

Nouveaux outils informatiques pour la Statistique exploratoire (=NOISE)

Christian P. Robert

Université Paris Dauphine
<http://www.ceremade.dauphine.fr/~xian>

Licence MI2E, 2006–2007

Outline

- ① Simulation de variables aléatoires
- ② Méthodes de Monte Carlo et algorithme EM
- ③ Méthode du bootstrap
- ④ Statistique non-paramétrique

Chapitre 1 : Simulation de variables aléatoires

- Introduction
- Générateur pseudo-aléatoire
- Distributions non-uniformes (1)
- Distributions non-uniformes (2)
- Méthodes de Markov

Introduction

Besoin de “produire le hasard” par ordinateur

- Evaluer le comportement d’un système complexe (programme, réseau, file d’attente, système de particules, atmosphère, épidémie, actions...)
- Déterminer les propriétés probabilistes d’une procédure statistique non-standard ou sous une loi inconnue [bootstrap]
- Validation d’un modèle probabiliste
- Approcher une espérance/intégrale sous une loi non-standard [loi des grands nombres]
- Maximiser une fonction/vraisemblance faiblement régulière

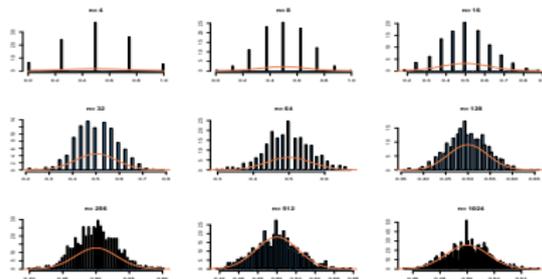
Exemple (TCL pour la loi binomiale)

Si

$$X_n \sim \mathcal{B}(n, p),$$

X_n converge en loi vers la loi normale :

$$\sqrt{n}(X_n - p) \underset{n \rightarrow \infty}{\rightsquigarrow} \mathcal{N}\left(0, \frac{p(1-p)}{n}\right)$$



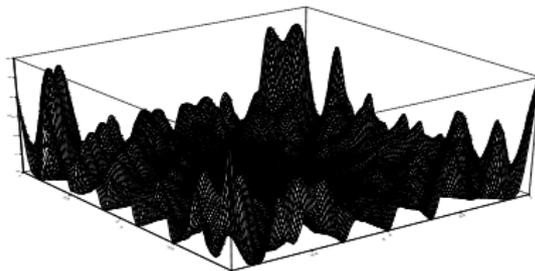
Histogrammes de 1000 réalisations $\mathcal{B}(n, .5)$

Exemple (Minimisation aléatoire)

On considère la fonction

$$h(x, y) = (x \sin(20y) + y \sin(20x))^2 \cosh(\sin(10x)x) \\ + (x \cos(10y) - y \sin(10x))^2 \cosh(\cos(20y)y),$$

à minimiser. (On sait que le minimum global vaut 0 en $(x, y) = (0, 0)$.)



Exemple (Minimisation aléatoire (2))

Au lieu de chercher à résoudre les équations du premier ordre

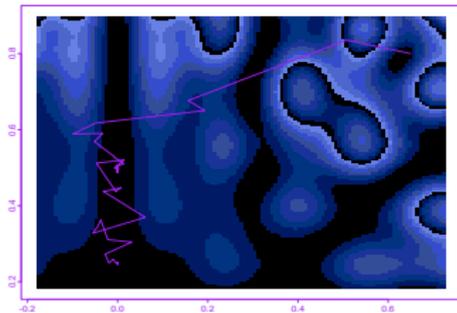
$$\frac{\partial h(x,y)}{\partial x} = 0, \quad \frac{\partial h(x,y)}{\partial y} = 0$$

et à vérifier les conditions du second ordre, on peut générer la suite aléatoire dans \mathbb{R}^2

$$\theta_{j+1} = \theta_j + \frac{\alpha_j}{2\beta_j} \Delta h(\theta_j, \beta_j \zeta_j) \zeta_j$$

où

- ◊ les ζ_j sont uniformes sur le cercle unité $x^2 + y^2 = 1$;
- ◊ $\Delta h(\theta, \zeta) = h(\theta + \zeta) - h(\theta - \zeta)$;
- ◊ (α_j) et (β_j) tendent vers 0

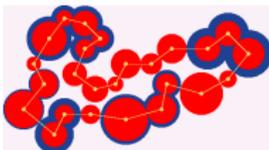


Cas où $\alpha_j = 1/10 \log(1 + j)$ et $\beta_j = 1/j$

Problème du voyageur de commerce

Problème classique d'allocation:

- ◊ Représentant devant visiter un ensemble de n villes
- ◊ Coûts de voyages entre deux villes fixés [et différents]
- ◊ Recherche du coût global minimum



Problème NP-complet

Problème du voyageur de commerce représentatif de problèmes mathématiques durs à résoudre

explosifs
 Nombre de chemins possibles $n!$ et solutions exactes disponibles en temps $O(2^n)$
 Problème à nombreuses applications (réseaux, conception de circuits imprimés, séquençage de génome, etc.)



