2} \cdot \frac{1+cos(2y)}{2} dy \\
&= \frac{1}{4} \int_0^{\frac{\pi}{2}} (1+2cos(2y)+cos(2y)^2) dy \\
&= \frac{1}{4} \int_0^{\frac{\pi}{2}} (1+2cos(2y)+\frac{1+cos(4y)}{2}) dy \\
&= \frac{1}{4} \left(y + sin(2y) + \frac{1}{2}y + \frac{1}{8} sin(4y) \right) |_0^{\frac{\pi}{2}} \\
&= \frac{1}{4} \cdot \frac{3\pi}{4} \\
&= \frac{3\pi}{16} \\
&\approx 0.589
\end{aligned} \tag{5}$$

Se observa din grafic ca aproximarea este din ce in ce mai fina cu cat creste numarul de iteratii.



4 Exercitiul 4

4. Construiți două funcții în **R** ***frcpois*** și respectiv ***frcexp*** care să calculeze **marginea inferioară Rao-Cramer (MIRC)** pentru varianța estimatorilor parametrilor repartițiilor ***Poisson*** și respectiv ***Exponențială*** pentru un eșantion de dimensiune n (generați voi un asemenea eșantion într-o manieră corespunzătoare și folosiți-l în apelul funcției!).

- Functia frcpois:

Functia de masa a repartitiei Poisson este $f_\lambda(x) = \dfrac{\lambda^x \cdot e^{-\lambda}}{x!}$
Aplicam logaritm natural:

$$
\begin{aligned}
&ln\left(f_\lambda(x)\right) = ln\left(\frac{\lambda^x}{e^\lambda \cdot x!}\right) = \\
&= x \cdot ln(\lambda) - \lambda \cdot ln(e) - ln(x!) = \\
&= x \cdot ln(\lambda) - \lambda - ln(x!)
\end{aligned} \tag{6}
$$

Derivam in functie de λ de doua ori:

$$
\begin{aligned}
\frac{\partial}{\partial\lambda} ln\left(f_\lambda(x)\right) &= \frac{x}{\lambda} - 1 \\
\frac{\partial^2}{\partial\lambda^2} ln\left(f_\lambda(x)\right) &= -\frac{x}{\lambda^2}
\end{aligned} \tag{7}
$$

$$
\begin{aligned}
&E\left[\frac{\partial^2}{\partial\lambda^2} ln\left(f_\lambda(x)\right)\right] = E\left[-\frac{x}{\lambda^2}\right] = \\
&= -\frac{1}{\lambda^2} \cdot E\left[x\right] \\
&= -\frac{1}{\lambda^2} \cdot \lambda \\
&= -\frac{1}{\lambda}
\end{aligned} \tag{8}
$$

In calculele de mai sus ne-am folosit de faptul ca $E[c \cdot x] = c \cdot E[x]$ si $E[x] = \lambda$ (pentru ca x e repartizata Poisson).

Acum inlocuim ce am calculat mai sus in formula pentru MIRC:

$$
\begin{aligned}
&MIRC_{pois} = \frac{1}{-n \cdot E\left[\frac{\partial^2}{\partial\lambda^2} ln\left(f_\lambda(x)\right)\right]} = \\
&= \frac{1}{-n \cdot \frac{-1}{\lambda}} \\
&= \frac{\lambda}{n}
\end{aligned} \tag{9}
$$

