Algorithme d'Hasting Metropolis - ProblÃÍme du voyageur de commerce

PEROTTINO Tony, VAILLANT Corentin,
LE BER Tom

8 mai 2025

Table des matières

PrÃľambule

Ce rapport s'inscrit dans le cadre de la matiÃĺre "Projet de MathÃľmatiques" de l'universitÃľ de Toulouse (Paul Sabatier).
L'objectif de ce rapport est de prÃľsenter et de prouver l'algorithme d'Hastings-Metropolis. ÃĂ la fois d'expliquer en dÃľtail son fonctionnement, mais aussi de comprendre l'intÃľrÃłt qu'il prÃľsente dans les sciences et en quoi il prend sa valeur.
Pour illustrer l'algorithme, nous nous intÃľresserons au problÃĺme du voyageur de commerce et Ãľtudierons sa complexitÃľ.

1 Introduction

Afin de comprendre en profondeur ce sujet, il est nÃľcessaire de s'en faire une idÃľe globale, mÃłme naÃŕve, pour pouvoir suivre convenablement la ligne directrice de notre discours.

1.1 Informations gÃľnÃľrales et description historique

L'algorithme d'Hastings-Metropolis (notÃľ "HM" dans ce rapport) consiste en une mÃľthode d'Ãľchantillonnage stochastique permettant, Ãă partir d'une distribution de probabilitÃľ donnÃľe, de pouvoir dÃľcrire son comportement et d'obtenir des statistiques dessus. Cet algorithme prend sa valeur quand la distribution est difficile Ãă analyser (systÃĺmes multidimensionnels, par exemple) et permet de rÃľsoudre de nouveaux problÃĺmes avec une complexitÃľ temporelle dÃľcente.
Cette mÃľthode est marquante, car elle a ÃľtÃľ conceptualisÃľe tÃťt dans l'histoire de l'informatique et a mis des dÃľcennies avant d'Ãłtre prouvÃľe et expliquÃľe entiÃĺrement.

Historiquement, c'est en 1949 que l'Ãľcriture de l'algorithme a ÃľtÃľ publiÃľe dans un article de Nicholas Metropolis et StanisÅĆaw Ulam. La paternitÃľ de l'algorithme est soumise Ãă dÃľbat, car l'algorithme s'inscrit sous le nom de son chef de projet (Metropolis), alors que l'Ãľquipe composÃľe de Nicholas Metropolis, Arianna et Marshall Rosenbluth, Augusta et Edward Teller a contribuÃľ Ãă cette mÃľthode. Ils Ãľtudiaient alors plus particuliÃĺrement le cas de la distribution de Boltzmann, une des distributions les plus utilisÃľes en physique statistique, dans des travaux datant de 1953.
Cela illustre une dynamique frÃľquente dans les sciences, oÃź le mÃľrite est disproportionnellement attribuÃľ Ãă une personne alors qu'il s'agit des efforts de toute une Ãľquipe. En particulier, Arianna Rosenbluth Ãľtait considÃľrÃľe comme brillante par le monde scientifique.

En 1970, W. K. Hastings (1930-2016) a Ãľtendu l'algorithme au cas d'une distribution quelconque, et c'est cette version gÃľnÃľralisÃľe qui est connue sous le nom d'algorithme de Metropolis-Hastings. Cette extension a eu de nombreuses applications dans divers domaines scientifiques, comme en statistique bayÃľsienne (espaces complexes multidimensionnels), en biologie computationnelle (analyse des sÃľquences gÃľnÃľtiques), en Ãľconomie et en finance (modÃĺles stochastiques MCMC en gÃľnÃľral), etc.

Quant Ãă lui, le problÃĺme du voyageur de commerce (dit "TSP", comme Travelling Salesman Problem en anglais) est un problÃĺme classique et bien connu pour Ãłtre un problÃĺme NP-difficile, ce qui signifie qu'il n'existe actuellement aucune mÃľthode connue capable de le rÃľsoudre de maniÃĺre optimale en temps polynomial pour toutes les tailles d'instances, et qu'il est au moins aussi difficile que les problÃĺmes de NP. L'origine du problÃĺme est assez incertaine : il a ÃľtÃľ formulÃľ pour la premiÃĺre fois vers 1850 dans un manuel d'un commerÃğant voyageant en Suisse et en Allemagne. Ce n'est que dans les annÃľes 1930 que le problÃĺme fut ÃľnoncÃľ d'abord comme un casse-tÃłte (par William Rowan et Thomas Kirkman), puis ÃľtudiÃľ (par, entre autres, Thomas Kirkman, Jillian Beardwood, J. H. Halton et John Hammersley).
Le problÃĺme consiste Ãă dÃľterminer le chemin le plus court passant par tous les points d'un graphe une seule fois chacun, en terminant par le point de dÃľpart (recherche d'un cycle hamiltonien le plus court). Les distances peuvent Ãłtre dites symÃľtriques ou asymÃľtriques, c'est-Ãă-dire que la distance entre eux varie en fonction de la direction du dÃľplacement. On peut illustrer ce problÃĺme grÃćce Ãă un voyageur de commerce devant vendre ses produits dans chacune des villes en un minimum de temps. Ce problÃĺme peut donc Ãłtre naturellement reprÃľsentÃľ par un graphe.

1.2 Pourquoi HM permet-il de rÃľsoudre le problÃĺme du voyageur de commerce ?

Le principe mathÃľmatique derriÃĺre HM repose sur la construction dynamique d'une chaÃőne de Markov, qui n'est pas connue au prÃľalable mais se dÃľveloppe au fil des itÃľrations. ÃĂ mesure que l'algorithme progresse, le comportement de cette chaÃőne converge vers la distribution cible, permettant d'obtenir un Ãľchantillon fiable de solutions par rapport aux Ãľtats de la distribution.
Quant au problÃĺme du voyageur de commerce, il peut Ãłtre reprÃľsentÃľ par un graphe orientÃľ ayant un sommet pour chaque ville et une arÃłte pour chaque temps de trajet. Nous verrons par la suite qu'une chaÃőne de Markov peut Ãłtre associÃľe Ãă un graphe orientÃľ, c'est-Ãă-dire que le problÃĺme du voyageur de commerce est rÃľsoluble par HM.
Par rÃľsoluble, il est important de prÃľciser qu'il s'agit d'une rÃľsolution appoximative parce que le TSP est NP-difficile comme mentionnÃľ prÃľcÃľdemment. En consÃľquence, l'objectif de HM n'est pas d'apporter la rÃľponse exacte au problÃĺme, mais une approximation fiable de la solution, ce qui peut

sembler contre-intuitif. Cette approximation est la caractÃľristique principale des mÃľthodes MCMC que nous aborderons dans leur partie dÃľdiÃľe.

