

Introduction aux méthodes de Monte Carlo par dynamique Hamiltonienne

Shmuel RAKOTONIRINA-RICQUEBOURG, Amaury DURAND

8 décembre 2017

Plan

- 1 Introduction
- 2 Dynamique hamiltonienne
- 3 Hamiltonian Monte-Carlo
- 4 Simulations

- 1 Introduction
 - Algorithmes MCMC
 - Algorithme de Metropolis (Random Walk Metropolis)
- 2 Dynamique hamiltonienne
- 3 Hamiltonian Monte-Carlo
- 4 Simulations

Principe des MCMC

- Objectif : pour π à densité h_π simuler π ou approcher πf
- Idée : trouver une chaîne de Markov X admettant π comme loi invariante et convergeant vers π

Théorème (Théorème ergodique)

Soit $(X_k)_{k \in \mathbb{N}}$ une chaîne de Markov de noyau P sur $(\mathbb{X}, \mathcal{X})$ admettant une unique loi invariante π . Alors pour tout $f \in \mathbb{F}_+(\mathbb{X}, \mathcal{X}) \cup \mathbb{F}_b(\mathbb{X}, \mathcal{X})$ et pour π -presque tout $x \in \mathbb{X}$,

$$\frac{1}{n} \sum_{k=0}^{n-1} f(X_k) \xrightarrow[n \rightarrow +\infty]{\mathbb{P}_x \text{- p.s.}} \pi f$$

Algorithme Random-walk Metropolis

Algorithme 1.1 : Random Walk Metropolis

Données : h_π proportionnel à la densité cible, Q loi simulable

$X_0 \leftarrow x \in \mathbb{X}$ arbitraire;

$(U_k)_{k \in \mathbb{N}} \stackrel{\text{iid}}{\sim} Q$;

répéter

$Y_{k+1} \leftarrow X_k + U_{k+1}$; // Proposer un mouvement

$\alpha_{k+1} \leftarrow \alpha(X_k, Y_{k+1})$ où $\alpha(x, y) = 1 \wedge \frac{h_\pi(y)}{h_\pi(x)}$;

$X_{k+1} \leftarrow \begin{cases} Y_{k+1} & \text{avec probabilité } \alpha_{k+1} \\ X_k & \text{with probability } 1 - \alpha_{k+1} \end{cases}$; // Accepter ou
rejeter le mouvement

jusqu'à *une condition d'arrêt*;

retourner $(X_k)_k$

- 1 Introduction
- 2 **Dynamique hamiltonienne**
 - Définition
 - Propriétés
 - Discrétisation
- 3 Hamiltonian Monte-Carlo
- 4 Simulations

Dynamique hamiltonienne

Définition

(Dynamique hamiltonienne) Soit $H : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$
 $(x, p) \mapsto H(x, p)$. Deux
fonctions de position $x : \mathbb{R}_+ \rightarrow \mathbb{R}^d$ et de quantité de mouvement
 $p : \mathbb{R}_+ \rightarrow \mathbb{R}^d$ sont dites solutions du hamiltonien H (ou suivant la
dynamique hamiltonienne de H) si

$$x'(t) = \frac{\partial H}{\partial p}(x(t), p(t))$$

$$p'(t) = -\frac{\partial H}{\partial x}(x(t), p(t))$$

Pour $T > 0$, H peut être associée à la densité (sur \mathbb{R}^{2d})

$$h(z) \propto \exp\left(-\frac{H(z)}{T}\right)$$

Hamiltonien pour l'algorithme HMC

Pour avoir $X \sim \pi$, on cherche à simuler $(X, P) \sim \tilde{\pi} = \pi \otimes \nu$ pour une loi ν choisie. Dans toute la suite, on fera les hypothèses suivantes :

Hypothèses (H)

- π et ν sont à densité h_π et h_ν strictement positives sur \mathbb{R}^d
- $\exists k \geq 1$, $\ln(h_\pi)$ et $\ln(h_\nu)$ sont de classe \mathcal{C}^k sur \mathbb{R}^d
- h_ν est paire
- $H : (x, p) \mapsto U(x) + K(p)$ avec $U = -T \ln(h_\pi)$, $K = -T \ln(h_\nu)$

Ainsi, la densité $h \propto e^{H/T}$ est la densité jointe $h = h_\pi \otimes h_\nu = h_{\tilde{\pi}}$.

