Matematici speciale

(curs 8 - S8)

11.04.2024

	$X \sim N(m, \sigma)$	$Z \sim N(0,1)$
Fct. de prob.	$f(x)$	$\varphi(t) = \frac{1}{\sqrt{2\pi}} = \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{t^2}{2}}$
Fct. de repartiție	$F_X(x) = P(X \leq x)$	$F_X(x) = \phi(z)$
Media / Dispersia	$M(X) = m$ $\sigma^2(X) = \sigma^2$	$M(Z) = 0$, $\sigma^2(Z) = 1$ $\phi(-x) = 1 - \phi(x)$ $z_{1-\alpha} = -z_\alpha$, $\phi(z_\alpha) = \alpha$

1. T - v.a. ce înregistrează timpul dedicat unui proiect

a) $T \sim N(m = 12, \sigma^2 = 4)$

$P(8 \leq T \leq 16) = ?$

standardizare:

$$P(8 \leq T \leq 16) \underset{Z \sim N(0,1)}{\overset{Z = \frac{T-m}{\sigma}}{=\!=}} P\left(\frac{8-12}{2} \leq \frac{T-12}{2} \leq \frac{16-12}{2}\right) = P(-2 \leq Z \leq 2) =$$

$$= \underset{F_Z}{\phi}(2) - \phi(-2) = \phi(2) - (1 - \phi(2)) = 2\phi(2) - 1 = 0,95$$

b) $P(T > t_{min}) = 0,95 \overset{?}{\Rightarrow} t_{min} = ?$

$$P\left(\frac{T-m}{\sigma} > \frac{t_{min}-12}{2}\right) = 0,95 \Rightarrow P\left(Z > \frac{t_{min}-12}{2}\right) = 0,95$$

$$\Leftrightarrow 1 - P\left(Z \leq \frac{t_{min}-12}{2}\right) = 0,95$$

$$\Leftrightarrow -\phi\left(\frac{t_{min}-12}{2}\right) = 0,95 - 1$$

$$\phi\Big(\underbrace{\frac{t_{min}-12}{2}}_{z_{0,05}}\Big) = 0,05 \Rightarrow \frac{t_{min}-12}{2} = z_{0,05}$$

$$z_{0,95} = -z_{0,05} \Rightarrow z_{0,05} = -1,64 \Rightarrow$$

$\Rightarrow t = 2 \cdot (-1{,}64) + 12$

(2.) Punctaj - v.a ce inregistreaza punctajul obtinut la testul de admitere

$Punctaj \sim N(m = 500, \bar{\sigma} = 200)$

$P(Punctaj > x_{min}) = 0{,}2$

$\xRightarrow{\text{Standardizare}} P\left(\frac{Punctaj - 500}{200} > \frac{x_{min} - 500}{200}\right)$

$Z \sim N(0,1)$

$1 - \phi\left(\frac{x_{min} - 500}{200}\right) = 0{,}2$

$\Rightarrow \phi\left(\frac{x_{min} - 500}{200}\right) = 0{,}8$

$\frac{x_{min} - 500}{200} = 0{,}85 \Rightarrow x_{min} = 500 + 200 \cdot 0{,}85 = 670$

Simularea variabilelor aleatoare

A simula o variabilă aleatoare discretă X, presupune a genera independent, conform unui algoritm, un șir de numere $y_1, y_2, \ldots, y_N$, care să aibă particularitățile unor valori de observație asupra variabilei.

Mai precis: dacă variabila X are distribuția de probabilitate:

$$X = \begin{pmatrix} x_1 & x_2 & \ldots & x_n \\ p_1 & p_2 & \ldots & p_n \end{pmatrix}$$

atunci distribuția experimentală a datelor generate trebuie să fie foarte apropiată de distribuția teoretică a variabilei X:

dacă nr_k -numărul valorilor generate, egale cu x_k, $k \in \{1, n\}$, atunci $\frac{nr_k}{N}$ trebuie să fie apropiat de p_k

ex: aruncarea unei monede

$$X = \begin{pmatrix} 0 & 1 \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

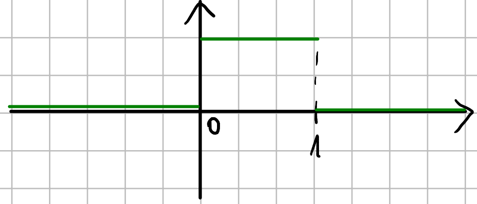
$N = 100 \longrightarrow$ generăm 100 de numere de 0 și 1

$\Rightarrow$ 50 de ori nr. 1

50 de ori nr. 0

$$Z = \begin{pmatrix} 0 & 1 \\ \frac{1}{3} & \frac{2}{3} \end{pmatrix}$$

$U \sim \text{Unif}[0,1] \Rightarrow \quad f_U(x) = \begin{cases} 1, & x \in [0,1] \\ 0, & \text{în rest} \end{cases}$



$$F_U(x) = \begin{cases} 0, & x < 0 \\ x, & x \in [0,1) \\ 1, & x \geq 1 \end{cases}$$