Il convient de souligner que, dans le cas du TSP, c'est Ãă la version d'optimisation que l'on s'intÃľresse, par opposition au problÃĺme dÃľcisionnel, qui, lui, est NP-complet. La diffÃľrence entre ces deux formulations repose sur leur objectif : dans le TSP dÃľcisionnel, on pose une question du type "Existe-t-il un chemin de coÃżt infÃľrieur ou Ãľgal Ãă une valeur donnÃľe ?", Ãă laquelle il s'agit de rÃľpondre par oui ou non. Alors que le TSP d'optimisation cherche Ãă partir de la situation initiale la meilleure solution : donc optimiser le chemin au fil des itÃľrations. C'est cette version du problÃĺme pour laquelle l'approche par Ãľchantillonnage stochastique est envisageable. PrÃľcisons toutefois que HM n'est habituellement pas la mÃľthode que l'on utilise pour le TSP en pratique. Usuellement, le nombre de ville Ãľtant grand (suffisemment pour que la solution par brute force ne soit pas trouvable) mais pas absurde (une centaine par exemple), on prÃľfÃĺre utiliser des versions dÃľterministes optimisÃľes comme l'algorithme de Held-Karp de complexitÃľ $O(2^n \times n^2)$, ou dans le cas de situation avec encore plus de villes, des mÃľthodes similaires Ãă HM plus adaptÃľes comme l'Ãľchantillonage de Gibbs.

En somme, il est important de comprendre que HM est un outil adaptÃľ Ãă la rÃľsolution de problÃĺmes dont l'espace des solutions est trop vaste pour Ãłtre explorÃľ exhaustivement dans un temps raisonnable. C'est donc par l'approximation de la solution optimale, Ãă travers des Ãľchantillons, que cette mÃľthode permet de trouver une solution satisfaisante. Cette nuance est essentielle, car elle met en lumiÃĺre la sophistication de HM en tant qu'outil d'approximation pour des problÃĺmes complexes.

1.3 Objectif de ce rapport

L'objectif de ce rapport Ãľtant d'apporter une solution fiable au TSP grÃćce Ãă HM, nous prouverons alors le fonctionnement dde ce dernier grÃćce Ãă la thÃľorie mathÃľmatique des chaÃőnes de Markov. Nous en expliciterons les dÃľfinitions et propriÃľtÃľs fondamentales dans un premier temps, qui permettront de prouver l'algorithme d'HM dans un second temps. Nous prÃľsenterons alors Ãă la fin du rapport une implÃľmentation personnelle de l'algorithme.

2 ChaÃőnes de Markov Ãă Ãľtats finis

2.1 DÃľfinitions fondamentales

Temps Discret et Temps Continu

Soit T un ensemble d'indices numÃľrotÃľes reprÃľsentant le temps.

Il existe deux types de modÃľlisation temporelle :

(1) Un processus **Ãă temps discret** signifie que l'on considÃĹre les valeurs de la modÃľlisation comme espacÃľes rÃľguliÃĹrement dans le temps. On peut prendre des ensembles dÃľnombrables comme $T = \mathbb{N}$ ou $T = \mathbb{Z}$ oÃź chaque instant est distinct.

(2) Un processus **Ãă temps continu** signifie que T n'est pas dÃľnombrable. Il existe toujours un temps intermÃľdiaire entre deux indices de T. Cela signifie que le processus Ãľvolue en tout instant dans un continuum temporel, on peut prendre des ensembles non dÃľnombrables comme $T = \mathbb{R}_+$ oÃź chaque instant est continu.

Dans ce rapport nous nous interesserons uniquement au temps discret, pour pouvoir modÃľliser le problÃĹme du voyageur de commerce. De plus, cette distinction joue un rÃťle fondamental dans la classification et l'analyse des **processus stochastiques** et n'implique pas les mÃłmes thÃľorÃĹmes.

Processus Stochastique

Soit $(\Omega, \mathcal{F}, \mathbb{P})$ un espace de probabilitÃľ.
Soit T un ensemble d'indices discret ou continu (souvent $T = \mathbb{N}$ ou $\mathbb{R}_+$).

Un **processus stochastique** est une famille de variables alÃľatoires $\{X_t\}_{t \in T}$ dÃľfinies sur $(\Omega, \mathcal{F}, \mathbb{P})$ et Ãă valeurs dans un espace d'Ãľtats E (appelÃľ espace d'Ãľtats du processus).

Un processus stochastique permet de modÃľliser un systÃĹme Ãľvoluant de maniÃĹre alÃľatoire en fonction du temps.

DiffÃľrents types de processus peuvent Ãłtre ÃľtudiÃľs en fonction des propriÃľtÃľs de dÃľpendance temporelle et de la nature de l'espace d'Ãľtats E. Au cours de ce rapport, nous nous concentrerons l'un des principaux processus stochastiques Ãă temps discret et Ãă espace d'Ãľtats fini ; les **chaÃőnes de Markov**.

ChaÃőne de Markov

Soit E un ensemble fini ou dÃľnombrable.
Soit $(X_n)_{n\in\mathbb{N}}$ une suite de variables alÃľatoires Ãă valeurs dans l'espace d'Ãľtats E.

$(X_n)_{n\in\mathbb{N}}$ est appellÃľe **chaÃőne de Markov** si et seulement si :

(1) Sa loi de probabilitÃľ initiale X_0 est bien dÃľfinie.

