Simulation de variables aléatoires

Matlab , comme tous les autres langages de programmation, contient une commande `rand` qui est un *générateur de nombres pseudo-aléatoires*. En première approximation, on peut considérer que ses appels successifs fournissent des réalisations de variables aléatoires indépendantes de loi uniforme sur l'intervalle $[0,1]$. Pour simuler d'autres variables aléatoires, le problème général est donc de se ramener à l'utilisation de `rand` (même si **Matlab** et la stixbox dispose aussi d'autres fonctions comme `randn`, etc.), comme dans les exemples suivants :

1. `1+3*rand+2*rand`
2. `1+floor(3*rand)+floor(2*rand)`
3. `rand∧2+floor(2*rand)`

Comme référence sur ce qui suit on pourra consulter *Introduction to Probability Models* de Sheldon M. Ross.

1 Méthode d'inversion

Soit μ la loi d'une variable aléatoire réelle X à simuler, et $F : \mathbb{R} \mapsto [0,1]$ sa fonction de répartition définie par $F(x) = \mathbb{P}(X \leqslant x)$: c'est une fonction croissante continue à droite qui tend vers 0 en $-\infty$ et vers 1 en $+\infty$. Son **inverse généralisé** F^- est défini par :

$$F^-(y) = \inf\{x \in \mathbb{R} : F(x) \geqslant y\}.$$

La méthode d'inversion s'appuie sur la propriété suivante :

$$\boxed{\text{Si } U \sim \mathcal{U}([0,1]), \text{ alors } F^-(U) \sim \mu}$$

La méthode d'inversion permet de simuler *très simplement* certaines variables aléatoires quand F^- est aisément manipulable.

1.1 Variables aléatoires continues

Quand F réalise un homéomorphisme entre le support de μ et l'intervalle $]0,1[$, F^- est simplement F^{-1}. Si cette fonction admet une expression simple, on obtient un algorithme immédiat de simulation.

Symbole	Nom complet	Densité
$\mathcal{E}(\lambda)$	loi exponentielle de paramètre $\lambda > 0$	$\lambda e^{-\lambda x}\mathbb{1}_{[0,+\infty[}(x)$
$\mathcal{C}$	loi de Cauchy	$\frac{1}{\pi}\frac{1}{1+x^2}$
$Ra(\sigma)$	loi de Rayleigh de paramètre $\sigma > 0$	$\frac{x}{\sigma^2}e^{-\frac{x^2}{2\sigma^2}}\mathbb{1}_{[0,+\infty[}(x)$
$\mathcal{P}a(a,b)$	loi de Pareto de paramètres $a > 0$ et $b > 0$	$\frac{ab^a}{x^{a+1}}\mathbb{1}_{[b,+\infty[}(x)$
$\mathcal{W}(a,b)$	loi de Weibull de paramètres $a > 0$ et $b > 0$	$\frac{a}{b^a}x^{a-1}e^{-\left(\frac{x}{b}\right)^a}\mathbb{1}_{[0,+\infty[}(x)$

Quant aux lois pour lesquelles on ne dispose pas d'une expression analytique pour F^-, on peut utiliser une procéder à la résolution numérique de l'équation $F(x) = y$.

Symbole	Nom complet	Densité
$\mathcal{N}(\mu,\sigma^2)$	loi normale de paramètres μ et $\sigma^2 > 0$	$\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(x-\mu)^2}{\sigma^2}}$
$\Gamma(\lambda,a)$	loi Gamma de paramètres $\lambda > 0$ et $a > 0$	$\frac{\lambda^a}{\Gamma(a)}x^{a-1}e^{-\lambda x}\mathbb{1}_{[0,+\infty[}(x)$

1.2 Variables aléatoires discrètes

Quant F est une fonction en escaliers, F^- l'est aussi. Soit $\{x_n : n \in \mathbb{N}\}$ le support de μ. On peut interpréter la méthode d'inversion de la façon suivante : soit $(I_n)_n$ la partition de l'intevalle $[0,1]$ définie par

$$I_n = [p_0 + \ldots + p_{n-1}, p_0 + \ldots + p_n[.$$

$$\boxed{\text{Si } U \sim \mathcal{U}([0,1]), \text{ alors } \textstyle\sum_{n\in\mathbb{N}} x_n \mathbb{1}_{I_n}(U) \sim \mu.}$$

