

Laborator 3 - Simulare. Metode de tip Monte Carlo.

I. Estimarea ariilor și a volumelor

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

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

Discul unitate este inclus în $[-1, 1] \times [-1, 1]$. Următoarea funcție 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 și obținem α_{MC} , putem măsura eroarea făcută (aceea de a folosi α_{MC} în loc de α_{actual}) în cel puțin două moduri:

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

Exerciții propuse.

- I.1. Estimați volumul sferei unitate (care este $4\pi/3$) folosind eșantioane de numere aleatoare de dimensiuni diferite și apoi calculați erorile (absolută și relativă) corespunzătoare.
- I.2. Estimați aria dintre parabola de ecuație $y = -2x^2 + 5x - 2$ și axa Ox (abscisă) - folosind 10000 valori uniforme. Determinați aria exactă prin integrare și calculați eroarea relativă.

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

II. Integrarea Monte Carlo

Exercițiu rezolvat. Estimați valoarea următoarei integrale folosind 20000 și apoi 50000 de valori aleatoare uniforme (determinați 30 astfel de aproximări pentru fiecare din cele două dimensiuni și calculați câte o medie și câte o deviație standard).

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

Următoarea funcție oferă o estimare pentru un eșantion 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 și și deviația standard corespunzătoare folosind următoarea funcție.

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

În urma execuției acestei funcții obținem

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

Exercițiu rezolvat. Estimați valoarea următoarei integrale folosind 20000 și apoi 50000 de valori aleatoare uniforme (determinați 30 astfel de aproximări pentru fiecare din cele două dimensiuni și calculați câte o medie și câte o deviație standard), utilizând metoda MC îmbunătățită, anume cu distribuția exponențială ($\lambda = 1$)

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

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

Mai întâi, următoarea funcție oferă o estimare pentru un eșantion 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 și și deviația standard corespunzătoare folosind următoarea funcție.

```
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 execuției acestei funcții obținem

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

Exerciții propuse.

- II.1. ((a) sau (b) și (c) sau (d)) Estimați valorile următoarelor integrale (comparând estimarea cu valoarea exactă) și calculați erorile absolute și relative corespunzătoare:

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

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

- II.2 Estimați valoarea următoarei integrale utilizând metoda MC îmbunătățită, cu distribuția exponențială ($\lambda = 3, N = 50000$)

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

Comparați rezultatul cu valoarea exactă, și calculați erorile absolute și relative corespunzătoare. Determinați apoi 30 astfel de aproximări și calculați o medie și o deviație standard. (Vezi exercițiul rezolvat.)

III. Estimarea mediilor

Exercițiu 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 software testers determină un număr aleator de erori care sunt apoi corectate. Numărul de erori găsite în ziua i urmează o distribuție 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 și 27 erori, respectiv.) Folosiți $N = 10000$ de simulări ("runs") pentru estimatorul Monte Carlo.

Generăm un număr de erori pentru fiecare zi, până când acest număr este 0. Următoarea funcție 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 funcție de $N = 10000$ de ori și returnăm media obținută

```

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.

Exerciții propuse.

- III.1 Rezolvați din nou exercițiul 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 și 13, respectiv).
- III.2 Doi mecanici schimbă filtrele de ulei pentru autoturisme într-un service. Timpul de servire este exponențial cu parametrul $\lambda = 4 \text{ hrs}^{-1}$ în cazul primului mecanic și $\lambda = 12 \text{ hrs}^{-1}$ în cazul celui de al doilea. Deoarece al doilea mecanic este mai rapid, el servește de 3 ori mai mulți clienți decât partenerul său. Astfel când un client ajunge la rând, probabilitatea de a fi servit de primul mecanic este $3/4$. Fie X timpul de servire pentru un mecanic. Generați valori pentru X și estimați media lui X .

IV. Estimarea probabilităților

Exercițiu rezolvat. Modelul stochastic pentru numărul de erori (bug-uri) ăsite într-un nou produs software se poate descrie după cum urmează. Zilnic cei care testează produsul software testers determină un număr aleator de erori care sunt apoi corectate. Numărul de erori găsite în ziua i urmează o distribuție $\text{Poisson}(\lambda_i)$ al cărei parametru este cel mai mic număr de erori din cele trei zile anterioare:

$$\lambda_i = \min \{X_{i-3}, X_{i-2}, X_{i-1}\}.$$

- (a) Estimați probabilitatea de a mai avea încă erori după 21 de zile de teste folosind 500 de simulări MC. (În primele trei zile numărul de erori găsite este 28, 22 și 18, respectiv.).
- (b) Estimați această probabilitate, cu o eroare de cel mult ± 0.01 cu probabilitate 0.95.