(2) Elle respecte la **propriÃľtÃľ de Markov**, telle que :

$$\forall n \geq 0, \quad \exists x_1, x_2, \ldots, x_{n+1} \in E,$$

$$\mathbb{P}(X_{n+1} = x_{n+1} \mid X_1 = x_1, \ldots, X_n = x_n) = \mathbb{P}(X_{n+1} = x_{n+1} \mid X_n = x_n).$$

Une chaÃőne de Markov est un processus de Markov Ãă temps discret ou Ãă temps continu et Ãă espace d'Ãľtats discret. Un processus de Markov est un processus stochastique possÃľdant la propriÃľtÃľ de Markov : l'information utile pour la prÃľdiction du futur est entiÃĺrement contenue dans l'Ãľtat prÃľsent du processus et n'est pas dÃľpendante des ses Ãľtats antÃľrieurs.

Autrement dit, la loi de probabilitÃľ $\mathbb{P}$ rÃľgissant la transition de l'Ãľtat prÃľsent X_n vers l'Ãľtat futur X_{n+1} dÃľpend uniquement du dernier terme X_n, et reste totalement indÃľpendante des tous ses Ãľtats antÃľrieurs $\{X_0, X_1, \ldots, X_{n-1}\}$.

Cette propriÃľtÃľ, que l'on peut qualifier de Âń sans mÃľmoire Âż ou de propriÃľtÃľ de Markov, constitue la caractÃľristique fondamentale de ces processus stochastiques.

Chaine de Markov HomogÃĺne

Un chaine de Markov est dite homogÃĺne quand $\forall n \in \mathbb{N}$:

$$\mathbb{P}[X_{n+1} = j \mid X_n = i] = \mathbb{P}[X_1 = j \mid X_0 = i].$$

En somme, une chaÃőne de Markov homogÃĺne ne dÃľpend pas des Ãľtats prÃľcÃľdents (propriÃľtÃľ de Markov) et garantit que son comportement reste inchangÃľ au fil du temps, c'est-Ãă-dire que les probabilitÃľs de transition restent constantes quelque doit $t \in T$ (homogÃľnÃľitÃľ de la chaine de Markov). Dans ce rapport, toutes les chaÃőnes de Markov seront considÃľrÃľes comme homogÃĺnes.

2.2 PropriÃľtÃľs fondamentales

2.2.1 Matrice de Transition

Matrice de Transition

Soit $(X_n)_{n\in\mathbb{N}}$ une chaÃőne de Markov homogÃĺne Ãă valeurs dans un ensemble fini ou dÃľnombrable E.
Soit la famille de nombres rÃľels $(X_n)_{n\in\mathbb{N}}$ permettant de passer d'un Ãľtat Ãă un autre, dÃľfinie comme :

$$\forall n \geq 0,\ \forall i, j \in E, \quad P_{ij} = \mathbb{P}(X_{n+1} = j \mid X_n = i).$$

La **matrice de transition** P regroupe ces probabilitÃľs tel que chaque ÃľlÃľment P_{ij} reprÃľsente la probabilitÃľ de passer de l'Ãľtat i Ãă l'Ãľtat j en une seule Ãľtape. Lorsque l'ensemble des Ãľtats E est discret, par exemple $E = \{1, 2, \ldots, N\}$, la matrice s'exprime sous la forme suivante :

$$P = \begin{bmatrix} P_{11} & P_{12} & \cdots & P_{1N} \\ P_{21} & P_{22} & \cdots & P_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ P_{N1} & P_{N2} & \cdots & P_{NN} \end{bmatrix}$$

Matrice stochastique par lignes/colonnes

Une matrice **stochastique par lignes** (appelÃľe aussi **stochastique Ãă droite**) est une matrice dont la somme des probabilitÃľs de ses lignes vaut 1 chacune et toutes ses probabilitÃľs sont positives :

$$\forall i, \quad \sum_{j=1}^{N} P_{ij} = 1 \quad \text{et} \quad 0 \leq P_{ij} \leq 1.$$

Respectivement, une matrice est **stochastique par colonnes** (dite aussi **stochastique Ãă gauche**) lorsque :

$$\forall j, \quad \sum_{i=1}^{N} P_{ij} = 1 \quad \text{et} \quad 0 \leq P_{ij} \leq 1.$$

Une matrice est dite **bistochastique** lorsqu'elle est Ãă la fois stochastique par lignes et par colonnes.

Dans le cadre des chaines de Markov homogÃĺnes, toutes les matrices de transition P sont stochastiques par lignes car les probabilitÃľs de transition depuis chaque Ãľtat sont normalisÃľes (chaque ligne reprÃľsentant le vecteur des probabilitÃľs de transition depuis un Ãľtat donnÃľ vers lâĂŹensemble des autres Ãľtats). C'est-Ãă-dire que la somme des probabilitÃľs pour passer d'un Ãľtat Ãă un autre vaut 1.

De maniÃĺre analogue, si l'espace des Ãľtats est infini dÃľnombrable (par exemple $E = \{1, 2, 3, \dots\}$), on indexe les Ãľtats de la mÃłme faÃğon et la condition

$$\forall i \in E, \quad \sum_{j \in E} P_{ij} = 1,$$

reste valable.

2.2.2 ReprÃľsentation sous forme de Graphe OrientÃľ

Graphe OrientÃľ d'une ChaÃőne de Markov

Une chaÃőne de Markov homogÃĺne peut toujours Ãłtre reprÃľsentÃľe sous forme d'un graphe orientÃľ $G = (V, A)$, oÃź :

(1) V est l'ensemble des sommets, correspondant aux Ãľtats de l'espace d'Ãľtats E.

(2) A est l'ensemble des arcs, oÃź chaque arc possÃĺde une pondÃľration correspondant Ãă la probabilitÃľ de transition $P_{i,j}$. Un arc de i vers j est reprÃľsentÃľ que si $P_{i,j} > 0$.

ReprÃľsenter graphiquement une chaÃőne de Markov homogÃĺne permet de clarifier visuellement les diffÃľrentes dynamiques de transitions entre chaque Ãľtat du systÃĺme et de comprendre la structure du processus stochastique.