Symbole	Nom complet	Densité
$B(p)$	loi de Bernoulli de paramètre $p \in [0,1]$	$(1-p)\delta_0 + p\delta_1$
$B(n,p)$	loi binomiale de paramètres $n \in \mathbb{N}^*$ et $p \in [0,1]$	$\sum_{k=0}^{n} \binom{n}{k} p^k (1-p)^{n-k} \delta_k$
$U(\{x_1,\ldots,x_n\})$	loi uniforme sur $\{x_1,\ldots,x_n\}$	$\sum_{k=1}^{n} x_k \delta_k$
$\mathcal{G}(p)$	loi géométrique de paramètre $p \in [0,1]$	$\sum_{k\in\mathbb{N}^*} p(1-p)^{k-1}\delta_k$
$\mathcal{P}(\lambda)$	loi de Poisson de paramètre $\lambda > 0$	$\sum_{k\in\mathbb{N}} e^{-\lambda}\frac{\lambda^k}{k!}\delta_k$

Au delà de ces exemples, on programmera une fonction `rmulti` prenant en argument un vecteur p et renvoyant une réalisation d'une variable aléatoire X à valeur dans $\{1,\ldots,\text{length}(p)\}$ telle que $\mathbb{P}(X=i) = p(i)$. On verra plus loin que d'autres méthodes sont plus efficaces dans certains cas (notamment pour la loi binomiale !).

1.3 Application aux statistiques d'ordre

Pour $(x_1,\ldots,x_n) \in \mathbb{R}^n$, on note $\left(x_{(1)},\ldots,x_{(n)}\right)$ sa *statistique d'ordre*, c'est-à-dire le n-uplet contenant les mêmes éléments rangés par ordre croissant : $\{x_{(1)},\ldots,x_{(n)}\} = \{x_1,\ldots,x_n\}$ et $x_{(1)} \leqslant \ldots \leqslant x_{(n)}$.

Soit $(X_1,\ldots,X_n)$ un n-échantillon de loi μ, et soit $(U_1,\ldots,U_n)$ un n-échantillon de variables uniformément distribuées sur $[0,1]$;

$$\boxed{\left(F^-\left(U_{(1)}\right),\ldots,F^-\left(U_{(n)}\right)\right) \text{ a même loi que } \left(X_{(1)},\ldots,X_{(n)}\right).}$$

Si l'on souhaite juste simuler $X_{(n)} = \max_{k=1:n} X_k$, il suffit de remarquer que sa fonction de répartition est donnée par $F_{(n)}(x) = F(x)^n$:

$$\boxed{\text{Si } U \sim \mathcal{U}([0,1]),\ F^-\left(U^{\frac{1}{n}}\right) \text{ a même loi que } X_{(n)}.}$$

2 Méthode d'acceptation-rejet

Soit $(U_n)_n$ une suite de VA iid de loi $\mathcal{U}([0,1])$, et (X_n) de VA iid de loi ν telle que μ soit absolument continue par rapport à ν, et telle qu'il existe un réel m vérifiant : $\forall x \in \mathbb{R}$, $\frac{d\mu}{d\nu}(x) \leqslant m$. Soit $T_m = \inf\{n \geqslant 1 : mU_n \leqslant \frac{d\mu}{d\nu}(X_n)\}$. Alors

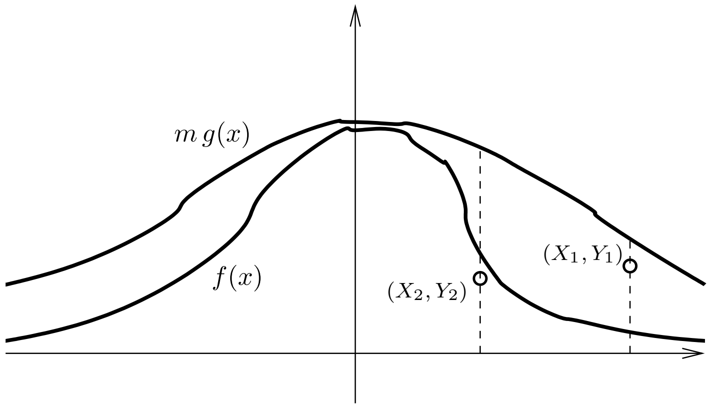
$$\boxed{X_{T_m} \sim \mu\,;\ X_{T_m} \text{ et } T_m \text{ sont indépendantes, et } T_m \sim \mathcal{G}\left(\tfrac{1}{m}\right).}$$

Ainsi, le nombre d'essai avant d'obtenir une valeur acceptée est en moyenne m.