Flot de l'équation différentielle

En notant $z = (x, p)$, les équations $x' = \frac{\partial H}{\partial p}$, $p' = -\frac{\partial H}{\partial x}$ se réécrivent $z' = F(z)$ où $F = J\nabla H$ et $J = \begin{bmatrix} 0_d & I_d \\ -I_d & 0_d \end{bmatrix}$.

Proposition (conservation du hamiltonien)

Le hamiltonien est conservé le long des trajectoires : si $z' = F(z)$, alors $H \circ z$ est constant.

Définition (flot hamiltonien)

Pour $t \in \mathbb{R}$, on définit le flot ϕ_t de sorte que pour tout $z_0 \in \mathbb{R}^{2d}$, $t \mapsto \phi_t(z_0)$ soit l'unique solution du hamiltonien avec condition initiale $z(0) = z_0$.

Proposition (conservation du volume)

La solution du hamiltonien conserve le volume : $\det \left(\frac{d\phi_t}{dz}(z) \right) = 1$.

Réversibilité du flot

Lemme (réversibilité du temps)

Soit $z = (x, p)$ une solution du hamiltonien H . On définit $\bar{z} = (\bar{x}, \bar{p})$ par $\bar{z}(t) = (x(-t), -p(-t))$. Alors

- ① \bar{z} est solution du hamiltonien H (avec d'autres conditions initiales).
- ② $\forall t \in \mathbb{R}_+, \phi_t(\bar{z}(-t)) = \bar{z}(0)$

Proposition (réversibilité du flot)

ϕ_t est un \mathcal{C}^k -difféomorphisme d'inverse
 $\phi_t^{-1} : (x, p) \mapsto (\phi_t^{(1)}(x, -p), -\phi_t^{(2)}(x, -p)).$

Preuve de la réversibilité du temps.

On définit la symétrie $s(x, p) = (x, -p)$. Par définition, $\bar{z}(t) = s \circ z(-t)$ et par hypothèse de parité de h_ν , $H = H \circ s$.

Notons $S \doteq \frac{ds}{dz}(z) = \begin{bmatrix} I_d & 0_d \\ 0_d & -I_d \end{bmatrix}$ (et ce pour tout z). Ainsi,

$\nabla H = S \nabla H \circ s$. On remarque que $SJ = -JS$ et $s^{-1} = s$.

$$\bar{z}'(t) = -S z'(-t) = -SJ \nabla H(z(-t)) = JS \nabla H(z(-t))$$

donc

$$\bar{z}'(t) = J \nabla H(s^{-1}(z(-t))) = J \nabla H(\bar{z}(t))$$



Preuve de la réversibilité du flot.

Il suffit de prouver la formule de l'inverse. On pose

$\bar{\phi}_t(x, p) = (\phi_t^{(1)}(x, -p), -\phi_t^{(2)}(x, -p))$. On fixe $z_0 = (x_0, p_0)$ et on note z la solution du hamiltonien avec $z(0) = z_0$.

- D'une part,

$$\bar{\phi}_t(\phi_t(x_0, p_0)) = \bar{\phi}_t(z(t)) = (\phi_t^{(1)}(\bar{z}(-t)), -\phi_t^{(2)}(\bar{z}(-t))) = (x_0, p_0)$$

car $\bar{z}(0) = (x_0, -p_0)$.

- D'autre part,

$$\bar{\phi}_t(x_0, -p_0) = (\phi_t^{(1)}(x_0, p_0), -\phi_t^{(2)}(x_0, p_0)) = (x(t), -p(t)) = \bar{z}(-t)$$

donc

$$\phi_t(\bar{\phi}_t(x_0, -p_0)) = \phi_t(\bar{z}(-t)) = \bar{z}(0) = (x_0, -p_0).$$



Algorithme du leapfrog

Variante de la méthode d'Euler : on discrétise

$x' = \nabla K(p)$, $p' = -\nabla U(x)$ en

- 1 $p_{t+\epsilon/2} = p_t - \frac{\epsilon}{2} \nabla U(x_t)$ (demi-pas en p).
- 2 $x_{t+\epsilon} = p_t + \epsilon \nabla K(p_{t+\epsilon/2})$ (pas en x).
- 3 $p_{t+\epsilon} = p_{t+\epsilon/2} - \frac{\epsilon}{2} \nabla U(x_{t+\epsilon})$ (demi-pas en p).