In R am luat un exemplu ca sa ne putem verifica. Am luat X∼Poisson(5). Am luat n=100 si $\lambda = 5$.
In acest caz, $MIRC = \frac{\lambda}{n} = \frac{5}{100} = 0.05$, aproximativ cat ne-a dat si in urma

rularii functiei frcpois in R.

```
MIRC                    0.0505050505050505
```

```
frcpois <- function(n,lambda,x)
{
  der2 <- function(lambda,x)
{
    (lambda^((x -1)-1)*(x- 1)*x* exp(-lambda) - lambda^(x-1)*x*exp(-lambda)-(lambda^(x-1)*
     x*exp(-lambda)-lambda^x*exp(-lambda)))/factorial(x)/(lambda^x*exp(-lambda)/factorial(x))-
      (lambda^(x-1)*x*exp(-lambda)-lambda^x*exp(-lambda))/factorial(x)*
((lambda^(x-1)*x*exp(-lambda)-lambda^x*exp(-lambda))/factorial(x))/(lambda^x*exp(-lambda)/factorial(x))^2
  }
  #derivata de ordin doi introdusa manual
MIRC <- 1/(-n*mean(eval(der2(lambda,x))))
return(MIRC)
}

p <- rpois(n,5)
MIRC <- frcpois(n,5,p)
```

- Functia frcexp:

Functia de densitate a repartitiei Exponentiale este:

$$f(x,\lambda) = \begin{cases} \lambda \cdot e^{-\lambda x} & ,x \geq 0 \\ \\ 0 & ,x < 0 \end{cases} \tag{10}$$

Aplicam logaritm natural:

$$ln\left(f_\lambda(x)\right) = ln\left(\lambda \cdot e^{-\lambda x}\right) = ln(\lambda) - \lambda x \tag{11}$$

Derivam de doua ori in functie de λ:

$$
\begin{aligned}
&\frac{\partial}{\partial \lambda}\left(f_\lambda(x)\right) = \\
&= \frac{\partial}{\partial \lambda}\left(ln(\lambda) - \lambda x\right) \\
&= \frac{1}{\lambda} - x \qquad (12) \\
&\frac{\partial^2}{\partial \lambda^2}\left(\frac{1}{\lambda} - x\right) = \\
&= -\frac{1}{\lambda^2}
\end{aligned}
$$

$$
E\left[\frac{\partial^2}{\partial \lambda^2}\left(f_\lambda(x)\right)\right] = E\left[-\frac{1}{\lambda^2}\right] = -\frac{1}{\lambda^2} \qquad (13)
$$

Inlocuim ce am calculat mai sus in formula pentru MIRC:

$$
\begin{aligned}
MIRC_{exp} &= \frac{1}{-n \cdot E\left[\frac{\partial^2}{\partial \lambda^2}\left(f_\lambda(x)\right)\right]} = \\
&= \frac{1}{-n \cdot -\frac{1}{\lambda^2}} \qquad (14) \\
&= \frac{\lambda^2}{n}
\end{aligned}
$$

In R am luat un exemplu ca sa ne putem verifica. Am luat X~Exp(3). Am luat n=100 si $\lambda = 3$.
In acest caz, $MIRC = \frac{\lambda^2}{n} = \frac{9}{100} = 0.09$, aproximativ cat ne-a dat si in urma rularii functiei frcexp in R.

```
MIRC          0.090000000000001
```

```
frcexp <- function(n,X,lambda)
{
  deriv2 <- function(lambda,X) {

    -((exp(-lambda*X)*X+((exp(-lambda*X)*X)-lambda*(exp(-lambda*X)*X^2)))/
        (lambda*exp(-lambda*X))+(exp(-lambda*X)-lambda*(exp(-lambda*X)*X))*(exp(-lambda*X)-
      lambda*(exp(-lambda*X)*X))/(lambda*exp(-lambda*X))^2)
  }
  #derivata de ordin 2 introdusa manual, din nou
  MIRC <- 1/(-n*mean(eval(deriv2(lambda,X))))
  return(MIRC)
}

e <- rexp(n)
MIRC <- frcexp(n,e,3)
```

5 Exercitiul 5

5. a) Construiți o funcție care calculează media unei variabile aleatoare, primind ca parametri de intrare tipul variabilei aleatoare(discretă sau continuă) și densitatea/funcția de masă, după caz. Comparați cu rezultatul teoretic.

b) Construiți o funcție care calculează dispersia unei variabile aleatoare, folosindu-vă de funcția care calculează media. Comparați cu rezultatul teoretic.

c) Construiți o funcție care calculează funcția generatoare de momente pentru variabila aleatoare X pentru un interval dat(discretizat de un vector t, care este parametru de intrare pentru funcție).

d) Aproximați media și dispersia variabilei aleatoare X folosind funcția generatoare de momente de la c). Comparați cu rezultatele obținute la a) și b).

