Laborator 3 - Simularea variabilelor aleatoare.
(Illustrări ale LNM şi TLC. Metodele Monte Carlo.)

I. Distribuţii continue remarcabile

RStudio. Nu uitaţi să alegeţi directorul de lucru: **Session** → **Set Working Directory** → **Choose Directory**.

Exerciţiu rezolvat. Reprezentaţi grafic funcţia de densitate a distribuţiei exponenţiale, $Exp(\lambda)$ ($\lambda > 0$).

Această distribuţie se anulează pe toată semiaxa negativă, deci este sufficient sa o reprezentăm pe semiaxa pozitivă (vom folosi, de fapt, un interval de forma $[0, a]$).

```
density_exponential = function(lambda, n, a) {
   x = seq(0, a, n);
   y = dexp(x, lambda);
   plot(x, y, type = 'l');
}
```

Exerciţiu propus.

I.1. Scrieţi o funcţie care să reprezinte grafic densităţile următoarelor distribuţii:

(a) $N(\mu, \sigma^2)$.

(b) $Student(r)$.

(c) $Gamma(\alpha, \lambda)$.

II. Legea numerelor mari (LNM).

Fie X_i, $1 \leqslant i \leqslant n$, un şir de variabile aleatoare independente şi identic distribuite, media lor aritmetică este

$$\overline{x}_n = \frac{X_1 + X_2 + \ldots + X_n}{n},$$

Atunci, conform LNM, $\overline{x}_n \to \mu$, unde $\mu = \mathbb{E}[X_i]$, $\forall 1 \leqslant i \leqslant n$.

Exerciţiu rezolvat. Verificaţi LNM utilizând şirul de variabile $X_i : Poisson(\lambda)$, $1 \leqslant i \leqslant n$. (Se ştie că $\mathbb{E}[X_i] = \lambda$.)

```
LLN_Poisson = function(lambda, n) {
   sum = 0;
   for(i in 1:n) {
      u = rpois(1, lambda);
      sum = sum + u;
   }
   return(sum/n);
}
```

O variantă mai simplă şi mai rapidă:

```
LLN_Poisson = function(lambda, n) {
   return(mean(rpois(n, lambda)));
}
```

Exerciţiu rezolvat. Verificaţi LNM utilizând şirul de variabile $X_i : Gamma(\alpha, \lambda)$, $1 \leqslant i \leqslant n$. (Se ştie că $\mathbb{E}[X_i] = \alpha/\lambda$.)

Folosim aici doar varianta mai simplă:

```
LLN_Gamma = function(alfa, lambda, n) {
  return(mean(rgamma(n, alfa, lambda)));
}
```

Exerciţii propuse.

II.1. Scrieţi câte o funcţie care să verifice LNM pentru următoarele şiruri de variabile aleatoare

(a) $X_i : Exponential(\lambda)$, $1 \leqslant i \leqslant n$. (Ştim că $\mathbb{E}[X_i] = 1/\lambda$.)

(b) $X_i : B(m, p)$, $1 \leqslant i \leqslant n$. (Ştim că $\mathbb{E}[X_i] = mp$.)

II.2 Rezolvaţi acelaşi exerciţiu pentru şirul de variabile aleatoare $X_i : Student(r)$, $1 \leqslant i \leqslant n$, pentru care se ştie că $\mathbb{E}[X_i] = 0$. Comparaţi rezultatele cu valorile exacte pentru următorii parametri: $n \in \{1000, 10000, 100000, 1000000\}$ and $r \in \{2, 3, 4, 5\}$.

III. Teorema Limită Centrală (TLC).

Fie X_i, $1 \leqslant i \leqslant n$, un şir de variabile aleatoare independente şi identic distribuite şi fie

$$\overline{x}_n = \frac{X_1 + X_2 + \ldots + X_n}{n}$$

media lor aritmetică, atunci, conform TLC, $\overline{x}_n$ (pentru valori foarte mari ale lui n) urmează o distribuţie $N(\mu, \sigma^2)$.

După standardizarea mediei de selecţie obţinem că $\dfrac{\overline{x}_n - \mu}{\sigma/\sqrt{n}} : N(0, 1)$.

Exerciţiu rezolvat. Verificaţi TLC folosind şirul de variabile aleatoare $X_i : Poisson(\lambda)$, $1 \leqslant I \leqslant n$. (Se ştie că $\mathbb{E}[X_i] = \lambda$ şi $Var[X_i] = \lambda$.)