Algorithme 2.1 : Discrétisation de l'évolution par saute-mouton (*leapfrog*)

Données : pas ϵ , nombre de pas L , état initial (x_0, p_0)

pour $k \in \llbracket 0, L-1 \rrbracket$ **faire** // Saute-mouton

[$x_{k+1} \leftarrow x_k + \epsilon \nabla K(p_k - \frac{\epsilon}{2} \nabla U(x_k));$
 $p_{k+1} \leftarrow p_k - \frac{\epsilon}{2} \nabla U(x_k) - \frac{\epsilon}{2} \nabla U(x_{k+1});$

retourner (x_L, p_L)

Flot approché

Données : pas ϵ , nombre de pas L , état initial (x_0, p_0)

pour $k \in \llbracket 0, L-1 \rrbracket$ **faire** // Saute-mouton

$x_{k+1} \leftarrow x_k + \epsilon \nabla K \left(p_k - \frac{\epsilon}{2} \nabla U(x_k) \right);$
 $p_{k+1} \leftarrow p_k - \frac{\epsilon}{2} \nabla U(x_k) - \frac{\epsilon}{2} \nabla U(x_{k+1});$

retourner (x_L, p_L)

Définition (flot approché)

On fixe $\epsilon > 0$. Pour $L \in \mathbb{N}^*$, on définit le flot approché du hamiltonien par L itérations de l'algorithme leapfrog par $\hat{\phi}_L = \hat{\phi}^L$ où $\hat{\phi}$ est défini par

$$\hat{\phi}^{(1)} : (x, p) \mapsto x + \epsilon \nabla K \left(p - \frac{\epsilon}{2} \nabla U(x) \right)$$

$$\hat{\phi}^{(2)} : (x, p) \mapsto p - \frac{\epsilon}{2} \nabla U(x) - \frac{\epsilon}{2} \nabla U(\hat{\phi}^{(1)}(x, p))$$

Propriétés du flot approché

Proposition (conservation du volume)

La solution approchée du hamiltonien par le leapfrog conserve le volume :
 $\det \left(\frac{d\hat{\phi}}{dz}(z) \right) = 1.$

Proposition (réversibilité du flot approché)

$\hat{\phi}$ est inversible d'inverse

$$\phi^{-1} = s \circ \hat{\phi} \circ s : (x, p) \mapsto (\hat{\phi}^{(1)}(x, -p), -\hat{\phi}^{(2)}(x, -p)).$$

- 1 Introduction
- 2 Dynamique hamiltonienne
- 3 Hamiltonian Monte-Carlo**
 - Cas idéal
 - Cas réel
- 4 Simulations

Algorithme 3.1 : Hamiltonian Monte-Carlo, cas idéal
$$X_0 \leftarrow x \in \mathbb{X} \text{ arbitraire;}$$

répéter

$$\tilde{Z}_k \leftarrow (X_k, \tilde{P}_k) ;$$
$$Z_{k+1} = (X_{k+1}, P_{k+1}) \leftarrow \phi_t(\tilde{Z}_k); // \text{ Suivre la dynamique}$$

jusqu'à *une condition d'arrêt;*

retourner $(X_k)_k$

Algorithme HMC réel

Algorithme 3.2 : Hamiltonian Monte-Carlo

Données : h_π proportionnel à la densité cible, ϵ pas du saute-mouton, L
nombre de pas du saute-mouton

$X_0 \leftarrow x \in \mathbb{X}$ arbitraire;

répéter

```
   $P_k \sim \mathcal{N}(0,1)$ ; // Tirer la quantité de mouvement  
   $(X_{prop}, P_{prop}) \leftarrow \text{leapfrog}(X_k, P_k)$ ; // Proposer un mouvement  
   $U_k \leftarrow U(X_k)$ ;  $K_k \leftarrow \|P_k\|^2/2$ ;  
   $U_{prop} \leftarrow U(X_{prop})$ ;  $K_{prop} \leftarrow \|P_{prop}\|^2/2$ ;  
  si  $\mathcal{U}([0,1]) < \exp(U_k - U_{prop} + K_k - K_{prop})$  alors  
    |  $X_{k+1} \leftarrow X_{prop}$  ; // Accepter  
  sinon  
    |  $X_{k+1} \leftarrow X_k$  ; // Rejeter
```

jusqu'à *une condition d'arrêt*;

retourner $(X_k)_k$

- 1 Introduction
- 2 Dynamique hamiltonienne
- 3 Hamiltonian Monte-Carlo
- 4 Simulations**