- Cazul discret:

$$X \sim \begin{pmatrix} x_1 & x_2 & ... & x_n \\ p_1 & p_2 & ... & p_n \end{pmatrix} \tag{15}$$

a) Media unei variabile aleatoare discrete X e definita astfel:

$$E[X] = x_1p_1 + ... + x_np_n = \sum_{i=1}^{n} x_ip_i \tag{16}$$

Varianta:

$$Var(X) = E[X^2] - E[X]^2 \tag{17}$$

b) Dispersia(deviatia standard):

$$Dispersia = \sqrt{Var(X)} \tag{18}$$

c) Functia generatoare de momente pentru cazul discret are urmatoarea formula:

$$FGM(t) = \sum_{i=1}^{n} p_i e^{tx_i}, \tag{19}$$

unde t este vectorul care discretizeaza intervalul.

- Cazul continuu:

  a) Media unei variabile aleatoare continue X este:

$$E[X] = \int_{-\infty}^{\infty} x \cdot f(x)dx, \tag{20}$$

unde f(x) este densitatea de probabilitate lui X.

Varianta:

$$Var(X) = E[X^2] - E[X]^2 = \int_{-\infty}^{\infty} x^2 \cdot f(x)dx - E[X]^2 \tag{21}$$

b) Dispersia(deviatia standard):

$$Dispersia = \sqrt{Var(X)} \tag{22}$$

c) Functia generatoare de momente pentru cazul continuu are urmatoarea formula:

$$FGM(t) = \int_{-\infty}^{\infty} e^{tx} \cdot f(x)dx, \tag{23}$$

unde f(x) este densitatea de probabilitate a lui X si t este vectorul care discretizeaza intervalul.

In R am luat exemplul pentru $X \sim Exp(\lambda)$ cu $\lambda = 5$. N-am reusit sa calculez in R functia generatoare de momente. Aceasta era functia initiala:

```
FGM_continuu <- function(t)
{ phi2 <- c()
  for(i in 1:length(t))
  { f3 <- function(x) {return(exp(t[i]*x)*lambda*exp(-lambda*x))} #functia din interiorul integralei pt FGM
    phi2[i] <- integrate(f3,lower=0,upper=Inf)}
   return(phi2)
}
phi2=FGM_continuu(t)
```

Am notat cu phi2=functia generatoare de momente pt cazul continuu, FGMcontinuu=functia in care vreau sa calculez phi2, f3=functia in care aflu integrandul pentru fiecare t[i].
Eroarea care imi apare este aceasta:

```
Error in integrate(f3, lower = 0, upper = Inf) :
  non-finite function value
In addition: There were 50 or more warnings (use warnings() to see the first 50)
```

Asa ca am calculat de mana functia generatoare de momente:

$$
\begin{aligned}
&MGF(t) = \int_0^\infty e^{tx} \cdot \lambda \cdot e^{-\lambda x} dx \\
&= \lambda \int_0^\infty e^{x(t-\lambda)} dx \\
&S.V.: x(t-\lambda) = y, x = 0 \longrightarrow y = 0, x = \infty \longrightarrow y = -\infty \\
&= \lambda \cdot \int_0^{-\infty} e^y \cdot \frac{1}{t-\lambda} dy \\
&= ... = \frac{\lambda}{\lambda - t} \qquad\qquad , \lambda > t
\end{aligned}
\tag{24}
$$

d) Legatura dintre functia generatoare de momente si medie si dispersie:

$$
\begin{aligned}
&E[X] = MGF'(0) \\
&E[X^2] = MGF''(0) \\
&Var(X) = MGF''(0) - (MGF'(0))^2 \\
&Dispersia = \sqrt{MGF''(0) - (MGF'(0))^2}
\end{aligned}
\tag{25}
$$

