

## Proiectul 3: Câteva aplicații cu variabile aleatoare în R Studio

### Echipa:

**Lider :** Barbu Mariana 233

Balu Ionut Valentin 233

Cretu Daria Stefana 233

Cantu Denisa Maria 233

Am avut de implementat 4 exercitii despre variabile aleatoare și statistica descriptiva.

La primul exercitiu, avand doua variabile aleatoare, am citit valorile corespunzatoare de la tastatura cu ajutorul functiei readline, apoi le-am transformat în valori întregi cu ajutorul functiei as.integer.

```
#citim n și m
n <- readline(prompt="Introduceti n: ")
n <- as.integer(n)
m <- readline(prompt="Introduceti m: ")
m <- as.integer(m)
#citim variabila aleatorie X
vals <- c()
probabs <- c()

for(val in c(1:n))
{
  x <- readline(prompt="Introduceti x: ")
  x <- as.integer(x)
  p1 <- readline(prompt="Introduceti numarator p: ")
  p1 <- as.integer(p1)
  p2 <- readline(prompt="Introduceti numitor p: ")
  p2 <- as.integer(p2)
  p <- fractions(p1/p2)
  vals <- append(vals,x)
  probabs <- append(probabs,p)
}

X <- RV(vals,probabs)
X_prob <- fractions(probs(X))
```

```

#citim variabila aleatoare Y
vals <- c()
probabs <- c()

for(val in c(1:m))
{
  x <- readline(prompt="Introduceti x: ")
  x <- as.integer(x)
  p1 <- readline(prompt="Introduceti numarator p: ")
  p1 <- as.integer(p1)
  p2 <- readline(prompt="Introduceti numitor p: ")
  p2 <- as.integer(p2)
  p <- fractions(p1/p2)
  vals <- append(vals,x)
  probabs <- append(probabs,p)
}
Y <- RV(vals,probabs)
Y_prob <- fractions(probs(Y))

```

Am rezolvat punctul a), calculand repartitia comuna a celor doua variabile aleatoare, dar lasand necompletata ultima linie. Am obtinut astfel o repartitie comuna incompleta.

Am construit o matrice care va memora repartitia comuna.

```

frepcomgen <- function(x,y)
{
  a <- matrix(0,nrow=x,ncol=y)
  for (val in c(1:x-1))
  {
    for(val1 in c(1:y))
    {
      a[val,val1] <- fractions(X_prob[val]*Y_prob[val1])
    }
  }
  return (fractions(a))
}
rep_com <- frepcomgen(n,m)

```

Pentru punctul b), am parcurs matricea dupa coloane și am facut suma fiecareia pentru a putea obtine valorile finale de pe ultima linie.

```
#b) completam repartitia comuna
fcomplrepcom <- function(matr1)
{
  matr <- matrix(0,nrow=n,ncol=m)
  matr <- matr1
  for(val in c(1:ncol(matr)))
  {
    sum <- 0
    for(val1 in c(1:(nrow(matr)-1)))
    {
      sum <- sum+matr[val1,val]
    }
    cnt <- (Y_prob[[val]])-sum
    matr[nrow(matr),val] <- cnt
  }
  return(matr)
}
rep_com <- fcomplrepcom(rep_com)
```

Punctul c), avand repartitia comuna pentru X și Y, trebuia sa calculam în primul rand covarianta dintre  $2X+3$  și  $4Y-11$ .

Pe baza formulei am obtinut:

```
#c)
#1)
#Cov( $2X+3$ ,  $4Y-11$ )= $E((2X+3)*(4Y-11))-E(2X+3)*E(4Y-11)$ 
#Cov( $2X+3$ ,  $4Y-11$ )= $2*4*\underline{\text{Cov}}(X,Y)+\underline{\text{Cov}}(2X,-11)+\underline{\text{Cov}}(3,4Y)+\underline{\text{Cov}}(3,-11)$ 
#Cov( $2X+3$ ,  $4Y-11$ )= $8\underline{\text{Cov}}(X,Y)$ 
#Cov( $X,Y$ )= $E(X*Y)-E(X)*E(Y)$ 
```