TLC spune că

$$P\left(\frac{\overline{x}_n - \mu}{\sigma/\sqrt{n}} \leqslant z\right) \cong P(Z \leqslant z),$$

Unde $z \in \mathbb{R}$ şi $Z : N(0, 1)$ - legea normală standard. Probabilitatea din dreapta este $pnorm(z)$, în timp ce probabilitatea din stânga poate fi aproximată prin astfel: fie N (un număr foarte mare) de eşantioane aleatoare simple, fiecare de dimensiune n, $\left(X_i^k\right)_{i=\overline{1,n}}^{k=\overline{1,N}}$, apoi calculăm

$$P^N(z) = \frac{|\{k \,:\, \overline{x}_n^k \leqslant z\sigma/\sqrt{n} + \mu\}|}{N}.$$

(X_i^k sunt valori simulate conform distribuţiei date - Poisson în cazul exerciţiului de faţă). Se compară apoi această probabilitate cu $pnorm(z)$.

```
CLT_Poisson = function(lambda, n, N, z) {
  expectation = lambda;
  st_dev = sqrt(lambda);
  upper_bound = z * st_dev/sqrt(n) + expectation;
  sum = 0;
  for(i in 1:N) {
    x_n = mean(rpois(n, lambda));
    if(x_n ⩽ upper_bound) {
      sum = sum + 1;
    }
  }
  return(sum/N);
}
```

Remarcă: n ar trebui să fie cel puţin 30; e. g., $n = 30$, $N = 10000$, $z \in \{0, 1, 1.5, 2\}$.

Exerciţii propuse.

III.1. Scrieţi o funcţie care să verifice TLC folosind şirul de variabile aleatoare $X_i : Exponential(\lambda)$, $1 \leqslant i \leqslant n$. (Ştim că $\mathbb{E}[X_i] = 1/\lambda, Var[X_i] = 1/\lambda^2$.)

III.2 Rezolvaţi acelaşi exerciţiu pentru şirul de variabile aleatoare $X_i : Gamma(\alpha, \lambda)$, $1 \leqslant i \leqslant n$. (We know that $\mathbb{E}[X_i] = \alpha/\lambda$ and $Var[X_i] = \alpha/\lambda^2$.). Folosiţi $n = 50$, $N \in \{5000, 10000, 20000\}$ şi $z \in \{0.5, 1, 1.5, 2, 2.5\}$.

IV. Estimarea ariilor şi a volumelor

RStudio. Nu uitaţi să va setaţi directorul de lucru: **Session** → **Set Working Directory** → **Choose Directory**.

Exerciţiu rezolvat. Aria discului unitate este π. Acoperim discul cu un pătrat de dimensiuni 2 pe 2 şi estimăm numărul π folosind 10000, 50000 şi 100000 valori uniforme aleatoare. Comparăm apoi rezultatele cu valoarea cunoscută a lui $\pi = 3.14159265358...$

Discul unitate este inlcus în $[-1, 1] \times [-1, 1]$. Următoarea funcţie estimează π utilizând N numere aleatoare.

```
disc_area = function(N) {
  N_C = 0;
  for(i in 1:N) {
    x = runif(1, -1, 1);
    y = runif(1, -1, 1);
    if(x*x + y*y <= 1)
      N_C = N_C + 1;
    }
  return(4*N_C/N); }
}
```

Dacă am estimat o valoare α_{actual} prin metoda Monte Carlo şi obţinem α_{MC}, putem măsura eroarea făcută (aceea de a folosi α_{MC} în loc de α_{actual}) în cel puţin două moduri:

- **Eroarea absolută**: $\epsilon_{abs} = |\alpha_{MC} - \alpha_{actual}|$.
- **Eroarea relativă**: $\epsilon_{rel} = \frac{|\alpha_{MC} - \alpha_{actual}|}{|\alpha_{actual}|}$. Această valoare poate fi scrisă şi procentual, obţinând **eroarea procentuală**: $\epsilon_{per} = \epsilon_{rel} \cdot 100\%$.

Exerciţii propuse.

IV.1. Estimaţi volumul sferei unitate (care este $4\pi/3$) folosind eşantioane de numere aleatoare de dimensuni diferite şi apoi calculaţi erorile (absolută şi relativă) corespunzătoare.

IV.2. Estimaţi aria dintre parabola de ecuaţie $y = y = -2x^2 + 5x - 2$ şi axa Ox (abscisă) - folosind 10000 valori uniforme. Determinaţi aria exactă prin integrare şi calculaţi eroarea relativă.

Indicaţie: parabola intersectează axa Ox în punctele $(1/2, 0)$ şi $(2, 0)$ şi are vârful în $(5/4, 9/8)$. Un domeniu rectangular din planul real care acoperă această arie poate fi $[0, 2] \times [0, 2]$.

V. Integrarea Monte Carlo

Exerciţiu rezolvat. Estimaţi valoarea următoarei integrale folosind 20000 şi apoi 50000 de valori aleatoare (determinaţi 30 astfel de aproximări pentru fiecare eşantion şi calculaţi câte o medie şi câte o deviaţie standard).

$$\int_0^{10} e^{-u^2/2}\, du.$$

Următoarea funcţie oferăo estimare pentru un eşantion de dimensiune N

```
MC_integration = function(N) {
  sum = 0;
  for(i in 1:N) {
    u = runif(1, 0, 10);
    sum = sum + exp(-u*u/2;
  }
  return(10*sum/N);
}
```