Pentru cazul discret, functia generatoare de momente ne da:

6 Exercitiul 6

6. a) Pentru repartițiile ***logistică*** și respectiv ***Cauchy***(*vezi* problema 1) construiți *funcția de verosimilitate* pentru parametrul μ considerând că parametrii β și respectiv σ sunt cunoscuți(alegeți valori potrivite pentru aceștia).

b) Reprezentați grafic funcțiile de verosimilitate pentru cele două cazuri și folosind funcția ***optimise*** determinați o estimație pentru μ în baza unui eșantion de dimensiune 1000 pe care l-ați construit în prealabil. Explicați modul în care ați generat valorile din eșantion. Comentați și interpretați rezultatele.

a) Stim de la 1.a) inversa functiei de repartitie pentru repartitia logistica:

$$F^{-1}(u) = -\beta \cdot ln(1-u) + \beta \cdot ln(u) + \mu \tag{26}$$

Functia de verosimilitate are formula:

$$L_{logis}(\mu/(x_1, ..., x_n)) = \prod_{i=1}^{n} f_\mu(x_i) = \prod_{i=1}^{n} \frac{1}{\beta} \cdot \frac{e^{\dfrac{-(x-\mu)}{\beta}}}{\left(1 + e^{\dfrac{-(x-\mu)}{\beta}}\right)^2}, \tag{27}$$

unde: f_μ este densitatea data la exercitiul 1, iar β este fixat.

Rezolvarea in R:

```
#generam valori din repartitia logistica cu functia predefinita rlogis

n <- 100
x <-rlogis(n,0,1) #x este un vector de 100 de elemente
                  #ce va contine valori din repartitia logistica

L <- 1; #initializam produsul cu 1
beta <- 1; #alegem o valoare potrivita pentru beta fixat


#mai jos vom calcula functia de verosimilitate
#miu este parametrul nostru(theta din formulele din curs)

verosim_logi <- function(miu)
{
  for(i in 1:n)
  {
    L <- L*(1/beta)*exp(-(x[i]-miu)/beta)/(1+exp(-(x[i]-miu)/beta))^2;
  }
  return(L)
}
```

Inversa functiei de repartitie pentru repartitia Cauchy este:

$$F^{-1}(u) = \sigma \cdot tg\left(\pi\left(\frac{2\pi - 1}{2}\right)\right) + \mu \tag{28}$$

Functia de verosimilitate are formula:

$$L_{Cauchy}(\mu/(x_1, ..., x_n)) = \prod_{i=1}^{n} f_\mu(x_i) = \prod_{i=1}^{n} \frac{1}{\pi\sigma} \cdot \frac{1}{1 + \left(\frac{x-\mu}{\sigma}\right)^2}, \tag{29}$$

unde: f_μ este densitatea data la exercitiul 1, iar σ este fixat.