Am calculat prima data inmultirea celor 2 variabile dupa regula inmultirii:

```
#X*Y
inmultire <- function(X,Y)
{
  vals <- c()
  probabs <- c()
  for(val1 in c(1:n))
  {
    for(val2 in c(1:m))
    {
      cnt <- X[val1]*Y[val2]
      vals <- append(vals,cnt)
      buff <- rep_com[val1,val2]
      probabs <- append(probabs,buff)
    }
  }
  XY <- RV(vals,probabs)
  return(XY)
}
XY <- inmultire(X,Y)
XY_prob <- fractions(probs(XY))
```

Mai departe, am implementat o functie prin intermediul careia calculam media unei variabile:

```
#functia pt medie
media <- function(X,X_prob)
{
  sum <- 0
  for(val in c(1:length(X_prob)))
  {
    cnt <- (X_prob[[val]])*(X[val])
    sum <- sum + cnt
  }
  return(fractions(sum))
}
```

În final, am calculat media produsul și produsul mediilor celor doua variabile, urmand sa le inlocuim în formula finala:

```
EX <- media(X,X_prob)
EY <- media(Y,Y_prob)
EXY <- media(XY,XY_prob)

#Cov(X,Y)=E(X*Y)-E(X)*E(Y)
Cv <- EXY-(EX*EY)
#Cov(2X+3,4Y-11)=8Cov(X,Y)
Cov <- 8*Cv
```

În al doilea rând, a trebuit să calculăm  $P(0 < X < 8 \mid Y > 3)$ .

Am luat două variabile aleatoare pentru care am calculat probabilitatea:

```
#2) P(0 < X < 8 | Y > 3)
X1 <- RV(c(4,5),c(1/2,1/2))
X2 <- RV(c(1,2,3,4),c(1/4,1/4,1/4,1/4))
P1 <- P((X1>2)%AND%(X1<8))%AND%(X2>3))
P2 <- P(X2>3)
P_final=P1/P2
```

Pentru ultimul subpunct, unde aveam de calculat  $P(X > 2, Y < 7)$ , am aplicat același lucru:

```
#3) P(X>2, Y<7)
P((X1>2)%AND%(X2<7))
```

La punctul d), am verificat dacă variabilele sunt independente cu ajutorul funcției independent și am printat răspunsul:

```
#d)
fverind <- function(X,Y)
{
  if(independent(X,Y)==TRUE)
    print("Variabilele sunt independente!")
  else
    print("Variabilele sunt dependente!")
}
fverind(X,Y)
```

Pentru a verifica daca variabilele sunt corelate sau nu, am calculat coeficientul de corelatie Pearson :

```
#coeficientul de corelatie Pearson :  $RO(X,Y)=COV(X,Y)/SQRT(Var(X)*Var(Y))$ 
fvernecor <- function(X,Y)
{
  numitor <- sqrt(V(X)*V(Y))
  pearson <- Cov/numitor

  if(pearson==0)
    print("Variabilele sunt necorelate!")
  if((abs(pearson)>0) %AND% (abs(pearson)<0.25))
    print("Variabilele sunt slab corelate!")
  if((abs(pearson)>=0.25) %AND% (abs(pearson)<=0.75))
    print("Variabilele sunt corelate!")
  if((abs(pearson)>0.75) %AND% (abs(pearson)<=1))
    print("Variabilele sunt puternic corelate!")
  if(abs(pearson)>1)
    print("Nu avem suficiente informatii!")
}
fvernecor(X,Y)
```

La exercitiul 2), primind doua densitati de probabilitate și optiunea utilizatorului, trebuie sa afisam, dupa caz, suma, diferenta, media și dispersia sau reprezentarea grafica.

Citim de la tastatura parametri functiilor care implementeaza densitatile și optiunea utilizatorului:

```
#EX2
print("Introduceti densitatile de probabilitate a doua variabile aleatoare continue independente si una dintre urmatoarele optiuni!")
x <- readline(prompt="Introduceti valoarea lui x pentru prima densitate de probabilitate : ")
x <- as.numeric(x)

x1 <- readline(prompt="Introduceti valoarea lui x pentru a doua densitate de probabilitate : ")
x1 <- as.numeric(x1)

x2 <- readline(prompt="Introduceti valoarea lui y pentru formula de convolutie : ")
x2 <- as.numeric(x2)

optiune <- readline(prompt="Optiuni :
1 pentru suma variabilelor aleatoare
2 pentru diferenta variabilelor aleatoare
3 pentru media si dispersia celor 2 variabile aleatoare
4 pentru reprezentarea grafica, in acelasi reper, a celor 2 densitati, cu culori diferite")
optiune <- as.integer(optiune)
```