Pour comprendre cette méthode, plaçons nous dans le cas continu où μ et ν ont respectivement pour densités f et g par rapport à la mesure de Lebesgue. Pour toute fonction $h : \mathbb{R} \to \mathbb{R}^+$, soit $D_h = \left\{(x,y) \in \mathbb{R}^2 : 0 \leqslant y \leqslant h(x)\right\}$.

Si (X,Y) suit une loi uniforme sur l'ensemble D_h, alors la loi de X a pour densité $\frac{h}{\int h}$.

Ainsi, si $X \sim \nu$ et si la loi conditionnelle de Y sachant X est $\mathcal{U}([0, mg(X))$, alors $(X,Y) \sim \mathcal{U}(D_{mg})$. Conditionnellement à $\{(X,Y) \in D_f\}$, la loi de (X,Y) est uniforme sur D_f et $X \sim \mu$.



(X_1,Y_1) est rejeté, (X_2,Y_2) est accepté.

On peut ainsi simuler des variables aléatoires gaussiennes en prenant pour référence ν la loi de Laplace : on peut alors prendre $m = \sqrt{\frac{2e}{\pi}} \approx 1,32$. D'autres choix sont possibles : la loi de Cauchy par exemple.

La complexité de cette méthode est aléatoire, mais on connait sa loi. On a bien sûr intérêt à choisir une loi de référence qui "colle" le mieux possible à μ.

Cette méthode permet aussi de simuler des lois conditionnelles : si A est un borélien de $\mathbb{R}$, soit $\mu_{|A} = \frac{1}{\mu(A)}\mathbb{1}_A(x)\mu(dx)$ la restriction de μ à A.Si X_n est une suite de VA iid de loi μ et si $T_A = \inf\{n \geqslant 1 : X_n \in A\}$, alors $X_{T_A} \sim \mu_{|A}$ et $T_A \sim \mathcal{G}(\mu(A))$. On peut ainsi simuler des variables suivant une loi uniforme sur le disque unité (il y a bien sûr d'autres méthodes pour cela !), et obtenir ainsi, par exemple, l'algorithme polaire de simulation de variables aléatoires gaussiennes :

$$\text{Soit } (U,V) \sim \mathcal{U}(D(0,1)),\ R = \sqrt{U^2+V^2},\ \rho = \frac{\sqrt{-4\log R}}{R}, \text{ alors } (\rho U, \rho V) \sim \mathcal{N}(0, I_2).$$

3 Méthodes utilisant la structure de la loi à simuler

3.1 Mélanges

Si μ est un mélange fini de lois, c'est-à-dire s'il existe une loi de probabilité π un ensemble fini I, et des lois de probabilités ν_i telles que

$$\mu = \sum_{i\in I} \pi(i)\nu_i,$$

et si l'on sait facilement simuler sous les lois ν_i, alors on peut penser à simuler X en deux temps :

$$\boxed{\text{Si } Z \sim \pi, \text{ et si conditionnellement à } \{Z = i\} \text{ la loi de } X \text{ est } \nu_i, \text{ alors } X \sim \mu.}$$

Cela s'applique à la loi de Laplace de paramètres μ et $b > 0$, qui a pour densité $f(x) = \frac{b}{2}e^{-b|x-\mu|}$. On peut aussi l'utiliser pour simuler une variable aléatoire de densité, par exemple, $f(x) = \frac{4}{3}e^{-4x} + e^{-2x} + \frac{1}{6}e^{-x}$.

3.2 Théorèmes limites

On peut aussi penser aux théorèmes limites pour des simulations approchées :

- **Loi normale approchée** Si $U_1, \ldots, U_{12}$ sont des VA iid de loi $U([0,1])$, alors $U_1 + \ldots + U_{12} - 6 \approx \mathcal{N}(0,1)$.
- **Loi de Poisson approchée** Soit $\lambda > 0$ et $N \sim \mathcal{P}(\lambda)$. Si $N \in \mathbb{N}$, $p = \frac{\lambda}{n}$, et si $S_n \sim B(n,p)$, alors

$$\sup_{k\geqslant 0} |P(S_n = k) - P(N = k)| \leqslant \frac{2\lambda^2}{n}.$$

Pour prouver ce genre d'inégalités par couplage, voir la *méthode de Chen-Stein*.

3.3 Méthodes ad hoc

Utiliser les propriétés suivantes pour simuler les variables aléatoires concernées.