Concours Procter & Gamble 1962

Problème toujours ouvert



Solution exacte pour 15,112 villes allemandes trouvée en 2001 en 22.6 années CPU.



Résolution pour les 24,978 villes suédoises en 2004 en 84.8 années CPU

Résolution par simulation

Algorithme du **recuit simulé**:

Répéter

- Modifications aléatoires de parties du circuit de coût C_0
- Evaluation du coût C du nouveau circuit
- Acceptation du nouveau circuit avec probabilité

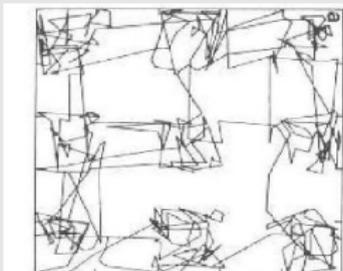
$$\exp\left\{\frac{C_0 - C}{T}\right\} \wedge 1$$

T , **température**, est réduite progressivement.

[Metropolis, 1953]

Illustration

Exemple (400 villes)



$T = 1.2$



Pricing d'options

Calcul complexe d'espérances/valeurs moyennes d'options, $\mathbb{E}[C_T]$, nécessaire pour évaluer le prix d'achat $(1+r)^{-T} \mathbb{E}[C_T]$

Exemple (Options européennes)

Cas où

$$C_T = (S_T - K)^+$$

avec

$$S_T = S_0 \times Y_1 \times \dots \times Y_T, \Pr(Y_i = u) = 1 - \Pr(Y_i = d) = p.$$

Résolution par simulation des binomiales Y_i

Pricing d'options (suite)

Exemple (Options asiatiques)

Modèle en temps continu où

$$C_T = \left(\frac{1}{T} \int_0^T S(t) dt - K \right)^+ \approx \left(\frac{1}{T} \sum_{n=1}^T S(n) - K \right)^+,$$

avec

$$S(n+1) = S(n) \times \exp \{ \Delta X(n+1) \}, \Delta X(n) \stackrel{iid}{\sim} \mathcal{N}(0, \sigma^2).$$

Résolution par simulation des normales ΔX_i

Générateur pseudo-aléatoire

Elément central des méthodes de simulation : elles reposent toutes sur la transformation de variables uniformes $\mathcal{U}(0, 1)$

Definition (**Générateur pseudo-aléatoire**)

Un *générateur pseudo-aléatoire* est une transformation déterministe Ψ de $]0, 1[$ dans $]0, 1[$ telle que, pour toute valeur initiale u_0 et tout n , la suite

$$\{u_0, \Psi(u_0), \Psi(\Psi(u_0)), \dots, \Psi^n(u_0)\}$$

a le même comportement statistique qu'une suite iid $\mathcal{U}(0, 1)$

¡Paradoxe!

Sans appel au "hasard", la suite déterministe $(u_0, u_1 = \Psi(u_0), \dots, u_n = \Psi(u_{n-1}))$ doit ressembler à une suite aléatoire

En R, appel à la procédure

```
runif( )
```

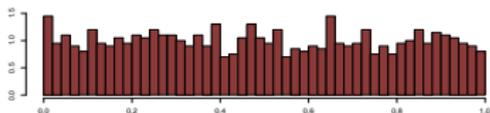
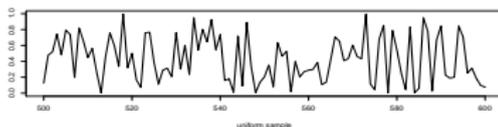
Description:

'runif' generates random deviates.

Example:

```
u = runif(20)
```

'Random.seed' is an integer vector, containing the random number generator (RNG) state for random number generation in R. It can be saved and restored, but should not be altered by the user.



