

- 0/ Probabilités, statistiques comme outils informatiques.
- 1/ Rappels de probabilités
- 2/ Génération aléatoire (principes et méthodes). **ON EST ICI.**
- 3/ Rappels et éléments de statistiques.
- 4/ Evaluation de performances.
- 5/ Modèles mathématiques et analyse.
- 6/ Simulation.
- 7/ Méthodologie de l'évaluation de performances

# Génération aléatoire

# Génération aléatoire

- ➊ Introduction & motivations.
- ➋ Les premiers exemples.
- ➌ Les générateurs à  $k$  pas. **ON EST ICI.**
- ➍ Générateur de loi uniforme. **ON EST ICI.**

## Détection d'un G1P, améliorations possibles?

- Boîte noire ) suite ultimement périodique!  
**QUESTION** : comment savoir qu'il s'agit d'un G1P?
- Comment améliorer?

## Détection d'un G1P, améliorations possibles?

- Boîte noire ) suite ultimement périodique!  
**QUESTION** : comment savoir qu'il s'agit d'un G1P?
- Comment améliorer?

## Propriétés d'un G1P

Soit  $(x_n)$  une suite ultimement périodique alors les deux propriétés suivantes sont équivalentes :

- (i) La suite est produite par un G1P (rappel : G1P est défini par  $(F, x_0, f)$   $x_{n+1} = f(x_n)$  dans l'ens.  $F$ )
- (ii)  
$$\exists n, m, x_n = x_m \text{ ) } x_{n+1} = x_{m+1} .$$

On peut construire des générateurs à deux pas du type “Fibonacci”:

$$x_0, x_1, x_{n+2} = f(x_n, x_{n+1})$$

On peut construire des générateurs à deux pas du type “Fibonacci”:

$$x_0, x_1, x_{n+2} = f(x_n, x_{n+1})$$

Cependant, on remarque vite que cette nouvelle suite n'est rien d'autre qu'une projection du G1P défini par le triplet

$(F^2, g, X_0 = (x_0, x_1))$  où  $g$  est donnée par

$$g(X) = g(x, y) = (y, f(x, y)) .$$

Donc comme pour les G1P, pour les G2P on peut montrer la propriété suivante.



Donc comme pour les G1P, pour les G2P on peut montrer la propriété suivante.

### Propriété d'un G2P

Soit  $(x_n)$  une suite ultimement périodique alors les deux propriétés suivantes sont équivalentes :

(i) La suite est produite par un G2P ( G2P est défini par  $(F, f, (x_0, x_1))$  dans l'ens.  $F, f : F^2 \rightarrow F$ )

(ii)

$$\exists n, m, [x_n, x_{n+1}] = [x_m, x_{m+1}] \Rightarrow x_{n+2} = x_{m+2}.$$

Donc comme pour les G1P, pour les G2P on peut montrer la propriété suivante.

### Propriété d'un G2P

Soit  $(x_n)$  une suite ultimement périodique alors les deux propriétés suivantes sont équivalentes :

(i) La suite est produite par un G2P ( G2P est défini par  $(F, f, (x_0, x_1))$  dans l'ens.  $F, f : F^2 \rightarrow F$ )

(ii)

$$\exists n, m, [x_n, x_{n+1}] = [x_m, x_{m+1}] \Rightarrow x_{n+2} = x_{m+2}.$$

### Conséquences

Les algorithmes de Brent et Floyd s'appliquent sur les vecteurs!

$$x_0 = 1, x_1 = 1 \text{ et } x_{n+2} = x_{n+1} + x_n \text{ MOD } 5.$$

$$x_0 = 1, x_1 = 1 \text{ et } x_{n+2} = x_{n+1} + x_n \text{ MOD } 5.$$

On a la suite 1, 1, 1, 5, 4, 5, 1, 6, 6, 1, 5, 4, 5, 1, 6, 6, 1, 5, 4, 5, 1, On  
peut écrire tout ceci sous la forme matricielle:

$$x_0 = 1, x_1 = 1 \text{ et } x_{n+2} = x_{n+1} + x_n \text{ MOD } 5.$$

On a la suite 1, 1, 1, 5, 4, 5, 1, 6, 6, 1, 5, 4, 5, 1, 6, 6, 1, 5, 4, 5, 1, On peut écrire tout ceci sous la forme matricielle:

$$\begin{pmatrix} x_0 \\ x_1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}; \begin{pmatrix} x_{n+2} \\ x_{n+1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_{n+1} \\ x_n \end{pmatrix}.$$