- **Loi binomiale** Soit $n \in \mathbb{N}$ et $p \in [0,1]$, et $Z_1, \ldots, Z_n$ des VA iid de loi $\mathcal{B}(p)$. Alors

$$Z_1 + \ldots + Z_p \sim B(n,p).$$

- **Loi du premier succès** Soit $p \in [0,1]$, et $(U_n)_n$ une suite de VA iid de loi $\mathcal{U}([0,1])$. Alors

$$N = \inf\{n \geqslant 1 : U_n \leqslant p\} \sim \mathcal{G}(p).$$

- **Troncature** Si $X \sim \mathcal{E}(\lambda)$, alors $\lceil X \rceil \sim \mathcal{G}\left(1 - e^{-\lambda}\right)$.
- **Processus de Poisson** Soit $\lambda > 0$, et $(E_n)_n$ une suite de VA iid de loi $\mathcal{E}(1)$. Alors

$$N = \inf\{n \geqslant 1 : \sum_{k=1}^{n+1} E_k > \lambda\} \sim \mathcal{P}(\lambda).$$

– **Algorithme de Box-Müller** Soient U et V deux VA iid de loi $\mathcal{U}([0,1])$, et soit

$$\begin{pmatrix} \sqrt{-2\log U}\cos 2\pi V \\ \sqrt{-2\log U}\sin 2\pi V \end{pmatrix} \sim \mathcal{N}(0, I_2).$$

– **Loi Gamma** Si $X_1, \ldots, X_n$ sont des VA iid de loi $\mathbb{E}(\lambda)$, alors

$$X_1 + \ldots + X_n \sim \Gamma(\lambda, n).$$

– **Loi du χ^2** Si $X_1, \ldots, X_n$ sont des VA iid de loi $\mathcal{N}(0,1)$, alors

$$X_1^2 + \ldots + X_n^2 \sim \chi^2(n) = \Gamma\left(\frac{1}{2}, n\right).$$

4 Données dépendantes

4.1 Vecteurs gaussiens

Soit $n \in \mathbb{N}^+$, $\mu \in \mathbb{R}^n$ et $C \in \mathcal{M}_n(R)$ une matrice symétrique définie positive : la *factorisation de Choleski* permet d'obtenir $A \in \mathcal{M}_n(R)$ telle que $C = AA^T$. Si $(X_1, \ldots, X_n)$ est un n-uplet de VA iid de loi $\mathcal{N}(0,1)$, alors

$$\boxed{Y = AX + \mu \sim \mathcal{N}(\mu, C).}$$

4.2 Chaînes de Markov

La simulation de chaînes de Markov à espace d'état fini est immédiate grâce à la fonction `rmulti` introduite plus haut.

La fonction ci-contre simule $(X_1, \ldots, X_n)$, où $(X_k)_k$ forme une chaîne de Markov sur l'ensemble $\{1, 2, \ldots, \texttt{length}(p)\}$ de matrice de transition A et de loi initiale p, c'est-à-dire avec $\mathbb{P}(X_k = j | X_{k-1} = i) = A_{ij}$ et $\mathbb{P}(X_1 = i) = p_i$.

```
function X=cdm(p, A, n)
X=zeros(1, n) ;
X(1)=rmulti(p) ;
for k=2 :n
   X(k)=rmulti(A(X(k-1), :)) ;
end
```

Pour simuler une chaîne de Markov à espace d'état infini (par exemple une marche aléatoire), on peut le plus souvent suivre le même schéma en utilisant la loi de transition concernée.

4.3 Loi multinomiale aléatoire

Une loi multinomiale sur un ensemble à k éléments est déterminée par un point du simplexe

$$S_k = \left\{ (x_1, \ldots, x_k) \in [0,1]^k : \sum_{i=1}^{k} x_k = 1 \right\}.$$

La loi du vecteur aléatoire p généré par le programme ci-contre n'est pas uniforme sur S_k, même si tous ses points ont la même loi et sont d'espérance $1/k$.

```
function p=rdist(k)
p=rand(1,k) ;
p=p/sum(p) ;
```

Pour obtenir une loi uniforme sur S_k, on peut :
– utiliser l'idée précédente, mais avec des variables exponentielles ;
– ou tirer $k-1$ variables aléatoires $(u_i)_{i \in \{1,\ldots,k-1\}}$ iid de loi $U([0,1])$, poser $u_0 = 0$ et $u_1 = 1$, et prendre $p_i = u_i - u_{i-1}$ pour $i \in \{1, \ldots, k\}$.