En C, appel à la procédure

rand() / random()

SYNOPSIS

```
# include <stdlib.h>  
long int random(void);
```

DESCRIPTION

The `random()` function uses a non-linear additive feedback random number generator employing a default table of size 31 long integers to return successive pseudo-random numbers in the range from 0 to `RAND_MAX`. The period of this random generator is very large, approximately $16*((2^{**}31)-1)$.

RETURN VALUE

`random()` returns a value between 0 and `RAND_MAX`.

En Scilab, appel à la procédure

rand()

rand() : with no arguments gives a scalar whose value changes each time it is referenced. By default, random numbers are uniformly distributed in the interval (0,1). `rand('normal')` switches to a normal distribution with mean 0 and variance 1. `rand('uniform')` switches back to the uniform distribution

EXAMPLE

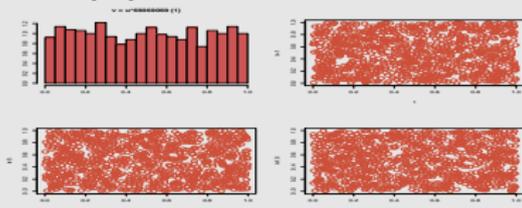
```
x=rand(10,10,'uniform')
```

Exemple (Générateur usuel)

Le générateur congruenciel

$$D(x) = (ax + b) \bmod (M + 1).$$

est de période M pour les bons choix de (a, b) et se transforme en générateur sur $]0, 1[$ par division par $M + 2$.



Conclusion :

Utiliser la fonction appropriée sur l'ordinateur ou le logiciel en service plutôt que de construire un générateur aléatoire de mauvaise qualité

Autres distributions que la loi uniforme (1)

Problème réglé en principe puisque

Théorème (Inversion générique)

Si U est une variable aléatoire uniforme sur $[0, 1]$ et F_X est la fonction de répartition de la variable X , $F_X^{-1}(U)$ a même loi que X

Preuve. On a

$$P(F_X^{-1}(U) \leq x) = P(U \leq F_X(x)) = F_X(x)$$

Note. Si F_X n'est pas strictement croissante, on prend

$$F_X^{-1}(u) = \inf \{x; F_X(x) \geq u\}$$

Applications...

- Loi binomiale, $\mathcal{B}(n, p)$,

$$F_X(x) = \sum_{i \leq x} \binom{n}{i} p^i (1-p)^{n-i}$$

et $F_X^{-1}(u)$ s'obtient numériquement

- Loi exponentielle, $\mathcal{E}xp(\lambda)$,

$$F_X(x) = 1 - \exp(-\lambda x) \quad \text{et} \quad F_X^{-1}(u) = -\log(1-u)/\lambda$$

- Loi de Cauchy, $\mathcal{C}(0, 1)$,

$$F_X(x) = \frac{1}{\pi} \arctan(x) + \frac{1}{2} \quad \text{et} \quad F_X^{-1}(u) = \tan(\pi(u-1/2))$$

Autres transformations...

[Indice]

Trouver des transformations reliant la loi d'intérêt et des lois plus simples/mieux connues

Exemple (Transformation de Box-Müller)

Pour la loi normale $\mathcal{N}(0, 1)$, si $X_1, X_2 \stackrel{i.i.d.}{\sim} \mathcal{N}(0, 1)$,

$$X_1^2 + X_2^2 \sim \chi_2^2, \quad \arctan(X_1/X_2) \sim \mathcal{U}([0, 2\pi])$$

[Jacobien]

Comme χ_2^2 est identique à $\mathcal{E}xp(1/2)$, il vient par inversion

$$X_1 = \sqrt{-2 \log(U_1)} \sin(2\pi U_2) \quad X_2 = \sqrt{-2 \log(U_1)} \cos(2\pi U_2)$$

Exemple

Les lois de Student et de Fisher se déduisent naturellement de la loi normale et de la loi du chi-deux.

Exemple

La loi de Cauchy se déduit de la loi normale par : si

$$X_1, X_2 \stackrel{i.i.d.}{\sim} \mathcal{N}(0, 1), \quad X_1/X_2 \sim \mathcal{C}(0, 1)$$

Exemple

La loi Beta $\mathcal{B}(\alpha, \beta)$, de densité

$$f_X(x) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1},$$

s'obtient à partir de la loi gamma par: si $X_1 \sim \mathcal{G}a(\alpha, 1)$,
 $X_2 \sim \mathcal{G}a(\beta, 1)$, alors

$$\frac{X_1}{X_1 + X_2} \sim \mathcal{B}(\alpha, \beta)$$

Lois multidimensionnelles

Soit à générer dans \mathbb{R}^p

$$(X_1, \dots, X_p) \sim f(x_1, \dots, x_p)$$

dont les composantes ne sont pas nécessairement indépendantes

Cascade rule

$$f(x_1, \dots, x_p) = f_1(x_1) \times f_{2|1}(x_2|x_1) \dots \times f_{p|p-1}(x_p|x_1, \dots, x_{p-1})$$