Exemple

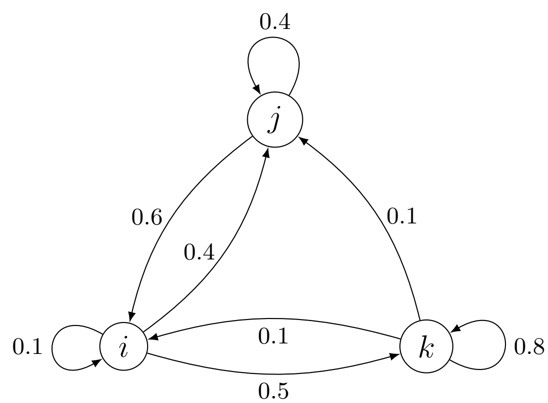
ConsidÃľrons par exemple une chaÃőne de Markov ayant trois Ãľtats i, j, k, ordonnÃľs respectivement, et dont la matrice de transition est donnÃľe par :

$$P = \begin{bmatrix} 0.1 & 0.4 & 0.5 \\ 0.6 & 0.4 & 0 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$$

Dans cette configuration, la probabilitÃľ de transition de l'Ãľtat i vers l'Ãľtat j est donnÃľe par :

$$P_{i,j} = P_{1,2} = \mathbb{P}(X_{n+1} = j \mid X_n = i) = 0.4$$

Nous obtenons ainsi le graphe suivant :



2.2.3 Matrice de transition en k-pas

Matrice de transition pour k-transitions

Soit $(X_n)_{n\in\mathbb{N}}$ une chaine de Markov homogÃÍne d'espace d'Ãľtats E fini ou dÃľnombrable tel que $\{1, 2, \ldots, N\}$. On note P sa matrice de transition.

Pour tout Ãľtat E_i et E_j et pour tout entier naturel $k \geq 1$, le coefficient de la ligne i et de la colonne j de **la matrice P^k est la probabilitÃľ de passer de l'Ãľtat E_i Ãă celui E_j en k transitions**.

Les chaÃőnes de Markov homogÃÍnes permettent de dÃľterminer l'Ãľtat d'un systÃÍme aprÃÍs k transitions, c'est-Ãă-dire au bout du k-iÃÍme mouvement dans la chaine, en Ãľlevant la matrice de transition Ãă la puissance k. Plus formellement, on a :

$$P_{i,j}^{(k)} = \mathbb{P}(X_k = j \mid X_0 = i).$$

Ce rÃľsultat vient de la propriÃľtÃľ de Markov, qui indique que la probabilitÃľ de transition d'un Ãľtat i Ãă un Ãľtat j dÃľpend uniquement du dernier pas rÃľalisÃľ, et non de tous les prÃľcÃľdents.

DÃľmonstration

La dÃľmonstration de cette propriÃľtÃľ passe par celle de l'**Ãľquation de Chapman-Kolmogorov**, telle que $\forall i, j \in E$ et $\forall n, m \in \mathbb{N}$:

$$\mathbb{P}(X_{n+m} = j \mid X_0 = i) = \sum_{k \in E} \mathbb{P}(X_m = j \mid X_0 = k) \cdot \mathbb{P}(X_n = k \mid X_0 = i).$$

Ce qui Ãľquivaut en termes matriciels :

$$P_{i,j}^{(n+m)} = \sum_{k \in E} P_{i,k}^{(n)} \cdot P_{k,j}^{(m)}.$$

Cette relation peut s'interprÃľter en disant que pour passer de i Ãă j en $n+m$ Ãľtapes il a fallu en n Ãľtapes aller de i Ãă un certain k puis en m Ãľtapes aller de k Ãă j.

On reconnait alors l'expression de l'associativitÃľ du produit matriciel tel que :

$$P^{n+m} = \underbrace{P \cdot \ldots \cdot P}_{n\ fois} \cdot \underbrace{P \cdot \ldots \cdot P}_{m\ fois} = P^n P^m.$$

Exemple

Reprenons par exemple la matrice P donnÃľe dans la sous-section 2.2.2 ci-dessus, la matrice de transition en 5 Ãľtapes, nommÃľe P^5, reprÃľsente l'ensemble des probabilitÃľs permettant de passer de chaque Ãľtat i Ãă un Ãľtat j au bout d'exactement 5 Ãľtapes :

$$P^5 = \begin{bmatrix} 0,22350 & 0,24385 & 0,53265 \\ 0,24150 & 0,26140 & 0,49710 \\ 0,20595 & 0,22252 & 0,57153 \end{bmatrix}$$

Toujours en reprennant notre exemple, la probabilitÃľ de passer de l'Ãľtat i Ãă l'Ãľtat j au bout d'exactement 5 Ãľtapes est donc de :

$$P_{i,j}^{(5)} = \mathbb{P}(X_{k+5} = j \mid X_k = i) = 24,385\%$$

2.3 Classes d'Ãľquivalence

2.3.1 Classification des Ãľtats

Dans une chaÃőne de Markov, chaque Ãľtat peut Ãłtre classifiÃľ en diffÃľrentes catÃľgories en fonction de ses liaisons avec les autres Ãľtats. Ces diffÃľrentes classifications nous permettrons d'analyser le comportement de la chaÃőne en temps long.

AccessibilitÃľ et communication

Soient i et j deux ÃľlÃľments de E. On dit que j est **accessible** Ãă partir de i si :

$$\exists n \geq 1, \quad P_{i,j}^{(n)} > 0.$$

Ce que l'on note $i \to j$.

On dit que les deux Ãľtats i et j **communiquent** si $i \to j$ et $j \to i$. On a ainsi :

$$\exists n \geq 0, \quad P_{i,j}^{(n)} > 0 \quad \text{et} \quad \exists m \geq 0, \quad P_{j,i}^{(m)} > 0.$$

Ce que l'on note $i \leftrightarrow j$.

En particulier, $i \to j$ si et seulement s'il existe une suite d'Ãľtats $(i_0, i_1, \ldots, i_n)$ avec $i_0 = i$ et $i_n = j$ telle que :

$$P(X_n = j, \ldots, X_1 = i_1 \mid X_0 = i) = P(i_{n-1}, j) \cdots P(i, i_1) > 0.$$

Ce qui veut dire qu'il existe un chemin de probabilitÃľ strictement positive qui mÃĺne de i Ãă j.