En utilisant le produit par blocks, nous avons

$$x_0 = 1, x_1 = 1 \text{ et } x_{n+2} = x_{n+1} + x_n \text{ MOD } 5.$$

On a la suite 1, 1, 1, 5, 4, 5, 1, 6, 6, 1, 5, 4, 5, 1, 6, 6, 1, 5, 4, 5, 1, On peut écrire tout ceci sous la forme matricielle:

$$\begin{pmatrix} x_0 \\ x_1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}; \begin{pmatrix} x_{n+2} \\ x_{n+1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_{n+1} \\ x_n \end{pmatrix}.$$

En utilisant le produit par blocks, nous avons

$$\begin{pmatrix} x_{n+1} & x_n \\ x_{n+2} & x_{n+1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n \begin{pmatrix} 1 & 2 \\ 1 & 1 \end{pmatrix}$$

Par conséquent, la période est celle de  $\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n$  (modulo 5).

Ce sont les générateurs définis par une **réurrence linéaire d'ordre 2**.

$$x_0, x_1 \text{ donnés et } x_{n+2} = ax_n + bx_{n+1}.$$

Cette récurrence est liée à l'équation caractéristique  $r^2 = a + br$ . On observe :

Ce sont les générateurs définis par une **réurrence linéaire d'ordre 2**.

$$x_0, x_1 \text{ donnés et } x_{n+2} = ax_n + bx_{n+1}.$$

Cette récurrence est liée à l'équation caractéristique  $r^2 = a + br$ . On observe :

### Proposition

Pour qu'une suite de puissances  $(r^n)_{n \geq 0}$  vérifie une récurrence linéaire d'ordre 2 il faut et il suffit qu'elle la vérifie pour  $n = 0$ , soit  $r^2 = a + br$ .



Ce sont les générateurs définis par une **réurrence linéaire d'ordre 2**.

$$x_0, x_1 \text{ donnés et } x_{n+2} = ax_n + bx_{n+1}.$$

Cette récurrence est liée à l'équation caractéristique  $r^2 = a + br$ . On observe :

### Proposition

Pour qu'une suite de puissances  $(r^n)_{n \geq 0}$  vérifie une récurrence linéaire d'ordre 2 il faut et il suffit qu'elle la vérifie pour  $n = 0$ , soit  $r^2 = a + br$ .

### Vers des générateurs efficaces

Il faut paramétrer de façon à obtenir de grandes périodes. On peut décomposer en petits générateurs dont on peut maîtriser les périodes: ce seront les suites de la forme  $(r^n)$  et  $(nr^n)$ .

On a vu que la suite  $(r^n)$  modulo  $p$  est produite par un GL1P et la suite  $(nr^n)$  modulo  $p$  par un GL2P.

### Proposition

Soit  $m > 0, \alpha, \beta, r_1, r_2$  entiers. La suite  $(\alpha r_1^n + \beta r_2^n)_n$  est produite par le générateur

$$x_0 = \alpha + \beta, x_1 = \alpha r_1 + \beta r_2 \text{ ET } x_{n+2} = (r_1 + r_2)x_{n+1} - (r_1 r_2)x_n.$$

Dans les mêmes conditions, la suite  $(\alpha r^n + \beta nr^n)$  est produite par

$$\alpha + \beta, \alpha r + \beta r \text{ ET } x_{n+2} = 2rx_{n+1} - r^2x_n.$$

[MITCHELL] et [MOORE] ont écrit le générateur suivant:

$$X_n = (X_{n-24} + X_{n-55}) \text{ MOD } m$$

où les termes initiaux  $X_0 \dots X_{54}$  sont des entiers arbitraires non tous pairs. La période d'une telle suite est  $2^f(2^{55} - 1)$  pour  $m = 2^a$  et  $0 < f < a$  (voir [KNUTH]). Une telle suite est difficile dans notre cadre à analyser.

D'autres générateurs sont à trouver! Les très bons générateurs sont souvent (très) difficiles à analyser (expérimentalement et analytiquement).

On suppose disposer d'un générateur de v.a. uniformes  $(0, 1)$ , noté  $U[0, 1]$ , ou d'un générateur pseudo-aléatoire  $(0, M)$  correct. On note par  $x$  'resp.  $x_i$ ) la v.a. produite par  $U[0, 1]$  (resp. la  $i$ -ème occurrence d'une telle production).

On suppose disposer d'un générateur de v.a. uniformes  $(0, 1)$ , noté  $U[0, 1]$ , ou d'un générateur pseudo-aléatoire  $(0, M)$  correct. On note par  $x$  'resp.  $x_i$ ) la v.a. produite par  $U[0, 1]$  (resp. la  $i$ -ème occurrence d'une telle production).