Putem calcula o medie pentru $k = 30$ astfel de aproximări şi şi deviaţia standard corespunzătoare folosind următoarea funcţie.

```
MC_integr_average= function(k, N) {
  for(i in 1:k)
    estimates[i] = MC_integration(N);
  print(mean(estimates));
  print(sd(estimates));
}
```

În urma execuţiei acestei funcţii obţinem

```
> MC_integr_average(30, 20000
[1] 1.249768
[1] 0.02327472
> MC_integr_average(30, 50000
[1] 1.253072
[1] 0.01373724
```

Exerciţiu rezolvat. Estimaţi valoarea următoarei integrale folosind 20000 şi apoi 50000 de valori aleatoare (determinaţi 30 astfel de aproximări pentru fiecare eşantion şi calculaţi câte o medie şi câte o deviaţie standard), utilizând metoda MC îmbunătăţită, anume cu distribuţia exponenţială ($\lambda = 1$)

$$\int_0^{+\infty} e^{-u^2}\, du.$$

(Valoarea exactă acestei integrale este $\sqrt{\pi}/2 \approx= 0.8862269$.)

Mai întâi, următoarea funcţie oferă o estimare pentru un eşantion de dimensiune N

```
MC_improved_integration = function(N) {
   sum = 0;
   for(i in 1:N) {
      u = rexp(1, 1);
      sum = sum + exp(-u*u)/exp(-u);
   }
   return(sum/N);
}
```

Putem calcula o medie pentru $k = 30$ astfel de aproximări şi şi deviaţia standard corespunzătoare folosind următoarea funcţie.

```
MC_imprvd_integr_average= function(k, N) {
estimates = 0;
   for(i in 1:k)
      estimates[i] = MC_improved_integration(N);
   print(mean(estimates));
   print(sd(estimates));
}
```

În urma execuţiei acestei funcţii obţinem

```
> MC_imprvd_integr_average(30, 20000)
[1] 0.8858024
[1] 0.002743676
> MC_imprvd_integr_average(30, 50000)
[1] 0.8861285
[1] 0.00213069
```

Exerciţii propuse.

V.1. ((a) sau (b) şi (c) sau (d)) Estimaţi valorile următoarelor integrale (comparând estimarea cu valoarea exactă) şi calculaţi erorile absolute şi relative corespunzătoare:

$$(a) \int_0^{\pi} \sin^2 x\, dx = \frac{\pi}{2};\ (b) \int_1^4 e^x\, dx = 51.87987;$$

$$(c) \int_0^1 \frac{dx}{\sqrt{1-x^2}} = \frac{\pi}{2};\ (d) \int_1^{+\infty} \frac{dx}{4x^2-1} = \ln 3/4.$$

V.2 Estimaţi valoarea următoarei integrale utilizând metoda MC îmbunătăţită, cu distribuţia exponenţială ($\lambda = 3, N = 50000$)

$$\int_0^{+\infty} e^{-2u^2}\, du = \sqrt{\pi/8}.$$

Comparaţi rezultatul cu valoarea exactă, şi calculaţi erorile absolute şi relative corespunzătoare. Determinaţi apoi 30 astfel de aproximări şi calculaţi o medie şi o deviaţie standard. (Vezi exerciţiul rezolvat.)

VI. Estimarea mediilor

Exerciţiu rezolvat. Modelul stochastic pentru numărul de erori (bug-uri) găsite într-un nou produs software se poate descrie după cum urmează. Zilnic cei care testează produsul determină un număr aleator de erori care sunt apoi corectate. Numărul de erori găsite în ziua i urmează o distribuţie Poisson(λ_i) al cărei parametru este cel mai mic număr de erori din cele două zile anterioare:

$$\lambda_i = \min\{X_{i-2}, X_{i-1}\}.$$

Care este numărul mediu de zile în care sunt detectate toate erorile? (Presupunem că în primele două zile sunt găsite 31 şi 27 erori, respectiv.) Folosiţi $N = 10000$ de simulări (”runs”) pentru estimatorul Monte Carlo.

Generăm un număr de erori pentru fiecare zi, până când acest număr este 0. Următoarea funcţie oferă numărul de zile până când nu mai apar erori (pentru o singură simulare - ”run”).

```
Nr_days = function() {
   nr_days = 1;
   last_errors = c(27, 31);
   nr_errors = 27;
   while(nr_errors > 0) {
      lambda = min(last_errors);
      nr_errors = rpois(1, lambda);
      last_errors = c(nr_errors, last_errors[1]) ;
      nr_days = nr_days + 1;
   }
   return(nr_days);
}
```

Executăm această funcţie de $N = 10000$ de ori şi returnăm media obţinută

```
MC_nr_days = function(N) {
   s = 0;
   for(i in 1:N)
      s = s + Nr_days();
   return(s/N);
}
```

Rezultatul este 28.0686, astfel, în aproximativ 4 săptămâni toate erorile vor fi găsite.

Exerciţii propuse.

VI.1 Rezolvaţi din nou exerciţiul anterior considerând că λ_i este media numărului de erori din cele trei zile anterioare (În primele trei zile numărul de erori găsite a fost de 9, 15 şi 13, respectiv).