ÃĽtat rÃľcurrent/transient

Un Ãľtat i est dit **rÃľcurrent** si, en partant de cet Ãľtat, il est certain d'y retourner en un nombre fini de pas. Un Ãľtat non rÃľcurrent est dit **transient**.

On pose :

$$\tau_i = \inf\{\exists n \in \mathbb{N}^*, n \geq 1 \mid X_n = i\}$$

On dit alors que i est rÃľcurrent si :

$$\mathbb{P}_i(\tau_i < \infty) = \mathbb{P}(\tau_i < \infty \mid X_0 = i) = 1$$

Et on dit que i est transient si :

$$\mathbb{P}_i(\tau_i < \infty) = \mathbb{P}(\tau_i < \infty \mid X_0 = i) < 1$$

Ces propriÃľtÃľes impliquent que si l'Ãľtat i est rÃľcurrent (respectivement transient), on y retournera un nombre infini (respectivement fini) de fois pour une infinitÃľ de transitions.

ÃĽtat absorbant/Ãľvanescent

Un Ãľtat i est appelÃľ **absorbant** si, une fois atteint, la chaÃőne reste dans cet Ãľtat avec probabilitÃľ 1. Formellement, cet Ãľtat satisfait :

$$P_{i,i} = 1 \quad \text{et} \quad \forall j \neq i, \quad P_{i,j} = 0.$$

A l'inverse, tout Ãľtat ne respectant pas cette condition est appelÃľ Ãľtat **non-absorbant** ou encore Ãľtat **Ãľvanescent**.

ÃĽtat pÃľriodique/apÃľriodique

Un Ãľtat i est dit **pÃľriodique** si la chaÃőne ne peut revenir Ãă cet Ãľtat qu'aprÃĺs un nombre d'Ãľtapes multiple (un certain entier $d > 1$), appelÃľ la pÃľriode de i. La pÃľriode $d(i)$ est dÃľfinie comme :

$$d(i) = \mathrm{PGCD}\{n \geq 1 \mid P_{i,i}^{(n)} > 0\}.$$

Si $d(i) = 1$, l'Ãľtat i est dit **apÃľriodique**, ce qui signifie qu'il est possible de revenir Ãă cet Ãľtat Ãă chaque nouvelle transition / sans contrainte de pÃľriodicitÃľ.

2.3.2 Classes de communication

Classe de communication

Soit deux Ãľtats communicants i et j. Selon la dÃľfinition prÃľcÃľdente, on a que $\leftrightarrow$ est une relation d'Ãľquivalence. De plus dans le contexte des chaines de Markov, ces classes d'Ãľquivalences sont appelÃľes **classes de communications**. Une classe de communication est donc un ensemble d'Ãľtats accessibles les uns depuis les autres.

DÃľmonstration

Montrons que la relation $\leftrightarrow$ est une relation d'Ãľquivalence.

Il nous suffit alors de vÃľrifier les trois propriÃľtÃľs suivantes :

1. **RÃľflexivitÃľ** : Pour tout Ãľtat $i \in E$, $i \leftrightarrow i$.
   Pour tout Ãľtat $i \in E$, il existe trivialement un chemin $i \to i$. En effet, en considÃľrant le chemin de longueur zÃľro (via le fait que $P_{ii}^{(0)} = 1$ en dÃľfinissant $P^{(0)}$ comme la matrice identitÃľ), on

conclut directement que $i \leftrightarrow i$.

2. **SymÃľtrie** : Si $i \leftrightarrow j$, alors $j \leftrightarrow i$.
   Si $i \leftrightarrow j$, par dÃľfinition il existe un chemin $i \to j$, (et inversement un chemin de $j \to i$). Ainsi, la relation est symÃľtrique : $i \leftrightarrow j \Rightarrow j \leftrightarrow i$.
3. **TransitivitÃľ** : Si $i \leftrightarrow j$ et $j \leftrightarrow k$, alors $i \leftrightarrow k$.
   Supposons que $i \leftrightarrow j$ et $j \leftrightarrow k$. En composant ces deux chemins, on obtient un chemin $i \to k$. De maniÃĺre analogue, la composition des chemins $k \to j$ et $j \to i$ nous fournit un chemin $k \to i$. Ainsi, $i \leftrightarrow k$.

Comme la relation $\leftrightarrow$ satisfait les propriÃľtÃľs de rÃľflexivitÃľ, de symÃľtrie et de transitivitÃľ, elle est bien une relation d'Ãľquivalence sur E. Les classes d'Ãľquivalences ainsi obtenues sont allors appelÃľes **classes de communication**.

L'ensemble des classes de communication, engendrÃľes par $\leftrightarrow$, d'une chaÃőne de Markov sont **disjointes** et forment une **partition** de l'ensemble des Ãľtats de cette chaÃőne. Autrement dit, la relation d'Ãľquivalence $\leftrightarrow$ partitionne l'ensemble des Ãľtats d'une chaÃőne de Markov en classes de communication disjointes.

PropriÃľtÃľs de classe

On appelle **propriÃľtÃľ de classe** une propriÃľtÃľ structurelle commune Ãă tous les Ãľtats d'une mÃłme classe de communication.

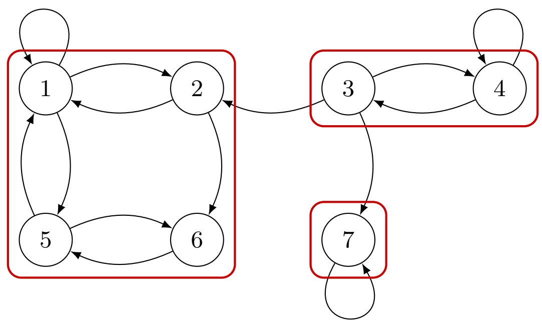
Les propriÃľtÃľs de rÃľcurrence ou transience d'un Ãľtat, ÃľnoncÃľes dans la partie 2.3.1 sont des propriÃľtÃľs de classe.

Cela signifie donc qu'au sein d'une mÃłme classe de communication, tous les Ãľtats sont soit rÃľcurrents, soit transients. On parle alors de classes de rÃľcurrence ou de transience.