### v.a. binomiale $B(n, p)$

**Définition :** nombre de succès sur  $n$  tentatives indépendantes, chacune ayant une probabilité  $p$  de réussir.

**Cas particulier :** Bernoulli  $B(1, p)$ .

On suppose disposer d'un générateur de v.a. uniformes  $(0, 1)$ , noté  $U[0, 1]$ , ou d'un générateur pseudo-aléatoire  $(0, M)$  correct. On note par  $x$  resp.  $x_i$ ) la v.a. produite par  $U[0, 1]$  (resp. la  $i$ -ème occurrence d'une telle production).

### v.a. binomiale $B(n, p)$

**Définition :** nombre de succès sur  $n$  tentatives indépendantes, chacune ayant une probabilité  $p$  de réussir.

**Cas particulier :** Bernoulli  $B(1, p)$ .

### Bernoulli

Pour générer une occurrence de Bernoulli  $y$  de paramètre  $p$ , il suffit de poser:

$y = 1$  si  $x < p$  et  $y = 0$  sinon.

On peut distinguer trois cas:

- Si  **$n$  est petit**, on utilise le fait que  $B(n, p)$  est la somme de  $n$  Bernoulli :

On génère  $x_1, \dots, x_n$ , on déduit  $y_1, \dots, y_n$  et  $y = \sum_{i=1}^n y_i$ .

On peut distinguer trois cas:

- Si  **$n$  est petit**, on utilise le fait que  $B(n, p)$  est la somme de  $n$  Bernoulli :

On génère  $x_1, \dots, x_n$ , on déduit  $y_1, \dots, y_n$  et  $y = \sum_{i=1}^n y_i$ .

- Si  **$n$  est pas trop grand**, on utilise la méthode dite de **la fonction inverse** (voir plus loin).



On peut distinguer trois cas:

- Si  **$n$  est petit**, on utilise le fait que  $B(n, p)$  est la somme de  $n$  Bernoulli :

On génère  $x_1, \dots, x_n$ , on déduit  $y_1, \dots, y_n$  et  $y = \sum_{i=1}^n y_i$ .

- Si  **$n$  est pas trop grand**, on utilise la méthode dite de **la fonction inverse** (voir plus loin).
- Si  **$n$  est très grand**, on utilise la loi normale et l'approximation

$$B(n, p) \approx N(np, \sqrt{np(1-p)}).$$

**Idée :** On veut simuler une v.a.  $X$  de fonction de répartition  $F$ .

### Proposition

Soit  $X$  une v.a. de fonction de répartition  $F$  strictement croissante. On a

$$F(X) \sim U[0, 1] .$$

**Idée :** On veut simuler une v.a.  $X$  de fonction de répartition  $F$ .

## Proposition

Soit  $X$  une v.a. de fonction de répartition  $F$  strictement croissante. On a

$$F(X) \sim U[0, 1] .$$

## Preuve.

On pose  $u = F(x)$ . Donc  $x = F^{-1}(u)$ . Par définition,  $F(x) = P[X \leq x]$ . Donc

$$F(F^{-1}(u)) = P[X \leq F^{-1}(u)] \underbrace{= u}_{\text{par déf. de la réciproque}}$$

## La méthode de la fonction inverse

**Idée :** On veut simuler une v.a.  $X$  de fonction de répartition  $F$ .

### Proposition

Soit  $X$  une v.a. de fonction de répartition  $F$  strictement croissante. On a

$$F(X) \stackrel{\text{d}}{=} U \quad \text{avec } U \sim \text{Uniforme}(0,1)$$

**Idée :** On veut simuler une v.a.  $X$  de fonction de répartition  $F$ .

## Proposition

Soit  $X$  une v.a. de fonction de répartition  $F$  strictement croissante. On a

$$F(X) \sim U[0, 1] .$$

## Preuve.

On pose  $u = F(x)$ . Donc  $x = F^{-1}(u)$ . Par définition,  $F(x) = P[X \leq x]$ . Donc

$$F(F^{-1}(u)) = P[X \leq F^{-1}(u)] \underbrace{= u}_{\text{par déf. de la réciproque}}$$

Comme  $F$  est strictement croissante

$$P[X \leq F^{-1}(u)] = P[F(X) \leq u] .$$

On a donc  $u = P[F(X) \leq u]$  qui n'est rien d'autre que la déf. de la loi uniforme.