Implémentation

Simuler pour $t = 1, \dots, T$

① $X_1 \sim f_1(x_1)$

② $X_2 \sim f_{2|1}(x_2|x_1)$

⋮

p. $X_p \sim f_{p|p-1}(x_p|x_1, \dots, x_{p-1})$

Autres distributions que la loi uniforme (2)

- F_X^{-1} rarement disponible
- algorithme résident sur machine seulement pour lois usuelles
- lemme d'inversion ne s'applique qu'en dimension 1
- nouvelle distribution demandant résolution rapide

Méthode d'acceptation-rejet

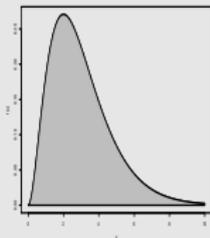
Distribution de densité f à simuler

Théorème (fondamental de la simulation)

La loi uniforme sur le sous-graphe

$$\mathcal{S}_f = \{(x, u); 0 \leq u \leq f(x)\}$$

a comme loi marginale en x la loi de densité f .



Raison :

Loi marginale donnée par

$$\int_0^\infty \mathbb{I}_{0 \leq u \leq f(x)} du = f(x)$$

et indépendance à la constante de normalisation

Exemple

Pour une loi normale, il "suffit" de simuler (u, x) au hasard dans

$$\{(u, x); 0 \leq u \leq \exp(-x^2/2)\}$$

Algorithme d'acceptation-rejet

① Trouver une densité g simulable telle que

$$\sup_x \frac{f(x)}{g(x)} = M < \infty$$

② Générer

$$Y_1, Y_2, \dots \stackrel{i.i.d.}{\sim} g, \quad U_1, U_2, \dots \stackrel{i.i.d.}{\sim} \mathcal{U}([0, 1])$$

③ Prendre $X = Y_k$ où

$$k = \inf\{n; U_n \leq f(Y_n)/Mg(Y_n)\}$$

Théorème (Acceptation-rejet)

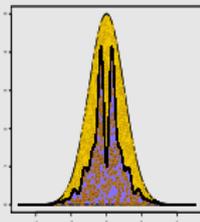
La variable produite par la règle d'arrêt ci-dessous est distribuée suivant la loi f_X

Preuve (1) : On a

$$\begin{aligned} P(X \leq x) &= \sum_{k=1}^{\infty} P(X = Y_k, Y_k \leq x) \\ &= \sum_{k=1}^{\infty} \left(1 - \frac{1}{M}\right)^{k-1} P(U_k \leq f(Y_k)/Mg(Y_k), Y_k \leq x) \\ &= \sum_{k=1}^{\infty} \left(1 - \frac{1}{M}\right)^{k-1} \int_{-\infty}^x \int_0^{f(y)/Mg(y)} du g(y) dy \\ &= \sum_{k=1}^{\infty} \left(1 - \frac{1}{M}\right)^{k-1} \frac{1}{M} \int_{-\infty}^x f(y) dy \end{aligned}$$

Preuve (2)

Si (X, U) est uniforme sur $A \supset B$, la distribution de (X, U) retrainte à B est uniforme sur B .



Propriétés

- Fonctionne sans constante de normalisation
- Ne nécessite pas une borne exacte M
- Autorise le recyclage des Y_k pour une autre loi f (les Y_k refusés ne sont plus de loi g)
- Demande en moyenne M vs Y_k pour un X (mesure d'efficacité)

Exemple

Soit $f(x) = \exp(-x^2/2)$ et $g(x) = 1/(1+x^2)$

$$\frac{f(x)}{g(x)} = (1+x^2) e^{-x^2/2} \leq 2/\sqrt{e}$$

Probabilité d'acceptation $\sqrt{e/2\pi} = 0.66$

Théorème (Enveloppe)

S'il existe une densité g_m , une fonction g_l et une constante M telles que

$$g_l(x) \leq f(x) \leq M g_m(x),$$

alors

- 1 Générer $X \sim g_m(x)$, $U \sim \mathcal{U}_{[0,1]}$;
- 2 Accepter X si $U \leq g_l(X)/M g_m(X)$;
- 3 sinon, accepter X si $U \leq f(X)/M g_m(X)$

donne des variables aléatoires suivant la loi f .