L'intÃľrÃłt de ces distinctions est de diviser les chaÃőnes de Markov en sous-ensembles analytiquement indÃľpendants, ce qui permet ensuite de tirer des conclusions sur les sous-ensembles prÃľsentant ces propriÃľtÃľs.

Exemple

Prenons par exemple la chaÃőne de Markov suivante, dont la reprÃľsentation sous forme de graphe orientÃľ est donnÃľe ci-dessous :



Dans cette chaÃőne de Markov, nous avons trois classes de communication :

- La classe 1, formÃľe par les Ãľtats $\{1, 2, 5, 6\}$.
  - Cette classe est fermÃľe car il n'est pas possible de quitter cette classe une fois qu'un Ãľtat de celle-ci est atteint. En effet, il n'existe pas d'Ãľtat j tel que $P_{i,j} > 0$ pour $i \in \{1, 2, 5, 6\}$ et $j \notin \{3, 4, 7\}$.
  - Chaque Ãľtat de la classe est alors rÃľcurrent car il est certain de revenir Ãă un des Ãľtats $i \in \{1, 2, 5, 6\}$ aprÃĺs un nombre infini de transitions.
- La classe 2, formÃľe par les Ãľtats $\{3, 4\}$.
  - Cette classe est ouverte car il est possible de la quitter via $3 \to 2$ et $3 \to 7$.
  - Chaque Ãľtat de la classe est alors transient car nous ne reviendrons pas aux Ãľtats $i \in \{3, 4\}$ aprÃĺs un nombre infini de transitions.
- Et la classe 3, formÃľe par l'Ãľtat $\{7\}$.
  - Cette classe est fermÃľe car il n'est pas possible de quitter cette classe une fois que l'Ãľtat 7 est atteint. En effet, en partant de l'Ãľtat 7, nous n'avons que $7 \to 7$.
  - Chaque Ãľtat de la classe est alors rÃľccurent car une fois que l'Ãľtat 7 est atteint, la chaÃőne reste dans cet Ãľtat avec probabilitÃľ 1.

2.4 ProbabilitÃľ invariantes et thÃľorÃĺme ergodique

2.4.1 probabilitÃľ invariantes

La loi stationnaire peut permettre sous certaines conditions Ãă dÃľcrire le comportement Ãă long terme de la chaÃőne de Markov.

ProbabilitÃľ invariantes/stationnaire

Soit une chaÃőne de Markov $(X_n)_{n\in\mathbb{N}}$ Ãă espace d'Ãľtat E, et ayant pour matrice de transition P.

Une probabilitÃľ $\pi = (\pi_i)_{i\in E}$ sur l'espace d'Ãľtat E est dite invariante (ou stationnaire) si elle satisfait la relation $\pi P = \pi$.

C'est-Ãă-dire

$$\forall y \in E, \quad \sum_{x\in E} \pi(x) P_{x,y} = \pi(y).$$

Cela signifie que π est un vecteur propre Ãă droite de la transposÃľe $P^\top$, associÃľ Ãă la valeur propre 1 :

$$\pi P = \pi \iff P^\top \pi^\top = \pi^\top.$$

De plus dans notre cas, comme π est une probabilitÃľ, elle respecte les conditions suivantes :

$$\forall i \in E, \quad \pi_i \geq 0 \quad \text{et} \quad \sum_{i\in E} \pi_i = 1.$$

Existence d'une probabilitÃľ invariante

Soit $(X_n)_{n\in\mathbb{N}}$ une chaÃőne de Markov homogÃĺne sur un espace d'Ãľtats E fini, de matrice de transition P. Alors $(X_n)_{n\in\mathbb{N}}$ admet au moins une probabilitÃľ invariante.

DÃľmonstration

Par dÃľfinition, π est invariante si $\pi P = \pi$, ou encore $P^\top \pi^\top = \pi^\top$, c'est-Ãă-dire que $\pi^\top$ est un vecteur propre de $P^\top$ associÃľ Ãă la valeur propre 1. Or P et sa transposÃľe $P^\top$ ont mÃłme valeurs propres (car mÃłme polynÃťme caractÃľristique), et comme P est une matrice stochastique, on a $P\mathbf{1} = \mathbf{1}$, oÃź $\mathbf{1}$ dÃľsigne le vecteur de $\mathbb{R}^{|E|}$ dont tous les coefficients sont Ãľgaux Ãă 1. Donc 1 est vecteur propre de P et donc de $P^\top$, donc il existe un vecteur ligne μ non nul tel que $\mu P = \mu$. Soit π le vecteur ligne tel que $\pi(x) = |\mu(x)|$, pour tout $x \in E$, et montrons que π est encore vecteur propre Ãă gauche de P. Comme les coefficients de π sont positifs, il suffira alors de diviser π par la somme de ses coefficients pour obtenir une probabilitÃľ invariante. On a, pour tout $x \in E$,

$$\begin{aligned} \pi(x) = |\mu(x)| = |\mu P_x| &= \left| \sum_{y \in E} \mu(y) P_{y,x} \right|, \\ &\leq \sum_{y \in E} |\mu(y)| P_{y,x} = \sum_{y \in E} \pi(y) P_{y,x} = \pi P_x, \end{aligned}$$

donc $\pi(x) \leq \pi P_x$, pour tout $x \in E$. De plus,

$$\begin{aligned} \sum_{x \in E} \pi(x) = \sum_{x \in E} |\mu(x)| &= \sum_{x \in E} \left| \sum_{y \in E} \mu(y) P_{y,x} \right|, \\ &\leq \sum_{x \in E} \sum_{y \in E} |\mu(y)| P_{y,x} = \sum_{y \in E} \pi(y) \sum_{x \in E} P_{y,x} = \sum_{y \in E} \pi(y). \end{aligned}$$

Finalement, $\pi(x) \leq \pi P_x$ pour tout $x \in E$, et les vecteurs ont mÃłme somme, câĂŹest donc quâĂŹils sont Ãľgaux : on a

$$\sum_{x \in E} (\pi P_x - \pi(x)) = 0,$$

et comme $\pi P_x - \pi(x) \geq 0$ pour tout $x \in E$, on a $\pi P_x = \pi(x)$. Ainsi, π renormalisÃľ par la somme de ses coefficients est bien une probabilitÃľ invariante.