## Exemple: la loi exponentielle

$X$  suit une loi exponentielle de paramètre  $\lambda > 0$  si sa fonction de répartition est  $F(x) = 1 - e^{-\lambda x}$ .

## Exemple: la loi exponentielle

$X$  suit une loi exponentielle de paramètre  $\lambda > 0$  si sa fonction de répartition est  $F(x) = 1 - e^{-\lambda x}$ . Cette fonction est inversible et est strictement croissante. On a  $F^{-1}(u) = \frac{1}{\lambda} \log \frac{1}{1-u}$ . On pourra poser  $X = -\log(1-U)/\lambda$  mais  $X = \frac{\log U}{\lambda}$  convient!

## Exemple: la loi exponentielle

$X$  suit une loi exponentielle de paramètre  $\lambda > 0$  si sa fonction de répartition est  $F(x) = 1 - e^{-\lambda x}$ . Cette fonction est inversible et est strictement croissante. On a  $F^{-1}(u) = -\frac{1}{\lambda} \log(1 - u)$ . On pourra poser  $X = -\log(1 - U)/\lambda$  mais  $X = \frac{\log U}{\lambda}$  convient!

### Remarques.

L'hypothèse de la connaissance de  $F^{-1}$  n'a de sens que si  $F$  est strictement croissante. Mais même dans ce cas, il se peut que  $F^{-1}$  existe mais n'ait pas d'expression simple. C'est le cas de la loi normale

$$F(x) = \frac{1}{2\pi} \int_{-\infty}^x e^{-t^2/2} dt.$$



Soit  $X$  une v.a. à valeurs dans  $\{1, \dots, K\}$ , on note

$$p_k = \mathbb{P}[X = k] \quad (\text{remarquez que } \sum_{k=1}^K p_k = 1)$$

On note  $P_k$  le cumul, i.e.  $P_0 = 0$  et  $P_k = \sum_{i=0}^k p_i$  (et donc  $P_K = 1$ ).  
On a donc un algorithme pour générer  $X$  :

Soit  $X$  une v.a. à valeurs dans  $\{1, \dots, K\}$ , on note

$$p_k = \mathbb{P}[X = k] \quad (\text{remarquez que } \sum_{k=1}^K p_k = 1)$$

On note  $P_k$  le cumul, i.e.  $P_0 = 0$  et  $P_k = \sum_{i=0}^k p_i$  (et donc  $P_K = 1$ ).  
On a donc un algorithme pour générer  $X$  :

## Algo. pour loi discrète

- On tire  $U$ .
- Ensuite, on a

$$X = \sum_{i=1}^K i \mathbf{1}_{P_{i-1} \leq U < P_i}$$

On veut simuler la loi de Poisson de paramètre  $\lambda$ .

$$X \sim \text{Poisson}(\lambda) \quad ( ) \quad p_k = \mathbb{P}[X = k] = e^{-\lambda} \frac{\lambda^k}{k!} \text{ (pour } k \geq 0 \text{)}.$$

On veut simuler la loi de Poisson de paramètre  $\lambda$ .

$$X \sim \text{Poisson}(\lambda) \quad ( ) \quad p_k = \mathbb{P}[X = k] = e^{-\lambda} \frac{\lambda^k}{k!} \text{ (pour } k \geq 0 \text{)}.$$

On remarque que  $p_{k+1} = \frac{\lambda}{k+1} p_k$ . On peut donc utiliser l'algorithme précédent en utilisant les cumulés et en décalant  $P_1 = 0$ ,  $P_0 = e^{-\lambda}$ .

On utilise la propriété suivante. Si des évènements surviennent à des dates séparées par des durées exponentielles de paramètre  $\lambda$ , le nombre d'évènements survenant en une unité de temps suit une loi de Poisson de paramètre  $\lambda$ . On simule des v.a. i.i.d.  $Y_1, Y_2, \dots, Y_g$  telles que  $Y_i \sim \text{expo}(\lambda)$ . La v.a.

$$X = \sum_{k=0}^{\infty} k \mathbf{1}_{\{Z_k \leq 1 < Z_{k+1}\}}$$

(où  $Z_k = \sum_{i=1}^k Y_i$ ) est Poisson( $\lambda$ ).

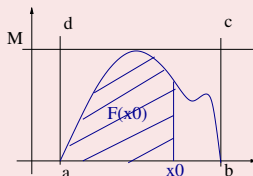
Il faut donc simuler des v.a. expo.  $\lambda$  et compter le nombre de simulations nécessaires pour dépasser 1 ou bien simuler des v.a. expo. de paramètre 1 et compter le nombre nécessaire pour dépasser  $\lambda$ .

