ENSIIE 2018-2019
2A UE Méthodes de Simulation

Feuille 3 de Travaux pratiques

Exercice 1. (*Simulation du modèle d'Ising*). Soit $N \in \mathbb{N}^*$. On considère le maillage $\Lambda = \{0, \ldots, N-1\}^2$ muni de la relation de voisinage verticale et horizontale (mais pas diagonale) $x \sim y$. A chaque spin $x = (i,j) \in \Lambda$ on associe un état $s(i,j) \in \{-1, 1\}$, de sorte que l'espace des états du système est $E = \{s : \Lambda \mapsto \{-1,1\}\} = \{-1,1\}^\Lambda$.

On note $S = (s(x))_{x\in\Lambda}$ une configuration possible de spins et on considère la mesure de Gibbs sur E qui, à toute configuration S associe la probabilité

$$\pi_\beta(S) = \frac{1}{Z(\beta)} \exp\Big(-\beta H(S)\Big) \qquad \text{avec} \quad H(S) = -\sum_{x,y\in E: x\sim y} s(x)s(y).$$

Dans ce modèle, les spins interagissent avec leurs voisins et l'énergie associée à la configuration S est $H(S)$. La quantité $T = 1/\beta$ est la température: lorsque la température T est élevée, les fluctuations thermiques dominent rendant le système désordonné tandis qu'en basse température (lorsque T est proche de 0), le système privilégie les configurations de basse énergie tendant à aligner les spins.

On note S_x la configuration obtenue en changeant le signe du spin en $x \in \Lambda$ de la configuration S. La matrice de référence Q décrivant le passage d'une configuration S à une configuration voisine S_x est définie par

$$\forall x \in \Lambda, \quad Q(S, S_x) = \frac{1}{\text{card}(\Lambda)}.$$

Ainsi, on passe d'une configuration S à une configuration S_x en choisissant un site x au hasard selon une loi uniforme sur Λ et en inversant le signe de son spin dans S. La variation d'énergie correspondante est

$$\Delta H(S, S_x) := H(S_x) - H(S) = 2s(x)\sum_{y\sim x} s(y).$$

Comme Q est symétrique, on a

$$h(S, S_x) = \frac{\pi_\beta(S_x)Q(S_x, S)}{\pi_\beta(S)Q(S, S_x)} = \exp\big(-\beta\, \Delta H(S, S_x)\big).$$

Cela nous permet de considérer l'algorithme suivant de Hastings-Metropolis pour simuler un modèle d'Ising.

- Choisir une configuration initiale $S \in E$.
- Répéter un grand nombre de fois:
  - Simuler indépendemment $V = x \sim \mathcal{U}(\Lambda)$ et $U \sim \mathcal{U}(]0,1[)$.
  - Si $U \leq \exp\big(-\beta\, \Delta H(S, S_x)\big)$, on remplace S par S_x, sinon, on laisse S inchangé.
- Returner S.

1. On pose $\beta = 0.1$ et $N = 60$. Implémentez l'algorithme précédent pour une configuration initiale S fixée. Implémentez trois fois votre algorithme pour la même configuration initiale S et représentez graphiquement les configurations finales dans trois fenêtres graphiques différentes.
2. On pose $\beta = 10$ et $N = 60$. Implémentez l'algorithme précédent pour une configuration initiale S fixée. Implémentez trois fois votre algorithme pour la même configuration initiale S et représentez graphiquement les configurations finales dans trois fenêtres graphiques différentes.
3. Interprétez les graphiques trouvés dans les deux questions précédentes.
4. Changez de configuration initiale dans les questions 1. et 2. et interprétez les graphiques obtenus.

Exercice 2. (*Voyageur de commerce*). On considère un voyageur de commerce qui doit visiter chacun des N villes en rotation (en terminant par la première ville visitée). Les villes sont numérotées de 1 à N et chaque ville i a pour coordonnées (x_i, y_i). L'ensemble S_N des tournées possibles est l'ensemble des permutations dans $\{1, \ldots, N\}$ qui est de cardinal $N!$. Une permutation $\sigma = (\sigma(1), \ldots, \sigma(N)) \in S_N$ donne l'ordre dans lequel les villes sont parcourues.

Le but du voyageur de commerce est de trouver l'itinéraire le plus court pour faire le tour de l'ensemble des N villes. Cela revient à trouver la permutation $\varpi = (\varpi(1), \ldots, \varpi(N)) \in S_N$ qui minimise la fonction

$$\begin{aligned}
\sigma \in S_N \mapsto H(\sigma) &= \sum_{i=1}^{N} \mathrm{dist}\Big(\big(x_{\sigma(i)}, y_{\sigma(i)}\big), \big(x_{\sigma(i+1)}, y_{\sigma(i+1)}\big)\Big) \\
&= \sum_{i=1}^{N} \sqrt{\big(x_{\sigma(i)} - x_{\sigma(i+1)}\big)^2 + \big(y_{\sigma(i)} - y_{\sigma(i+1)}\big)^2},
\end{aligned}$$

c'est-à-dire, tel que $H(\varpi) = \min_{\sigma \in S_N} H(\sigma)$, avec la convention que $\big(x_{\sigma(N+1)}, y_{\sigma(N+1)}\big) = \big(x_{\sigma(1)}, y_{\sigma(1)}\big)$.

On considère la relation de voisinage suivante dans S_N: $\varpi \sim \sigma$ s'il existe $i < k$ tels que

$$\varpi = (\sigma(1), \ldots, \sigma(i-1), \sigma(k), \sigma(i+1), \ldots, \sigma(k-1), \sigma(i), \sigma(k+1), \ldots, \sigma(N)).$$

Ainsi, partant d'une permutation σ, on génère une permutation voisine en choisissant uniformément un couple (i, j) dans l'ensemble $\{1, \ldots, N\} \times \{1, \ldots, N\}$ de cardinal N^2 et: si $i = k$ on laisse σ inchangé, sinon, on échange les positions des villes i et k. La matrice de voisinage est donc

$$Q(\sigma, \varpi) = \begin{cases} 1/N & \text{si } \sigma = \varpi \\ 2/N^2 & \text{si } \sigma \sim \varpi \text{ et } \sigma \neq \varpi \\ 0 & \text{si } \sigma \not\sim \varpi. \end{cases}$$

On en déduit l'algorithme du récuit simulé suivant. On choisira $\beta_n = c \ln(n+1)$, $c > 0$.

- Choisir une permutation initiale $\sigma_0 \in S_N$.
- Répétez un grand nombre de fois
  - Simuler indépendemment $\varpi \sim Q(\sigma_n, \cdot)$ et $U \sim \mathcal{U}(]0, 1[)$.
  - Si $U \leq \exp(-\beta_n(H(\varpi) - H(\sigma_n)))$, on pose $\sigma_{n+1} = \varpi$, sinon, on pose $\sigma_{n+1} = \sigma_n$.
- Returner σ_{n+1}.

1. On pose $N = 50$ et on choisit les coordonnées des villes en simulant un échantillon de taille N de la loi $\mathcal{N}(0; c^2 \times I_2)$, avec $c = 10$. Implémentez l'algorithme précédent.