Am implementat o functie care contine cele doua densitati de probabilitate și calculeaza suma folosind formula de convolutie.

```
ex2 <- function(x1,x, x2, optiune)
{
  dens11 <- function (x, x2)
  {
    if(0 < x && x < 2) return (3/8*(4*(x-x2) - 2*((x-x2)**2)))
    else return (0)
  }

  dens1 <- function (x)
  {
    if(0 < x && x < 2) return (3/8*(4*(x) - 2*((x)**2)))
    else return (0)
  }

  dens2 <- function (x)
  {
    if(0 <= x && x <= 1) return ((exp(1)/(exp(2)-1))*(exp(-x) + exp(x)))
    else return (0)
  }

  suma <- function(a)
  {
    rez <- integrate (dens11, 0, 2, x2 = a)$value * integrate(dens2, 0, 1)$value
    return (rez)
  }
}
```

În aceeași funcție am implementat și funcția reprezentare care reprezintă graficele celor două densități în același reper.

Aceasta primește câte 10 valori random pentru fiecare densitate de probabilitate și folosim plot și lines, pentru a reprezenta cele două grafice în același reper :

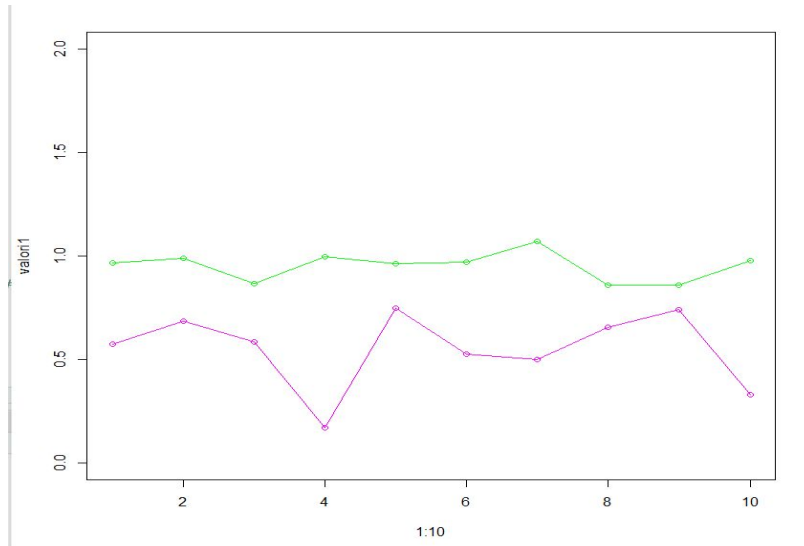
```
reprezentare <- function()
{
  valori1 <- runif(10, 0, 2)
  valori2 <- runif(10, 0, 1)

  for(val in c(1:10))
  {
    x <- dens1(valori1[val])
    valori1[val] <- x

    y <- dens2(valori2[val])
    valori2[val] <- y_
  }

  plot(1:10, valori1, col="magenta",ylim=c(0,2), type = "o")
  lines(1:10, valori2, col="green", ylim=c(0,2), type = "o")
}
```

Un exemplu de grafic ar fi :



Pentru optiunea 1, apelam functia suma, pentru optiunea 3, calculam media si dispersia cu ajutorul functiilor mean și var, iar pentru optiunea 4 apelam functia reprezentare:

```
if(optiune==1)
{
    print(suma(0.1))
}
else if(optiune==2)
{
}
else if(optiune==3)
{
    print(mean(c(dens1(x), dens2(x1))))
    print(var(c(dens1(x), dens2(x1))))
}
else if(optiune==4)
{
    reprezentare()
}
else
{
    print("Introduceti o optiune valida!")
}
```



La exercitiul 3), trebuia sa implementam un generator de numere aleatoare dupa anumite conditii, astfel:

Intrucat ipoteza specifica restrictia de a parcurge de maxim 3 ori ciclul dat, am implementat un while care se opreste dupa am parcurs a treia oara.