# Algorithme de rejet pour loi “quelconque”

On veut simuler une v.a.  $X$  de densité  $f$  et de loi de répartition  $F$

## Hypothèses

- a)  $f$  est à support compact ( $f$  est nulle en dehors d'un intervalle  $[a, b]$ ).
- b)  $f$  est majorée, i.e.  $\exists x \in [a, b] : f(x) \leq M$ .



- 1 On tire un point  $A = (X, Y)$  uniformément dans le rectangle  $(abcd)$ :  
On tire  $U_1$ :  $X = a + (b - a)U_1$   
On tire  $U_2$ :  $Y = MU_2$
- 2 Si  $Y \leq f(X)$ , on garde  $X$  sinon on retire (dehors!).
- 3 On réitère jusqu'à  $Y \leq f(X)$ .

- 1 On tire un point  $A = (X, Y)$  uniformément dans le rectangle  $(abcd)$ :  
On tire  $U_1$ :  $X = a + (b - a)U_1$   
On tire  $U_2$ :  $Y = MU_2$
- 2 Si  $Y \leq f(X)$ , on garde  $X$  sinon on retire (dehors!).
- 3 On réitère jusqu'à  $Y \leq f(X)$ .

### Théorème

$X$  ainsi obtenue a bien une densité  $f$  et une fonction de répartition  $F$ .



Il faut montrer que  $P[X \leq x_0] = F(x_0)$ . On a  
 $P[X \leq x_0] = P[X \leq x_0 / \text{on garde le point } A = (X, Y)]$ .  
Donc,

$$P[X \leq x_0] = P[X \leq x_0 / Y \leq f(X)] = \frac{P[X \leq x_0, Y \leq f(X)]}{P[Y \leq f(X)]}.$$

Or

$$P[X \leq x_0 / Y \leq f(X)] = \frac{\text{surface zone hachurée}}{\text{surface du rectangle } (abcd)} = \frac{F(x_0)}{M(b-a)}$$

et

$$P[Y \leq f(X)] = \frac{\text{surface sous la courbe}}{\text{surface du rectangle } (abcd)} = \frac{1}{M(b-a)}$$

La loi normale a la fonction de répartition

$$F(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-t^2/2} dt$$

On ne peut pas appliquer les méthodes précédentes (support non compact & inverse difficile ...).

La loi normale a la fonction de répartition

$$F(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-t^2/2} dt$$

On ne peut pas appliquer les méthodes précédentes (support non compact & inverse difficile ...). Néanmoins en appliquant le th. central limite (CLT), on a une méthode très simple de génération de v.a. normales. On sait que si  $x$  est une v.a. d'espérance  $\mu$  et de variance  $\sigma^2$  alors en sommant  $n$  v.a. i.i.d on a (CLT)

$$\frac{\text{Somme} - n\mu}{\sqrt{n}\sigma} \xrightarrow{\mathcal{L}} N(0, 1)$$

La loi normale a la fonction de répartition

$$F(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-t^2/2} dt$$

On ne peut pas appliquer les méthodes précédentes (support non compact & inverse difficile ...). Néanmoins en appliquant le th. central limite (CLT), on a une méthode très simple de génération de v.a. normales. On sait que si  $x$  est une v.a. d'espérance  $\mu$  et de variance  $\sigma^2$  alors en sommant  $n$  v.a. i.i.d on a (CLT)

$$\frac{\text{Somme} - n\mu}{\sqrt{n}\sigma} \xrightarrow{\mathcal{L}} N(0, 1)$$

Par exemple en tirant 12 v.a du générateur  $U[0, 1]$  on a  $E(U) = \frac{1}{2}$  et  $\sigma^2(U) = \frac{1}{12}$ . On simule 12 v.a  $U[0, 1]$ . On fait la somme. La variance de cette somme vaut 1 et il faut la centrer en lui retranchant son espérance:

$$X = \sum_{i=1}^{12} U_i - 6.$$

Cette loi est très proche de la loi normale  $N(0, 1)$ .

On tire  $U_1$  et  $U_2$  et on définit  $R = \sqrt{2 \log U_1}$  et  $\theta = 2\pi U_2$ . On pose  $X = R \cos \theta$  et  $Y = R \sin \theta$ .  $X$  et  $Y$  sont deux  $N(0, 1)$ . Pour avoir  $Z \sim N(\mu, \sigma^2)$  on fait la transformation

$$Z = \mu + \sigma X.$$