Algorithme du rapport d'uniformes

• Slice sampler

Résultat :

Simulation uniforme sur

$$\{(u, v); 0 \leq u \leq \sqrt{2f(v/u)}\}$$

produit

$$X = V/U \sim f$$

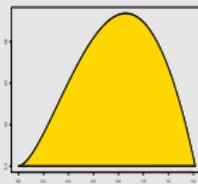
Raison :

Changement de variable $(u, v) \rightarrow (x, u)$ de Jacobien u et loi marginale de x donnée par

$$x \sim \int_0^{\sqrt{2f(x)}} u \, du = \frac{\sqrt{2f(x)}^2}{2} = f(x)$$

Exemple

Pour une loi normale, simuler (u, v) au hasard dans



$$\{(u, v); 0 \leq u \leq \sqrt{2} e^{-v^2/4u^2}\} = \{(u, v); v^2 \leq -4u^2 \log(u/\sqrt{2})\}$$

Slice sampler

Si la simulation uniforme sur

$$\mathfrak{G} = \{(u, x); 0 \leq u \leq f(x)\}$$

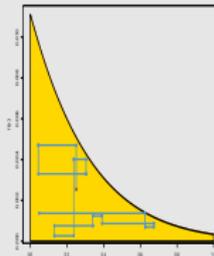
est trop compliquée [à cause de l'inversion en x de $u \leq f(x)$], on peut utiliser une **marche aléatoire** sur \mathfrak{G} :

Slice sampler

Simuler pour $t = 1, \dots, T$

- ① $\omega^{(t+1)} \sim \mathcal{U}_{[0, f(x^{(t)})]}$;
- ② $x^{(t+1)} \sim \mathcal{U}_{\mathfrak{G}(\omega^{(t+1)})}$, où

$$\mathfrak{G}(\omega^{(t+1)}) = \{y; f(y) \geq \omega^{(t+1)}\}.$$



Justification

La **marche aléatoire** se promène uniformément sur \mathcal{G} :

si

$$(U^{(t)}, X^{(t)}) \sim \mathcal{U}_{\mathcal{G}},$$

alors

$$(U^{(t+1)}, X^{(t+1)}) \sim \mathcal{U}_{\mathcal{G}}.$$

Preuve:

$$\begin{aligned} & \Pr((U^{(t+1)}, X^{(t+1)}) \in A \times B) \\ &= \int \int \int_B \int_A \mathbb{I}_{0 \leq u \leq f(x)} \frac{\mathbb{I}_{0 \leq u' \leq f(x)} \mathbb{I}_{f(x') \geq u'}(x')}{f(x) \int \mathbb{I}_{f(y) \geq u'} dy} d(x, u, x', u') \\ &= \int \int_B \int_A f(x) \frac{\mathbb{I}_{0 \leq u' \leq f(x)} \mathbb{I}_{f(x') \geq u'}(x')}{f(x) \int \mathbb{I}_{f(y) \geq u'} dy} d(x, x', u') \\ &= \int \mathbb{I}_{f(x) \geq u'} dx \int_B \int_A \frac{\mathbb{I}_{f(x') \geq u'}(x')}{\int \mathbb{I}_{f(y) \geq u'} dy} d(x', u') \\ &= \int_B \int_A \mathbb{I}_{f(x') \geq u' \geq 0} d(x', u') \end{aligned}$$

Exemple (Loi normale)

Pour la loi normale centrée réduite,

$$f(x) \propto \exp(-x^2/2),$$

un slice sampler est

$$\begin{aligned} \omega|x &\sim \mathcal{U}_{[0, \exp(-x^2/2)]}, \\ X|\omega &\sim \mathcal{U}_{[-\sqrt{-2 \log(\omega)}, \sqrt{-2 \log(\omega)}]} \end{aligned}$$

Note

La méthode fonctionne également si on remplace f par

$$g(x) \propto f(x)$$

Elle se généralise facilement au cas où f se décompose en

$$f(x) = \prod_{i=1}^p f_i(x)$$

Exemple (Loi normale tronquée)

Si on considère à la place la loi normale $\mathcal{N}(-3, 1)$ tronquée à $[0, 1]$, de densité

$$f(x) = \frac{\exp(-(x+3)^2/2)}{\sqrt{2\pi}[\Phi(4) - \Phi(3)]} \propto \exp(-(x+3)^2/2) = \varphi(x),$$

un slice sampler est

$$\begin{aligned}\omega|x &\sim \mathcal{U}_{[0, \exp(-(x+3)^2/2)]}, \\ X|\omega &\sim \mathcal{U}_{[0, 1 \wedge \{-3 + \sqrt{-2 \log(\omega)}\}]}\end{aligned}$$