ChaÃőne de Markov irrÃľductible

Une chaÃőne de Markov est dite **irrÃľductible** si le graphe qui lui est associÃľ est fortement connexe.
Autrement dit, pour tous les couples (i, j) d'Ãľtats de E, avec $i \neq j$, il existe un chemin (de probabilitÃľ strictement positive) permettant d'aller de i Ãă j, et rÃľciproquement : $i \leftrightarrow j$.

ProbabilitÃľ invariantes d'une chaÃőne de Markov irrÃľductible

Toute chaine de Markov homogÃĺne irrÃľductible admet une mesure invariante strictement positive sur E et toutes les probabilitÃľ invariantes sont proportionnelles entre elles.

2.4.2 ThÃľorÃĺme ergodique

ThÃľorÃĺme ergodique (admis)

Soit E fini et (X_n) une chaÃőne de Markov irrÃľductible, pour toute fonction f, et toute loi initial ν_0

$$\forall k > 0 : \lim_{n\to\infty} \frac{1}{n-k+1} \sum_{i=k}^{n} f(X_k) = \int_E f(x)\, d\mu(x)$$

avec μ la probabilitÃľ invariante.

On peut donc estimer $\mu(x)$ avec $\mathbb{1}_{\{x\}}$, oÃź :

$$\mathbb{1}_A : B \to \{0,1\},\ A \subseteq B$$

$$\mathbb{1}_A(x) = \begin{cases} 1, & \text{si } x \in A \\ 0, & \text{sinon} \end{cases}$$

Le thÃľorÃĺme ergodique nous dit donc que la moyenne de f le long de la trajectoire de la chaÃőne se rapproche empiriquement, lorsque ce temps n est grand, de sa moyenne spatiale par rapport Ãă la loi invariante. En particulier, ceci donne une nouvelle interprÃľtation de la mesure invariante comme la frÃľquence du temps dâĂŹoccupation dâĂŹun Ãľtat en temps long.

Le thÃľorÃĺme ergodique peut Ãľgalement Ãłtre perÃğu comme une sorte de loi des grands nombres pour les chaÃőnes de Markov.

2.5 Comportement en temps long

Maintenant que nous avons dÃľfini la loi stationnaire, nous pouvons Ãľnoncer le thÃľorÃĺme de convergence d'une chaine de Markov homogÃĺne.

ThÃľorÃĺme de convergence

Soit $(X_n)_{n\in\mathbb{N}}$ une chaÃőne de Markov homogÃĺne, irrÃľductible et apÃľriodique Ãă espace d'Ãľtats fini. Soit P sa matrice de transition.
Alors, il existe une unique mesure invariante π, et pour tout Ãľtat initial $i \in E$ et tout $j \in E$,

$$\lim_{n\to\infty} P^n_{i,j} = \pi_j.$$

Ce qui implique :

$$\lim_{n\to\infty} \frac{1}{n}\sum_{k=0}^{n-1} \pi_i P^k = \pi_j.$$

Donc la loi de X_n converge vers π indÃľpendamment de l'Ãľtat initial, on a une convergence en loi.

DÃľmonstration

CommenÃğons par montrer l'unicitÃľ de la probabilitÃľ invariante. La transposÃľe d'une probabilitÃľ invariante est un vecteur propre de $P^\top$ associÃľ Ãă la valeur propre 1. Comme les valeurs propres de P et de $P^\top$ sont les mÃłmes et ont mÃłme multiplicitÃľ, il suffit donc de montrer que l'espace propre associÃľ Ãă P pour la valeur propre 1 est de dimension 1.
Soit f un vecteur propre de P pour la valeur propre 1, i.e., $Pf = f$. Comme le vecteur 1 est vecteur propre, il s'agit de montrer que f est constante.

Soit x_i le point oÃź f atteint son maximum, i.e. $f(x_i) = \max\{f(x) \mid x \in E\}$. Soit y un autre Ãľtat. Comme P est irrÃľductible, y communique avec x_i, donc il existe $n \geq 1$ tel que $P^n(x_i, y) > 0$. Comme $Pf = f$, on a aussi $P^n f = f$. Alors, d'une part,

$$f(x_i) = P^n f(x_i) = \sum_{y\in E} f(y) P^n(x_i, y),$$

et d'autre part,

$$f(x_i) = \sum_{y\in E} f(x_i) P^n(x_i, y),$$

donc on a

$$\sum_{y \in E} P^n(x_i, y) \left(f(x_i) - f(y) \right) = 0,$$

qui est une somme de termes positifs. Ainsi, comme $P^n(x_i, y) > 0$, on a $f(y) = f(x_i)$, et donc, f est constante, et la chaÃőne admet une unique probabilitÃľ invariante, notÃľe π_j.

Valeur de convergence

Soit $(X_n)_{n \in \mathbb{N}}$ une chaÃőne de Markov homogÃĺne irrÃľductible. Soit Alors, pour toute mesure initiale λ et pour tout Ãľtat j :

$$\frac{1}{n} \sum_{k=1}^{n} 1_{\{X_k = j\}} \to \pi_j,$$

oÃź π est l'unique probabilitÃľ invariante.

Convergence des mÃľthodes de Monte-Carlo

Soit X une chaine de Markov homogÃĺne irrÃľductible, alors pour toute mesure initiale λ et pour toute fonction mesurable bornÃľe f, on a :

$$\frac{1}{n} \sum_{k=1}^{n} f(X_k) \xrightarrow{n \to \infty} \int f \, d\pi \quad \text{p.s. sous } \mathbb{P}_\lambda.$$

Il s'agit de la mÃľthode utilisÃľe dans les algorithmes de Monte-Carlo.

Ces thÃľorÃĺmes nous permettront de conclure quant Ãă la preuve de Hasting-Metropolis.