$P(c \leq U \leq d) = F_U(d) - F_U(c) = d - c \longrightarrow$ lungimea intervalului

$[c,d] \subset [0,1]$

Reamintim: dacă $U \sim$ Unif$[0,1)$, atunci $(a,b] \subset [0,1)$,

$$P(U \in (a,b]) = b - a$$

Exemplu: Dacă $A-$ eveniment $P(A) = 0.65$ și $B = \complement A$, $P(B) = 0.35$, atunci putem simula producerea unuia din cele două evenimente astfel:

- apelăm generatorul `urand()`:
  u=urand();
- Dacă $u < 0.65$, spunem că s-a produs evenimentul $(U < 0.65)$, $P(U < 0.65) = P(U \in [0, 0.65)) = 0.65$.
  A are aceeași probabilitate de realizare, deci putem considera că s-a produs evenimentul A.
- Dacă $u \in [0.65, 1)$, atunci s-a produs evenimentul $(U \in [0.65, 1))$, $P(U \in [0.65, 1)) = 1 - 0.65 = 0.35$.
  Deci, în acest caz putem considera că s-a produs evenimentul B.

1. Distributia uniforma discreta

Propoziție

Dacă $U \sim$Unif[0,1) v.a. uniform distribuită pe $[0,1)$, iar n este număr întreg, $n > 1$, atunci

$$X = [nU]$$

o variabilă aleatoare discretă ce are distribuția uniformă pe mulțimea

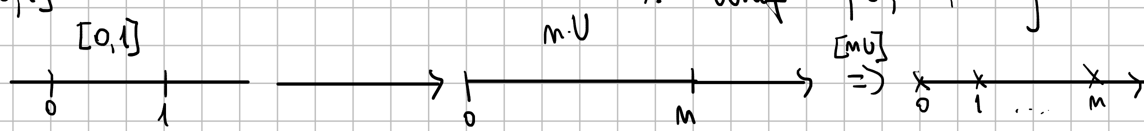
$B_X = \{0, 1, 2, \ldots, n-1\}$, adică $X = \begin{pmatrix} 0 & 1 & \ldots & n-1 \\ \frac{1}{n} & \frac{1}{n} & \ldots & \frac{1}{n} \end{pmatrix}$

$|B_X| = n$

$U \longrightarrow n \cdot U \longrightarrow X = [nU]$

$U \sim \text{Unif}[0,1] \qquad\qquad X \sim \text{Unif}\{0, n, n-1\}$

Demonstrație



- U ia valori în $[0,1)$, variabila nU ia valori în $[0,n)$ și $[nU] \in \{0, 1, \ldots, n-1\}$.
-
$$\begin{aligned} P(X = k) &= P([nU] = k) = P(k \leq nU < k+1) \\ &= P\left(\tfrac{k}{n} \leq U < \tfrac{k+1}{n}\right) = P\left(U \in \left[\tfrac{k}{n}, \tfrac{k+1}{n}\right)\right) = \tfrac{k+1}{n} - \tfrac{k}{n} = \tfrac{1}{n} \end{aligned}$$

```
Function SimDiscretU(n)
u=urand();
k=int(n*u);
return k;
end.
```

Observație: Inlocuind `return k;` cu return x_k; se returnează o valoare de observație asupra unei variabile aleatoare discrete, uniform distribuită pe o mulțime $\{x_0, x_1, \dots, x_{n-1}\}$, adică asupra variabilei aleatoare:

$$X = \begin{pmatrix} x_0 & x_1 & \dots & x_{n-1} \\ \frac{1}{n} & \frac{1}{n} & \dots & \frac{1}{n} \end{pmatrix}$$

Observație: Simularea acestei variabile aleatoare este echivalentă cu simularea experimentului de extragere la întamplare a unui element (obiect) din lista $x_0, x_1, \dots, x_{n-1}$ și returnarea lui în "recipientul" din care a fost scos. La întamplare înseamnă că fiecare element are aceeași șansă de a fi extras, adică probabilitatea $1/n$.

Generalizare

Algoritm ce extrage un număr la întâmplare din mulțimea de numere întregi $\{m, m+1, \dots, n\}$, $m < n$.