(a) Utilizăm $N = 5000$ simulări ("runs") Monte Carlo; în fiecare simulare generăm un număr de erori găsite în fiecare zi până când acest număr devine 0. Următoarea funcție returnează numărul de zile până când nu mai există erori (la o singură simulare).

```

Nr_days = function() {
  nr_days = 2;
  last_errors = c(18, 22, 28);
  nr_errors = 18;
  while(nr_errors > 0) {
    lambda = min(last_errors);
    nr_errors = rpois(1, lambda);
    last_errors = c(nr_errors, last_errors[1:2]) ;
    nr_days = nr_days + 1;
  }
  return(nr_days);
}

```

Aplicăm această funcție de $N = 5000$ ori și returnăm proporția valorilor care depășesc (strict) 21 de zile.

```

MC_nr_days_21 = function(N) {
  s = 0;
  for(i in 1:N) {
    if(Nr_days() > 21) ;
      s = s + 1;
  }
  return(s/N);
}

```

Proporția calculată, 0.246, este o estimare a probabilității de a mai avea erori nedetectate și după 21 de zile de testare.

(b) Vom estima probabilitatea în două moduri.

Mai întâi, folosim o valoare "prezumată", $p^* = 0.246$, și $N \geq p^*(1 - p^*) \left(\frac{z_{\frac{\alpha}{2}}}{\epsilon} \right)^2$:

```

> alfa = 1 - 0.95
> z = qnorm(alfa/2)
> epsilon = 0.01
> p = 0.246
> N_min = p(1 - p)*(z/epsilon)^2
> N_min
[1] 7125.291
> MC_nr_days_21(N_min + 1)
[1] 0.2547264

```

Obținem $N \geq 7125.291$ și putem aplica $MC_nr_days(N_min + 1)$

A doua metodă utilizează minorantul $N \geq \left(\frac{z_{\frac{\alpha}{2}}}{2\epsilon} \right)^2$:

```

> alfa = 1 - 0.95
> z = qnorm(alfa/2)
> epsilon = 0.01
> p = 0.246
> N_min = (1/4)*(z/epsilon)^2
> N_min
[1] 9603.647
> MC_nr_days(N_min + 1)
[1] 0.2496968

```

Obținem $N \geq 9603.647$ și putem aplica $\text{MC_nr_days}(N_{\min} + 1)$. De obicei, cu a cea de-a doua metodă numărul de simulări ("runs") este mai mare.

Exerciții propuse.

- IV.1 Estimați probabilitatea $P(X < Y^2)$, unde X și Y sunt variabile repartizate geometric, independente, cu parametrii 0.3 și 0.5, respectiv. Apoi estimați aceeași probabilitate cu o eroare care să nu depășească ± 0.005 cu probabilitate 0.95. Care ar trebui să fie numărul de simulări ("runs")?
- IV.2 Patruzeci de computere sunt conectate printr-un LAN. Unul dintre ele este infectat de un anumit virus. În fiecare zi, acest virus ajunge de la un computer infectat la unul "curat" cu probabilitate 0.2. De asemenea, în fiecare zi (începând din a doua zi), administratorul de sistem alege la întâmplare zece computere infectate (sau pe toate dacă sunt infectate mai puțin de 10) și îndepărtează virusul.
- (a) Estimați probabilitatea ca într-o anumită zi toate computerele să fie infectate.
 - (b) Estimați probabilitatea ca într-o anumită zi cel puțin 15 computere să fie infectate.
 - (c) Estimați această ultimă probabilitate cu o eroare de ± 0.01 cu probabilitatea 0.95.

Temă pentru acasă.

14 puncte [1p: C1] + [1p: C2] + [1p: C3] + [5p: C4 sau C5] + [6p: C6 sau C7]

C1. (1 punct) Volumul unui vas paraboloidal (de revoluție)

$$P(a) = \{(u, v, w) : w = u^2 + v^2, w \leq a\} \subseteq [-\sqrt{a}, \sqrt{a}] \times [-\sqrt{a}, \sqrt{a}] \times [-a, a]$$

este $\frac{\pi a^2}{2}$. Estimați acest volum utilizând metoda Monte Carlo pentru $a = 2$ și comparați rezultatul cu valoarea exactă. Folosiți eșantioane de dimensiune 10000, 20000 și 50000 și calculați erorile relative.