L'algorithme de Metropolis–Hastings

Généralisation du slice sampler à des contextes où le slice sampler ne peut être utilisé facilement

Idée

Créer une suite $(X_n)_n$ telle que, pour n "assez grand", la densité de la loi de X_n soit environ f

L'algorithme de Metropolis–Hastings (2)

Soit f la densité de la distribution d'intérêt. On choisit une densité conditionnelle

$$q(y|x)$$

dite **instrumentale** ou **propositionnelle**

- facile à simuler
- partout positive là où f est positive

Metropolis–Hastings

Partant de $X^{(t)} = x^{(t)}$,

- ① Générer $Y_t \sim q(y|x^{(t)})$.
- ② Prendre

$$X^{(t+1)} = \begin{cases} Y_t & \text{avec proba. } \rho(x^{(t)}, Y_t), \\ x^{(t)} & \text{avec proba. } 1 - \rho(x^{(t)}, Y_t), \end{cases}$$

où

$$\rho(x, y) = \min \left\{ \frac{f(y)}{f(x)} \frac{q(x|y)}{q(y|x)}, 1 \right\}.$$

Propriétés

- Accepte toujours les déplacements vers des y_t tels que

$$\frac{f(y_t)}{q(y_t|x_t)} \geq \frac{f(x_t)}{q(x_t|y_t)}$$

- Ne dépend pas des constantes de normalisation de f et $q(\cdot|x)$ (à condition qu'elles soient indépendantes de x)
- N'accepte jamais les valeurs de y_t telles que $f(y_t) = 0$
- La suite $(x^{(t)})_t$ peut prendre plusieurs fois la même valeur
- Les $X^{(t)}$ sont des v.a. dépendantes (markoviennes)

↳ liens

Justification

Loi jointe de $(X^{(t)}, X^{(t+1)})$

Si $X^{(t)} \sim f(x^{(t)})$,

$$(X^{(t)}, X^{(t+1)}) \sim f(x^{(t)}) \left\{ \rho(x^{(t)}, x^{(t+1)}) \times q(x^{(t+1)}|x^{(t)}) \right. \\ \left. \begin{array}{l} [Y_t \text{ accepté}] \\ + \int [1 - \rho(x^{(t)}, y)] q(y|x^{(t)}) dy \mathbb{I}_{x^{(t)}}(x^{(t+1)}) \\ [Y_t \text{ rejeté}] \end{array} \right\}$$

Condition de balance

$$\begin{aligned} f(x) \times \rho(x, y) \times q(y|x) &= f(x) \min \left\{ \frac{f(y)}{f(x)} \frac{q(x|y)}{q(y|x)}, 1 \right\} q(y|x) \\ &= \min \{ f(y)q(x|y), f(x)q(y|x) \} \\ &= f(y) \times \rho(y, x) \times q(x|y) \end{aligned}$$

Donc la loi de $(X^{(t)}, X^{(t+1)})$ est la même que celle de $(X^{(t+1)}, X^{(t)})$: si $X^{(t)}$ a la loi f , $X^{(t+1)}$ aussi

Liens avec le slice sampling

Le slice sampler est un cas (très) particulier d'algorithme de Metropolis-Hastings où la probabilité d'acceptation vaut toujours 1

- ① pour la génération de U ,

$$\frac{\mathbb{I}_{0 \leq u' \leq f(x)}}{\mathbb{I}_{0 \leq u \leq f(x)}} \times \frac{f(x)^{-1} \mathbb{I}_{0 \leq u' \leq f(x)}}{f(x)^{-1} \mathbb{I}_{0 \leq u \leq f(x)}} = 1$$

[loi jointe] [loi conditionnelle]

- ② pour la génération de X ,

$$\frac{\mathbb{I}_{0 \leq u \leq f(y)}}{\mathbb{I}_{0 \leq u \leq f(x)}} \times \frac{\int_{\{z: u \leq f(z)\}}(x) \int_{\{z: u \leq f(z)\}} f(z) dz}{\int_{\{z: u \leq f(z)\}}(y) \int_{\{z: u \leq f(z)\}} f(z) dz} = 1$$

[loi jointe] [loi conditionnelle]

Propositions indépendantes

Loi instrumentale q indépendante de $X^{(t)}$, dénotée g comme dans Acceptation-Rejet.

