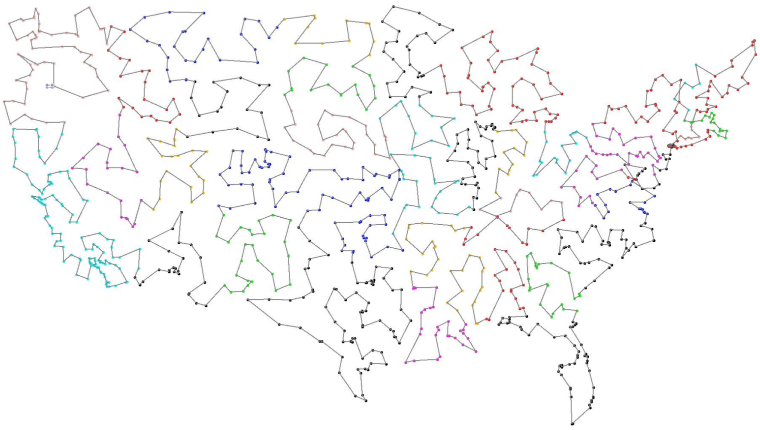
Un exemple d'optimisation par chaîne de Markov : Le voyageur de commerce



Source : K. Helsgaun. Solving the Clustered Traveling Salesman Problem Using the Lin-Kernighan-Helsgaun. Roskilde University, 2014.

Le contexte général est le suivant : soit E un (grand) ensemble fini, et une fonction $V : E \to \mathbb{R}$. On cherche à trouver $x \in E$ tel que $V(x)$ est le plus petit possible, l'ensemble E est tellement grand qu'une recherche exhaustive est exclue.

Le principe de l'optimisation par chaîne de Markov, ou méthode MCMC (pour *Monte Carlo Markov Chain*) est de parcourir de façon aléatoire mais intelligente l'ensemble E pour chercher à minimiser V.

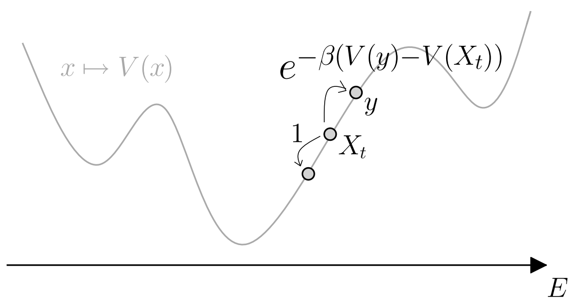
1 L'algorithme de Metropolis-Hastings

On suppose que E est muni d'une structure de graphe : certains points x, y sont reliés par des *arêtes*, on note alors $x \sim y$. On suppose que le graphe est connexe. Le principe de l'algorithme de Metropolis-Hastings est le suivant : on parcourt le graphe E en favorisant les arêtes qui font diminuer V, de temps en temps on s'autorise à augmenter V pour ne pas rester bloqué dans un minimum local.

Voici l'algorithme :

Paramètres :
$X_0 \in E$: valeur initiale
$\beta \in [0, +\infty)$
$T \in \mathbb{N}$: nombre d'itérations

Pour $t = 0$ à $T - 1$
  $y =$ v.a. uniforme parmi les voisins de X_t
  Si $V(y) < V(X_t)$ alors
    $X_{t+1} = y$
  Sinon si $V(y) \geq V(X_t)$ alors
    $X_{t+1} = y$ avec proba $e^{-\beta(V(y)-V(X_t))}$
    $X_{t+1} = X_t$ avec proba $1 - e^{-\beta(V(y)-V(X_t))}$
Renvoyer X_T



Question 1. Que fait l'algorithme pour $\beta = 0$? pour $\beta = +\infty$? Ce paramètre est parfois appelé "inverse de la température".