Primul pas a fost sa extragem ora în ore, minute și secunde cu ajutorul functiei `format(Sys.time(), "%H/M/S")` :

```
#1)
genx1 <- function()
{
  nr <- 3
  x1 <- NULL
  while(nr>0)
  {
    #calculam t1
    ore <- format(Sys.time(), "%H")
    ore <- as.integer(ore)
    minute <- format(Sys.time(), "%M")
    minute <- as.integer(minute)
    secunde <- format(Sys.time(), "%S")
    secunde <- as.integer(secunde)
    t1 <- minute*100+secunde
    t <- t1%%12
  }
}
```

Pentru fiecare conditie din enunt, am urmat pasii și am generat primul numar din sir. În cazul în care am intrat pe un if, am iesit din while.

```
if(t==0)
{
  x1 <- rnorm(1,minute,secunde)
  nr <- 0
}
else if(t==1)
{
  y1 <- runif(1,0,1)
  x1 <- rpois(1,minute+y1)
  nr <- 0
}
else if(t==5)
{
  x1 <- rexp(1,ore)
  nr <- 0
}
else if(t==7)
{
  y1 <- runif(1,0,5)
  x1 <- rbinom(1,ore,1/minute+y1)
  nr <- 0
}
...
}
```

```

else if(t==8)
{
  x1 <- runif(1,5,12)
  nr <- 0
}
else if(t==9)
{
  #nn=nr observatii, n=nr bile negre, m=nr bile albe,k=nr bile extrase
  y1 <- rhyper(1,21,15,10)
  #n=nr observatii,shape>=0,rate,scale>0(1/rate)
  x1 <- rgamma(1,2,1/5)
  x1 <- x1-y1
  nr <- 0
}
#daca nu a intrat pe niciunul, scadem nr
else
{
  nr <- nr-1
}
}

```

Cand iesim din while, verificam daca a parcurs de 3 ori ciclul și nu a obtinut niciun rezultat, ii atribuim valoarea specificata în enunt:

```

#daca s-a executat de 3 ori
if(is.null(x1))
{
  x1 <- rnorm(1,0,1)
}
return(x1)