Metropolis-Hastings indépendant

Partant de $X^{(t)} = x^{(t)}$,

- ① Générer $Y_t \sim g(y)$
- ② Prendre

$$X^{(t+1)} = \begin{cases} Y_t & \text{avec proba. } \min \left\{ \frac{f(Y_t) g(x^{(t)})}{f(x^{(t)}) g(Y_t)}, 1 \right\}, \\ x^{(t)} & \text{sinon.} \end{cases}$$

Propriétés

- Alternative à Acceptation-Rejet
- Evite le calcul de $\max f(x)/g(x)$
- Accepte plus souvent qu'Acceptation-Rejet
- Si x_t atteint $\max f(x)/g(x)$, presque identique à Acceptation-Rejet
- Mais la suite des x_t n'est pas indépendante

Exemple (Loi gamma)

Générer une loi $\mathcal{G}a(\alpha, \beta)$ à partir d'une loi instrumentale $\mathcal{G}a(\lfloor \alpha \rfloor, b = \lfloor \alpha \rfloor / \alpha)$, où $\lfloor \alpha \rfloor$ partie entière de α (qui peut être générée comme somme d'exponentielles)

- ① Générer $Y_t \sim \mathcal{G}a(\lfloor \alpha \rfloor, \lfloor \alpha \rfloor / \alpha)$
- ② Prendre

$$X^{(t+1)} = \begin{cases} Y_t & \text{avec proba. } \left(\frac{Y_t}{x^{(t)}} \exp \left\{ \frac{x^{(t)} - Y_t}{\alpha} \right\} \right)^{\alpha - \lfloor \alpha \rfloor} \\ x^{(t)} & \text{sinon.} \end{cases}$$

Metropolis-Hastings à marche aléatoire

Proposition

$$Y_t = X^{(t)} + \varepsilon_t,$$

où $\varepsilon_t \sim g$, indépendant de $X^{(t)}$, et g *symétrique*
Loi instrumentale de densité

$$g(y - x)$$

Motivation

Perturbation *locale* de $X^{(t)}$ / exploration de son voisinage

Metropolis–Hastings à marche aléatoire

Partant de $X^{(t)} = x^{(t)}$

- 1 Générer $Y_t \sim g(y - x^{(t)})$
- 2 Prendre

$$X^{(t+1)} = \begin{cases} Y_t & \text{avec proba. } \min\left\{1, \frac{f(Y_t)}{f(x^{(t)})}\right\}, \\ x^{(t)} & \text{sinon} \end{cases}$$

[symétrie de g]

Propriétés

- Monte toujours et descend parfois (cf. algorithmes de gradient)
- Dépend de la dispersion de g
- Probabilité moyenne d'acceptation

$$e = \int \int \min\{f(x), f(y)\} g(y - x) dx dy$$

- proche de 1 si g très peu dispersée
- loin de 1 si g très dispersée

[Danger!]
[Re-Danger!]

Exemple (Loi normale)

Générer $\mathcal{N}(0, 1)$ fondée sur une perturbation uniforme $[-\delta, \delta]$

$$Y_t = X^{(t)} + \delta \omega_t$$

Probabilité d'acceptation

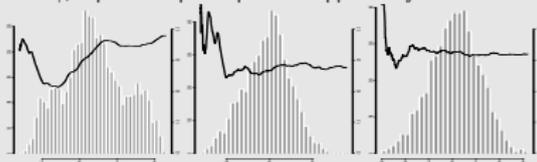
$$\rho(x^{(t)}, y_t) = \exp\{(x^{(t)2} - y_t^2)/2\} \wedge 1.$$

Exemple (Loi normale (2))

Statistiques après 15000 simulations

δ	0.1	0.5	1.0
mean	0.399	-0.111	0.10
variance	0.698	1.11	1.06

Quand $\delta \uparrow$, exploration plus rapide du support de f .



3 échantillons avec $\delta = 0.1, 0.5$ et 1.0 , avec convergence des moyennes empiriques (15000 simulations).