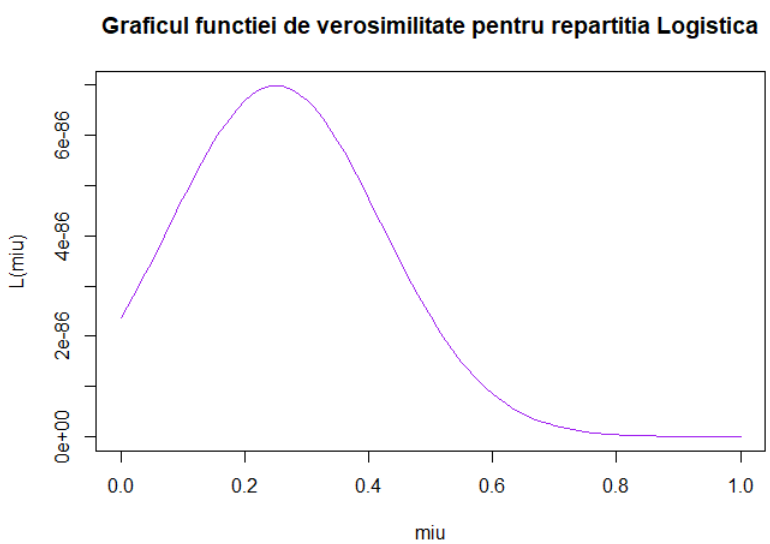
Rezolvarea in R:

```
#procedam exact ca mai sus de data aceasta pentru repartitia Cauchy


y<- rcauchy(n,0,1)
sigma <- 1;
L <- 1
verosim_cauchy <- function(miu){
  for(i in 1:n){
    L <- L*(1/(pi*sigma))*1/(1+((y[i]-miu)/sigma)^2)
  }
  return(L)
}
```

b)

• Codul in R pentru graficul repartitiei logistice:

```
#trasam graficul functiei de verosimilitate pentru repartitia Logistica
plot(verosim_logi,xlab='miu',ylab='L(miu)', col="purple")
title("Graficul functiei de verosimilitate pentru repartitia Logistica")
```



Cautam valoarea optima pentru μ :

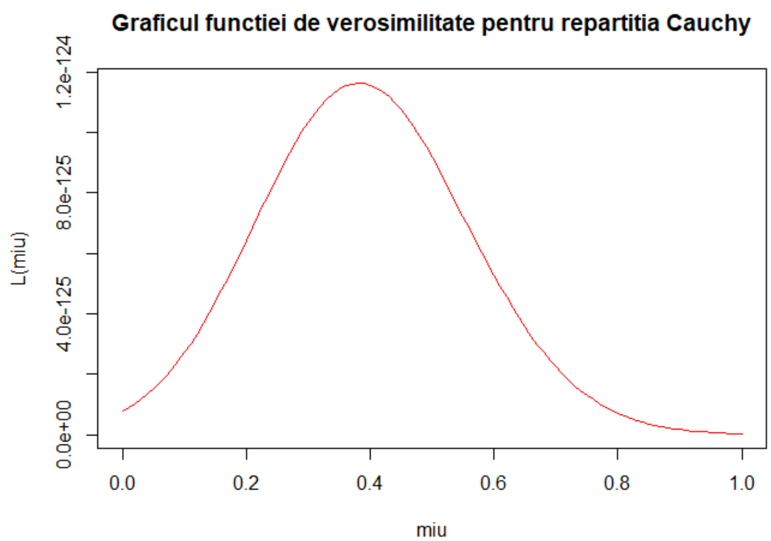
```
miu_logis <- optimise(verosim_logi,c(0,1),maximum=TRUE)
miu_optim <- miu_logis[[1]]
```

Valoarea este:

miu_optim	0.250254283271973

- Codul in R pentru graficul repartitiei Cauchy:

```
plot(verosim_cauchy,xlab='miu',ylab='L(miu)', col="red")
title("Graficul functiei de verosimilitate pentru repartitia Cauchy")
```



Cautam valoarea optima pentru μ :

```
miu_cauchy <- optimise(verosim_cauchy,c(0,1),maximum=TRUE)
miu_optim2 <- miu_cauchy[[1]]
```

Valoarea este:

miu_optim2	0.381527882906245

Interpretarea celor 2 grafice:
Dupa ce am generat mai multe grafice, am observat ca atat functia de verosimilitate pentru repartitia logistica, cat si cea pentru repartitia Cauchy

se apropie de 0 cu cat marim esantionul.
Referitor la constructia parametrului nostru, (μ), constatam ca aceasta este posibila pe un interval restrans(bucla din imagine) pentru ca, dupa aceea, μ are valoarea constant egala cu 0.