- Mulțimea conține $N = n - m + 1$ elemente.
- Un număr selectat la întamplare este o valoare de observație asupra variabilei aleatoare:

$$X = \begin{pmatrix} m & m+1 & \dots & n \\ \frac{1}{n-m+1} & \frac{1}{n-m+1} & \dots & \frac{1}{n-m+1} \end{pmatrix}.$$

```
Function randint(m,n)
u=urand();
k=int((n-m+1)*u);//k in {0,1,2,...,n-m}
return k+m;
end.
```

Aplicație: Generarea unei permutari aleatoare[2]

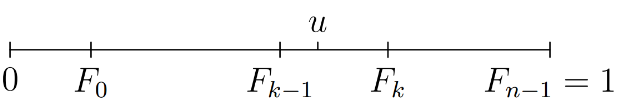
Simularea v.a. discrete cu o distribuție neuniformă

Fie X o v.a. discretă cu valorile ordonate, adică $x_0 < x_1 < \dots < x_{n-1}$.

$$X = \begin{pmatrix} x_0 & x_1 & \dots & x_{n-1} \\ p_0 & p_1 & \dots & p_{n-1} \end{pmatrix}, \quad \sum_{k=0}^{n-1} p_k = 1$$

- divizăm intervalul $[0, 1)$, prin punctele

$$F_0 = p_0, \quad F_1 = p_0 + p_1, \dots, F_k = p_0 + p_1 + \dots + p_k, \dots, F_{n-1} = 1$$



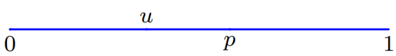
- Avem:

$$P(U \in (F_{k-1}, F_k]) = F_k - F_{k-1}{}_{lung.interv.} = p_k = P(X = x_k)$$

(am considerat $F_{-1} = 0$).

2. Simularea distribuției Bernoulli

Distribuția Bernoulli $X = \begin{pmatrix} 1 & 0 \\ p & 1-p \end{pmatrix}$ se simulează când avem de făcut o alegere dintre două alternative, codificate cu 1 și 0.



- se generează $u \in [0, 1)$, apelând $\mathtt{u} = \mathtt{urand}()$;
- Dacă $u \leq p$, atunci s-a produs evenimentul $(U \leq p)$, $P(U \leq p) = P(0 \leq U \leq p) = p - 0 = p$ (aceeași probab. ca $(X = 1)$). Putem presupune că s-a produs acesta și alegem 1.
- Dacă însă $u > p$, atunci facem alegerea codificată de 0.

```
Function Bernoulli(p);
u=urand();
if (u < p) return 1;
else return 0;
end
```

3. Simularea distribuției binomiale

Fie $X \sim Bin(n, p)$ cu $X = \begin{pmatrix} 0 & 1 & \dots & k & \dots & n \\ Pr_0 & Pr_1 & \dots & Pr_k & \dots & Pr_n \end{pmatrix}$

Obs.: se simulează conform algoritmului prezentat mai sus, singura problemă fiind calculul probabilităților $Pr_k = P(X = k) = C_n^k p^k (1-p)^{n-k}$ și a sumelor $F_k = Pr_0 + Pr_1 + \dots + Pr_k$, $k \in \{0, n\}$.

Folosind $C_n^{k+1} = C_n^k \frac{n-k}{k+1}$, avem

$$Pr_{k+1} = P(X = k+1) = C_n^{k+1} p^{k+1} (1-p)^{n-k-1} = \frac{n-k}{k+1} \frac{p}{1-p} P(X = k)$$

Notând cu $c = \frac{p}{1-p}$, obținem

$$Pr_{k+1} = \frac{n-k}{k+1} c\, Pr_k, \quad Pr_0 = (1-p)^n$$

4. Simularea distribuției geometrice

Fie X o variabilă aleatoare ce are distribuția geometrică de parametru $p \in (0, 1)$, $X \sim \mathrm{Geom}(p)$, $X = \begin{pmatrix} k \\ p(1-p)^{k-1} \end{pmatrix}$, $\quad k \in \mathbb{N} \setminus \{0\}$

Metoda directă de simulare= variabilă geometrică este asociată unui experiment Bernoulli:

```
FunctionGeom1(p)
k=0; // k contorul pentru incercarile Bernoulli
do {
    u=urand()
    k=k+1;
    } while(u > p);
return k
end
```

Obs.:

- Se execută blocul de instrucțiuni din bucla `do-while` atâta timp cât încercările sunt un eșec.
- La primul succes returnează numărul încercării respective.
- Dacă probabilitatea succesului p este mică numărul mediu de încercări până la primul succes este mare pentru că $M(X) = 1/p$ și în acest caz algoritmul `Geom1` este ineficient.

Propoziție

Dacă $p \in (0,1)$ și $U \sim \text{Unif}[0,1)$, atunci variabila aleatoare:

$$X = \left[\frac{\ln(1-U)}{\ln(1-p)}\right] + 1$$

are distribuția geometrică de parametru p.