```

Pentru urmatoarele valori din sir, construim functia gensir în care obtinem primul element cu ajutorul functiei create anterior și pe celelalte recursiv. Punem elementele într-un vector cu ajutorul functiei append.

```

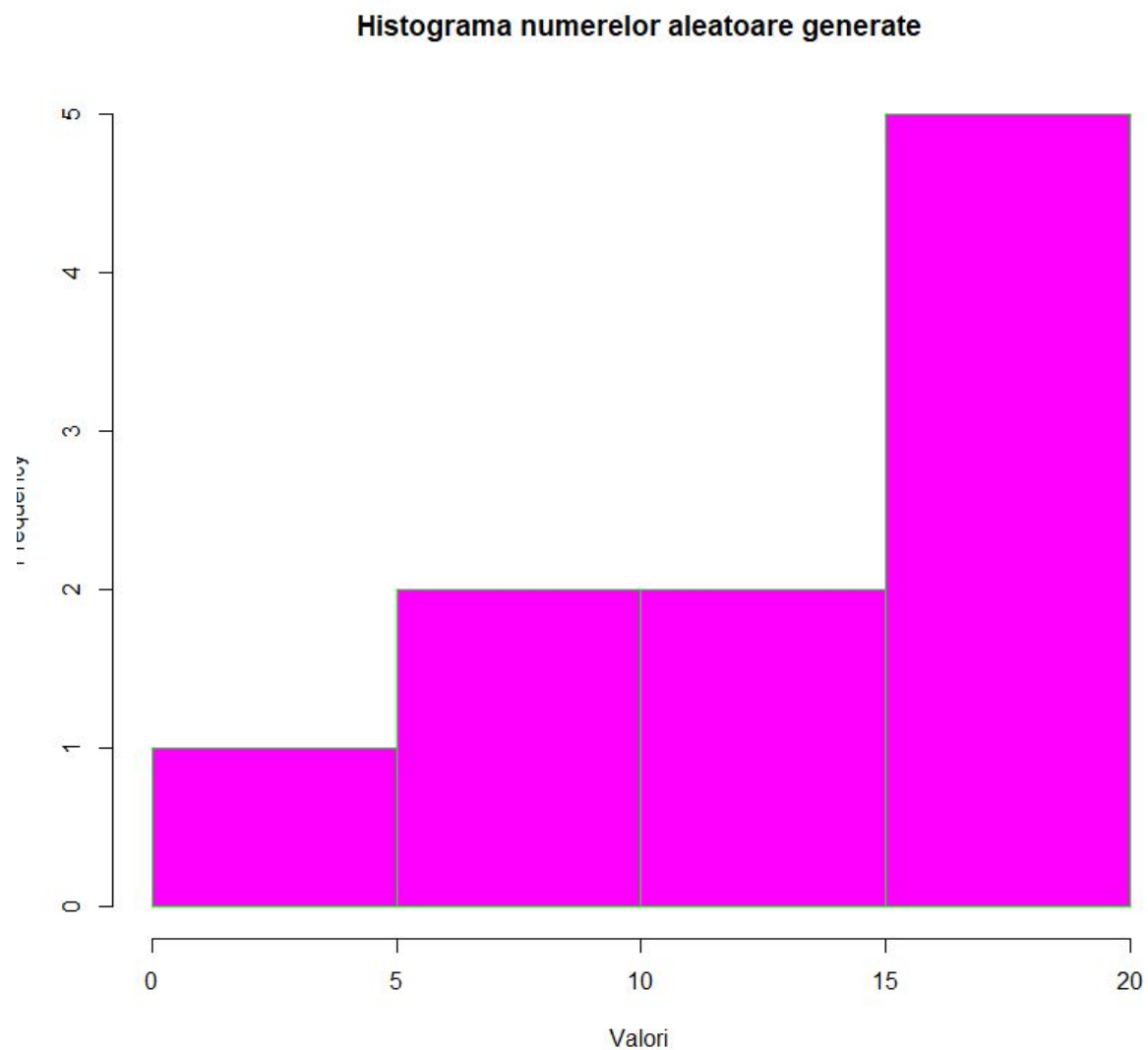
#2)
#xn=a*xn-1+b
#generam sirul
gensir <- function(n)
{
  x1 <- genx1()
  sir <- c()
  a <- rexp(1,2)
  b <- rnorm(1,5,1)
  sir <- append(sir,x1)
  n <- n-1
  while(n)
  {
    x <- a*x1+b
    sir <- append(sir,x)
    x1 <- x
    n <- n-1
  }
  return(sir)
}

```

În final, implementăm o funcție generală care apelează funcția creată anterior și realizează histograma valorilor din vector:

```
generator <- function(n)
{
  sir <- gensir(n)
  hist(sir,main="Histograma numerelor aleatoare generate",col="magenta",border="green",xlab="valori",breaks=3)
  print(sir)
}
generator(10)
```

Un exemplu ar fi:



Pentru exercitiul 4), am primit setul de date OrchardSprays și am efectuat operatii de statistica descriptiva, astfel:

```
#EX4
#OrchardSprays

set <- OrchardSprays

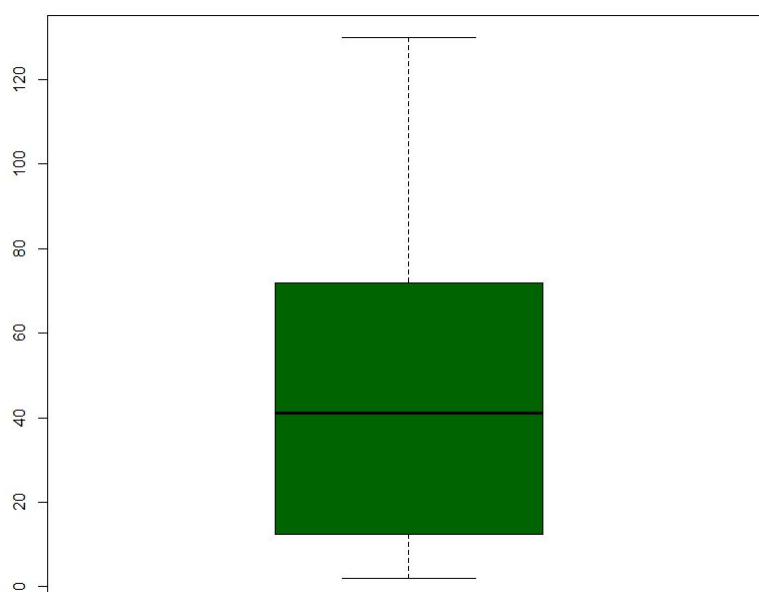
#media
mean(set$decrease)

#varianta
var(set$decrease)

#quartile
q1 <- quantile(set$decrease,1/4)
q2 <- quantile(set$decrease,2/4)
q3 <- quantile(set$decrease,3/4)
```