On peut démontrer la chose suivante (c'est par exemple la combinaison des Théorème 6.2 et 5.5 dans [1])

Théorème 1 *Si le graphe associé à E est connexe, alors pour tout $\beta > 0$ on a*

$$\mathbb{P}(X_t = x) \stackrel{t\to+\infty}{\to} \frac{1}{Z_\beta} e^{-\beta V(x)}, \qquad \textit{avec } Z_\beta = \sum_{z\in E} e^{-\beta V(z)}.$$

Question 2. En quoi ce Théorème assure-t-il que l'algorithme de Metropolis-Hastings remplit l'objectif de minimisation de V ? Quelles sont les limites de cet algorithme ?

Question 3. Implémenter l'algorithme pour

- $E = \{1, 2, \ldots, k\}$, avec $k = 40$,
- Graphe : chaque i est relié à $i-1$ et $i+1$,
- $V(x) = \cos(4\pi x/k) - \sqrt{4\pi x/k}$,
- différentes valeurs de β (on pourra prendre pour commencer $T = 2000$, $\beta = 0.5$ puis $\beta = 2$).

Tracer à chaque fois quelques trajectoires de $t \mapsto V(X_t)$ et comparer avec le graphe de la fonction V.

2 Application au problème du voyageur de commerce

On cherche maintenant à appliquer l'algorithme de Metropolis-Hastings au problème suivant. Soient $(X_1, Y_1), \ldots, (X_n, Y_n)$ les coordonnées de n villes dans le plan. On cherche le chemin le plus court passant par toutes ces villes, c'est-à-dire la permutation $\sigma_\star$ dans l'ensemble $\mathfrak{S}_n$ des permutations de n éléments telle que

$$\sigma_\star = \text{argmax}_\sigma V(\sigma) := \text{argmax}_\sigma \sum_{i=1}^{n-1} \parallel (X_{\sigma(i+1)}, Y_{\sigma(i+1)}) - (X_{\sigma(i)}, Y_{\sigma(i)}) \parallel$$

On met la structure suivante sur $\mathfrak{S}_n$: $\sigma \sim \sigma'$ si l'on peut passer de σ à σ' en permutant deux éléments. Ainsi pour $n = 4$, on peut passer de 1234 à 2134,1324, 1243, 1432, 3214, 4231. De façon générale, σ a toujours $\binom{n}{2}$ permutations "voisines".

Question 4. On va représenter les coordonnées par une matrice `Coordonnees` de taille $n \times 2$. Écrire une fonction `EchangerDeuxVilles(Coordonnees,i,j)` qui permute deux indices i et j dans le tableau des villes, et une fonction `Longueur(Coordonnees)` qui calcule la longueur d'un chemin.

Question 5. Implémenter l'algorithme de Metropolis-Hastings pour le problème du voyageur de commerce, sur le fichier de villes `PaysMystere.xls` téléchargeable sur le `moodle` (rappel : la commande `xlsread('PaysMystere.xls')` permet de charger un fichier Excel).

Quelques conseils :

- Testez toutes vos fonctions et vos premiers codes sur des fichiers de villes simples avant de vous attaquer à `PaysMystere` !
- Un choix de paramètre possible au début est $T = 50000$ itération, $\beta = 2$.

Remarque. Évaluer l'efficacité d'un algorithme pour le voyageur de commerce est difficile dans la mesure où l'on ne connaît pas la longueur $V(\sigma_\star)$ du chemin le plus court. Cependant, si les (X_i, Y_i) sont distribuées uniformément sur le carré $[0, 1]^2$, on peut montrer qu'il existe c telle que $V(\sigma_\star) \sim c\sqrt{n}$, des travaux récents [2] suggèrent que $c \approx 0.712$. Vous pouvez comparer cette borne au résultat donné par votre algorithme.

Références

[1] T.Bodineau. *Modélisation de phénomènes aléatoires.* Cours de l'École Polytechnique (2015).

[2] S.Steinerberger. New bounds for the traveling salesman constant. *Advances in Applied Probability* vol.47 (2015) n.1, p.27-36.