Deci, pentru variabila aleatoare $X \sim \text{Geom}(p)$, evenimentul $(X = k)$ are aceeași probabilitate ca evenimentul $\left(\left[\frac{\ln(1-U)}{\ln(1-p)}\right] + 1 = k\right)$,

```
FunctionGeom2(p)
u=urand();
return int(log(1 − u)/log(1 − p)) + 1;
end.
```

Demonstratie:

$$P(X=k) = P\left[\left(\frac{\ln(1-U)}{\ln(1-p)}\right) + 1 = k\right] = P\left[\frac{\ln(1-U)}{\ln(1-p)} = k-1\right] =$$

$$= P\left[k-1 \leq \frac{\ln(1-U)}{\ln(1-p)} \leq (k-1)+1\right]$$

$\ln(1-p) < 0 = \ln 1$

$1-p < 1$

$p > 0$

$$P\left[k \ln(1-p) < \ln(1-U) \leq (k-1)\ln(1-p)\right] =$$

$$= P\left[(1-p)^k \leq 1-U < (1-p)^{k-1}\right] =$$

$$= P\left[1-(1-p)^{k-1} < U \leq 1-(1-p)^k\right] =$$

$$= 1-(1-p)^k - 1 + (1-p)^{k-1} =$$

$$= p(1-p)^{k-1}$$

Simularea v.a. continue

X variabila aleatoare cu o anumita distribuție de probabilitate =: impunem condiția ca F_X - strict crescătoare pe I => $\exists F_X^{-1}$

Se poate demonstra că $Y = F_X^{-1}(U)$ și X sunt identice din punct de vedere probabilistic.

$$U \xrightarrow{F_X^{-1}} Y = F_X^{-1}(U) \overset{\text{asem.}}{\underset{\substack{\text{d.p.d.v.}\\ \text{probabilistic}}}{\equiv}} X$$

Dem.

Vom deduce fct. de repartiție a lui Y

(se poate nota cu G_Y această funcție)

$$G_Y(x) : P(Y \le x) = P(F_X^{-1}(U) \le x) = P(U \le F_X(x)) = F_U(F_X(x))$$
$$= F_X(x)$$

$U \sim \mathrm{Unif}(0,1)$

$G_Y(x) = F_X(x) \Rightarrow X$ și $Y = F_X^{-1}(U)$ au aceeași repartiție (identic d.p.d.v. prob.)

Metoda inversării

- se aplică pentru a genera numere pseudo–aleatoare ca valori de observație asupra unei variabile aleatoare X, ce are funcția de repartiție inversabilă.
- dacă F_X este strict crescătoare ,atunci ea este inversabilă.
- Pentru orice $u \in (0,1)$, $F_X^{-1}(u) \in \mathbb{R}$.
- u ca valoare de observație asupra unei variabile aleatoare $U \sim$Unif[0,1), x este valoare de observație asupra variabilei $F_X{-}1(U)$.
- determinăm distribuția de probabilitate a variabilei $Y = F_X^{-1}(U)$:

Propoziție

Fie $U \sim [0,1)$ și F_X o funcție de repartiție strict crescătoare și continuă pe intervalul de lungime minimă din $\mathbb{R}$, pe care variabila aleatoare X ia valori cu probabilitatea 1.
Atunci variabila aleatoare

$$Y = F_X^{-1}(U)$$

are aceeași funcție de repartiție ca și variabila X, adică Y și X sunt identic distribuite și nu se disting din punct de vedere probabilist.

Demo: $U \sim \mathrm{Unif}[0,1) \Rightarrow F_U(x) = x, \; x \in [0,1)$ și $F_U(x) = 0$, în rest.
Vom determina funcția de repartiție G_Y a variabilei $Y = F_X^{-1}(U)$:

$$G_Y(x) = P(Y \le x) = P(F_X^{-1}(U) \le x) = P(U \le F_X(x)) = F_U(F_X(x)) = F_X($$

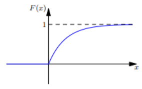
Prin urmare $G_Y(x) = F_X(x), \quad \forall x$, adică funcția de repartiție a variabilei aleatoare $Y = F_X^{-1}(U)$ este chiar F_X.

Metoda inversării aplicate distribuției exponențiale

Fie

- Funcția de repartiție: $F_X(x) = \begin{cases} 0 & \text{dacă } x < 0 \\ 1 - e^{-x/\theta} & \text{dacă } x \geq 0 \end{cases}$

$\lim_{x \to 0} F_X(x) = 1$



- F_X este strict crescătoare pe intervalul $[0, \infty)$
- $\forall u \in [0, 1)$, $F^{-1}(u) \in [0, \infty)$.
- O v. a. exponențial distribuită ia valori pozitive cu probabilitatea 1: $P(X \geq 0) = 1 - P(X < 0) = 1 - F_X(0) = 1 - 0 = 1$
- prin metoda inversării putem simula X.
- Din $1 - e^{-x/\theta} = u$, avem $x = F^{-1}(u) = -\theta \ln(1 - u)$