C2. (1 punct) Fie T un triunghi ale cărui laturi sunt incluse în următoarele trei drepte: $x - 2y = 0$, $2x - 3y = 0$ și $3x + 2y = 6$, respectiv. Estimați aria (necunoscută) acoperită de T folosind metoda Monte Carlo 10000 de simulări.

Indicație: punctele interioare triunghiului sunt caracterizate de următoarele inegalități: $x - 2y \leq 0$, $2x - 3y \geq 0$, și $3x + 2y \leq 6$. Este necesar să găsiți o zonă rectangulară care să acopere triunghiul în întregime.

C3. (1 punct) Estimați valorile următoarelor integrale și comparați rezultatul cu valorile exacte (dacă sunt date):

$$(a) \int_0^{\pi/3} (\sin^3 x + \cos^3 x) dx = \frac{3\sqrt{3}}{8} + \frac{13}{24}, (b) \int_0^{+\infty} \frac{dx}{2x^2 + 1} = \frac{\pi}{2\sqrt{2}}.$$

C4. (5 puncte) Două servere web oferă (serves) aceleași pagini posibililor clienți (web). Timpul necesar procesării unei cereri (request) HTTP este distribuit $\Gamma(6, 4)$ pe primul server și $\Gamma(5, 3)$ pe cel de-al doilea (în miliseconde). La această durată se adaugă latența dintre client și servere pe Internet care are o distribuție exponențială cu $\lambda = 1$ (în miliseconde⁻¹). Se știe că un client este direcționat către primul server cu probabilitatea 0.35 și către al doilea server cu probabilitatea 0.65. Estimați timpul mediu necesar servirii unui client (de la lansarea cererii până la primirea răspunsului).

C5. (5 puncte) Două zeci și cinci de computere sunt conectate într-un LAN. Un virus infectează această rețea în felul următor: în fiecare zi, acest virus ajunge de la un computer

infectat la unul "curat" cu probabilitate 0.25. De asemeni, în fiecare zi (începând cu a doua zi), administratorul de sistem îndepărtează virusul cu probabilitate 0.15 de pe unul două (dacă există) dintre calculatoarele infectate alese la întâmplare și independent.

- (a) Estimați numărul mediu de zile necesare îndepărtării virusului din întreaga rețea.
- (b) Estimați probabilitatea ca într-o anumită zi cel puțin cinci computere să fie infectate.

C6. (5 puncte) O pădure formată din 1600 de copaci are forma unui dreptunghi de dimensiuni 40×40 . În colțul sud-estic (dreapta-jos) ia naștere un incendiu. Vântul bate dinspre est și, cu probabilitate 0.7, orice copac ia foc de la vecinul să estic (dacă acesta deja arde). Probabilitățile ca un copac să ia foc de la vecinul din vest sau sud (dacă aceștia ard) sunt egale cu 0.4, iar probabilitatea ca un copac să ia foc de la vecinul din nord este 0.2. întreprindeți un studiu Monte Carlo pentru a estima probabilitatea ca cel puțin 25% din pădure să ardă la un moment dat.

C7. (5 puncte) Două zeci și cinci de studenți au conturi pe un același server de mail. Contul unul student este infectat de un malware. În fiecare zi, acest malware se răspândește și ajunge de la un cont infectat la unul neinfectat cu probabilitate 0.3. De asemeni, în fiecare zi (începând cu a doua zi), administratorul de sistem curăță la întâmplare 4 conturi (sau toate conturile dacă sunt infectate mai puțin de 4).

- (a) Estimați probabilitatea ca fiecare cont să fie infectat cel puțin o dată.
- (b) Estimați probabilitatea ca într-o anumită zi toate conturile să fie infectate.
- (c) Cât de mare trebuie să fie numărul de simulări ("runs") - adică dimensiunea eșantionului - astfel ca prima probabilitate să aibă o eroare de cel mult ± 0.01 cu probabilitatea 0.99? Găsiți o astfel de estimare.

Rezolvările acestor exerciții (funcțiile R și apelurile lor) vor fi redactate într-un script R.