Pentru boxplot, am implementat un grafic general și unul pentru fiecare categorie:

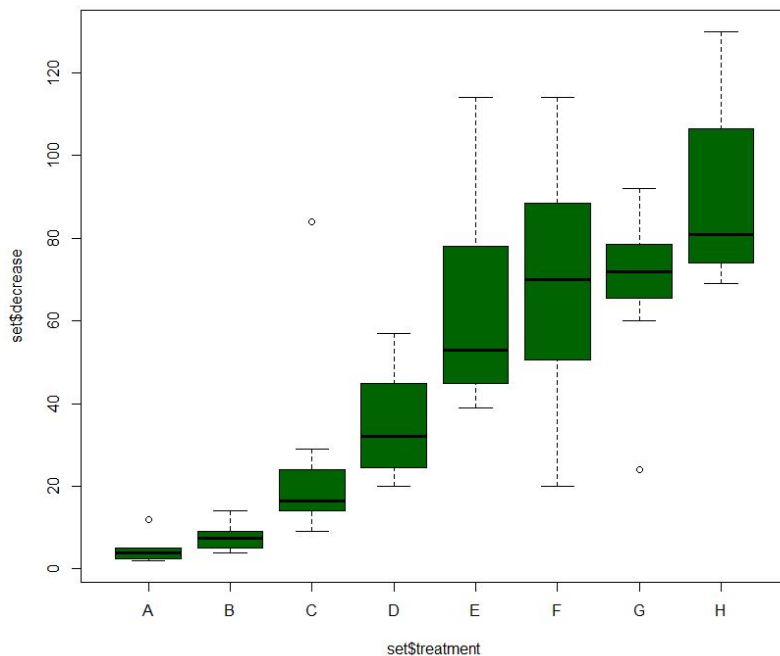
```
#boxplot
boxplot(c(set[1]),col="darkgreen")
boxplot(set$decrease~set$treatment ,col="darkgreen")
```



```
#boxplot
```

```
boxplot(c(set[1]),col="darkgreen")
```

```
boxplot(set$decrease~set$treatment ,col="darkgreen")
```



În final, am facut urmatoarele observatii dupa interpretarea graficelor:

```
#interpretari
```

```
hist(set$decrease)
```

```
#Pentru tratamentul A, fiind foarte putine experimente realizate,  
#valorile obtinute in decrease sunt insuficiente pentru a obtine un rezultat conclusiv.  
#Asadar, putem neglija aceasta categorie.
```

```
#Pe de alta parte, la o prima analiza a graficului, observam,  
#comparand categoriile de tratament E si F, ca media de descrestere are un  
#rezultat mai bun in categoria F fata de categoria E, insa limitele inferioare ale acestora  
#contrazic acest fapt.
```

```
#Observam prezenta a trei outlieri, ceea ce inseamna ca experimentele au avut o rata de succes ridicata.
```

```
#Observam ca mediile corespunzatoare rezultatelor sunt in ordine crescatoare de la A la H.
```

I

## Concluzii

Am intampinat dificultati la realizarea sumei și diferenței celor doua variabile primite prin densitatile de probabilitate de la exercitiul 2, pe aceasta din urma nereusind sa o implementam.

Proiectul a fost realizabil, am recapitulat cunostiintele acumulate pe parcursul semestrului, dar am și descoperit notiuni noi pe care nu le-am intalnit inainte.

Ne-am inteles foarte bine ca și echipa și consideram ca și acest fapt a dus la realizarea unui proiect reusit, din punctul nostru de vedere.