Modèles à données manquantes

Cas particulier de modèles où la densité à simuler s'écrit

$$f(x) = \int_Z \tilde{f}(x, z) dz$$

La variable Z est alors appelée **donnée manquante**

Principe de complétion

Idée

Simuler \tilde{f} produit des simulations suivant f

Si

$$(X, Z) \sim \tilde{f}(x, z),$$

marginalement

$$X \sim f(x)$$

Data Augmentation

Partant de $x^{(t)}$,

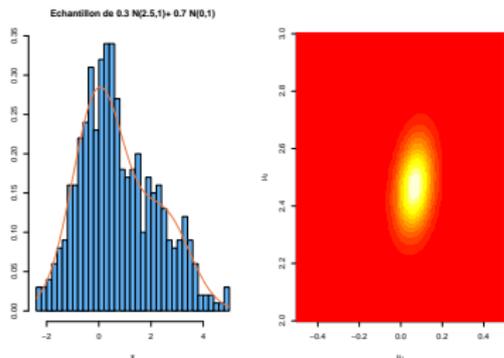
- 1 Simuler $Z^{(t+1)} \sim \tilde{f}_{Z|X}(z|x^{(t)})$;
- 2 Simuler $X^{(t+1)} \sim \tilde{f}_{X|Z}(x|z^{(t+1)})$.

Exemple (Mélange de lois)

Soit à simuler sur \mathbb{R}^2 la loi de densité proportionnelle à

$$e^{-\mu_1^2 - \mu_2^2} \times \prod_{i=1}^{100} \left\{ 0.3 e^{-(x_i - \mu_1)^2 / 2} + 0.7 e^{-(x_i - \mu_2)^2 / 2} \right\}$$

où les x_i sont donnés et (μ_1, μ_2) représente les coordonnées de la variable aléatoire

Histogramme des x_i et surface de la loi de (μ_1, μ_2) associée**Complétion (1)**

Remplacer chaque somme dans la densité par une intégrale:

$$0.3 e^{-(x_i - \mu_1)^2/2} + 0.7 e^{-(x_i - \mu_2)^2/2} = \int \left(\mathbb{I}_{[0, 0.3 e^{-(x_i - \mu_1)^2/2}]}(u_i) + \mathbb{I}_{[0.3 e^{-(x_i - \mu_1)^2/2}, 0.3 e^{-(x_i - \mu_1)^2/2} + 0.7 e^{-(x_i - \mu_2)^2/2}]}(u_i) \right) du_i$$

et simuler $((\mu_1, \mu_2), (U_1, \dots, U_n)) = (X, Z)$ par Data Augmentation**Complétion (2)**Remplacer les U_i par les ξ_i , où

$$\xi_i = \begin{cases} 1 & \text{si } U_i \leq 0.3 e^{-(x_i - \mu_1)^2/2}, \\ 2 & \text{sinon} \end{cases}$$

Alors

$$\begin{aligned} \Pr(\xi_i = 1 | \mu_1, \mu_2) &= \frac{0.3 e^{-(x_i - \mu_1)^2/2}}{0.3 e^{-(x_i - \mu_1)^2/2} + 0.7 e^{-(x_i - \mu_2)^2/2}} \\ &= 1 - \Pr(\xi_i = 2 | \mu_1, \mu_2) \end{aligned}$$

Conditionnement (1)La loi conditionnelle de $Z = (\xi_1, \dots, \xi_n)$ sachant $X = (\mu_1, \mu_2)$ est donnée par

$$\begin{aligned} \Pr(\xi_i = 1 | \mu_1, \mu_2) &= \frac{0.3 e^{-(x_i - \mu_1)^2/2}}{0.3 e^{-(x_i - \mu_1)^2/2} + 0.7 e^{-(x_i - \mu_2)^2/2}} \\ &= 1 - \Pr(\xi_i = 2 | \mu_1, \mu_2) \end{aligned}$$

Conditionnement (2)

La loi conditionnelle de $X = (\mu_1, \mu_2)$ sachant $Z = (\xi_1, \dots, \xi_n)$ est donnée par

$$\begin{aligned}
 (\mu_1, \mu_2) | Z &\sim e^{-\mu_1^2 - \mu_2^2} \times \prod_{\{i: \xi_i=1\}} e^{-(x_i - \mu_1)^2 / 2} \times \prod_{\{i: \xi_i=2\}} e^{-(x_i - \mu_2)^2 / 2} \\
 &\propto \exp \left\{ -(n_1 + 2) \left(\mu_1 - \frac{n_1 \hat{\mu}_1}{n_1 + 2} \right)^2 / 2 \right\} \\
 &\quad \times \exp \left\{ -(n_2 + 2) \left(\mu_2 - \frac{n_2 \hat{\mu}_2}{n_2 + 2} \right)^2 / 2 \right\}
 \end{aligned}$$

où n_j est le nombre de ξ_i égaux à j et $n_j \hat{\mu}_j$ est la somme des x_i associés à ces ξ_i égaux à j

[Easy!]