3 Fonctionnement de l’Algorithme de Hasting-Metropolis

3.1 Qu’est-ce qu’une mÃľthode dite MCMC ?

Les mÃľthodes MC dites de Monte-Carlo ont pour but d’approcher de maniÃÍre empirique des valeurs par un processus alÃľatoire rÃľpÃľtÃľ un nombre suffisant de fois. Elles interviennent quand une rÃľsolution dÃľterministe est trop difficile Ãă obtenir, comme pour calculer des intÃľgrales ou gÃľnÃľrer des Ãľchantillon de distributions statistiques. Les mÃľthodes MC sont trÃÍs fiables malgrÃľ l’alÃľatoire utilisÃľ, pourvu que l’algorithme ait les bonnes optimisations telles que le nombre de rÃľpÃľtitions, la prÃľcision exigÃľe, la vitesse de convergence vers les valeurs cherchÃľe, etc.
Les mÃľthodes MC sont utilisÃľes dans beaucoup de domaines des sciences (comme la physique, la chimie, la biologie, les mathÃľmatiques statistiques, l’intelligence artificielle, la finance et la cryptographie).

Les mÃľthodes MCMC (Monte-Carlo par chaÃőnes de Markov) sont une sous-famille des algorithmes MC qui construisent, dans leur fonctionnement, une chaÃőne de Markov dont la distribution stationnaire est celle que lâĂŹon souhaite estimer.
Il devient ainsi possible d’exploiter les propriÃľtÃľs des chaÃőnes de Markov, notamment leur convergence vers une distribution stationnaire et leur propriÃľtÃľ d’ergodicitÃľ, qui peuvent Ãłtre plus utiles que la loi des grands nombres seule, utilisÃľe dans les mÃľthodes MC classiques.
La principale diffÃľrence entre les mÃľthodes MC et MCMC rÃľside dans la dÃľpendance entre les points gÃľnÃľrÃľs. Les mÃľthodes MC choisissent alÃľatoirement des points de l’espace, rÃľalisant ainsi un Ãľchantillonnage direct de la distribution cible, avec des Ãľchantillons indÃľpendants les uns des autres. Leur seul point commun est d’appartenir Ãă l’intervalle des valeurs possibles de la distribution de dÃľpart. En revanche, les mÃľthodes MCMC reposent sur une marche alÃľatoire : elles gÃľnÃÍrent chaque nouveau point Ãă partir du prÃľcÃľdent selon une rÃÍgle de transition dÃľterminÃľe. Cela crÃľe une chaÃőne de points corrÃľlÃľs, qui possÃÍde les propriÃľtÃľs des chaÃőnes de Markov.
Ainsi, lorsque la distribution est concentrÃľe sur une certaine partie de l’espace, lâĂŹutilisation de MCMC sera gÃľnÃľralement prÃľfÃľrable, car elle permet d’obtenir, dans un temps raisonnable, des Ãľchantillons reprÃľsentatifs de cette distribution.

On pourra citer les mÃľthodes MCMC d’Ãľchantillonnage de Gibbs (et ses variantes comme le Blocked Gibbs Sampling ou l’Adaptive Gibbs Sampling), qui prennent leur intÃľrÃłt lorsque les variables de la distribution sont conditionnelles. Il existe aussi lâĂŹÃľchantillonnage par tranches, qui Ãľchantillonne progressivement la distribution par "tranches" de valeurs, ou bien des optimisations de HM, telles que la mÃľthode hamiltonienne de Monte-Carlo, qui calcule le gradient pour faciliter la marche du programme et rejeter moins de valeurs, tout en gagnant en performances.

3.2 Definitions

TODO

3.3 Description des Ãľtapes de l'algorithme

Voici une description de cet algorithme :
Hasting Metropolis

1. Il faut avant tout choisir un point x_0, comme Ãľtant le premier Ãľchantillon de notre loi cible, et aussi une probabilitÃľ de transition g, en donnant les fonctions correspondant Ãă $g_y(x)$ et $g(x)$, $\forall x, y \in E$.
2. Ensuite nous itÃľrons sur t allant de 0 Ãă N (notre nombre d'itÃľrations voulu)
   — On tire x avec $g_{x_t}(x)$
   — On pose $\alpha \coloneqq \frac{\pi(x)g_x(x_t)}{\pi(x_t)g_{x_t}(x)}$ (notons que si nous ne possÃľdons qu'une densitÃľ proportionnelle Ãă π, f nous pouvons poser $\alpha \coloneqq \frac{f(x)g_x(x_t)}{f(x_t)g_{x_t}(x)}$)
   — On tire $u \in [0;1]$, tel que $u \sim \mathcal{U}(0,1)$
      — Si $u \leqslant \alpha$ alors : $x_{t+1} \coloneqq x$
      — Sinon : on conserve l'Ãľtat prÃľcÃľdent $x_{t+1} \coloneqq x$
3. La sÃľquence $\{x_0, x_1, ..., x_{N-1}\}$ constitue donc l'Ãľchantillon obtenue Ãă partir de la chaÃőne de Markov associÃľ Ãă la loi π

3.4 Preuve de l'algorithme

TODO

3.5 Exemples d'applications simples de HM

Afin de comprendre le dÃľroulÃľ de l'algorithme, on peut prendre un graphe d'un dÃľbut d'une simulation de HM. Attention, il s'agit d'un espace Ãă Ãľtats continus ce qui diffÃÍre en soi du reste du rapport, bien que les propriÃľtÃľs y restent Ãľquivalentes. Cet exemple est visuel et simple donc pertinent pour comprendre le dÃľroulÃľ de l'algorithme.

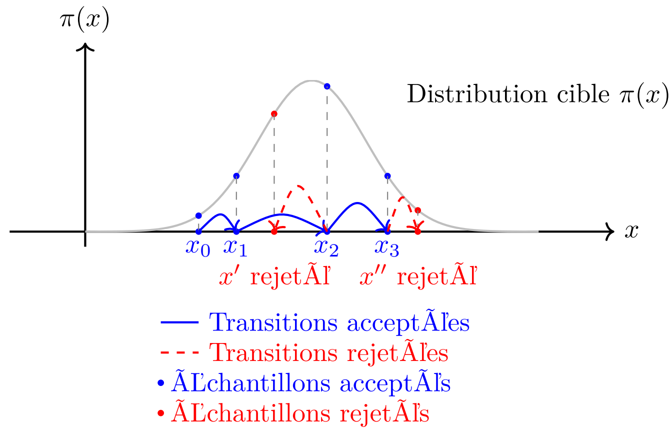
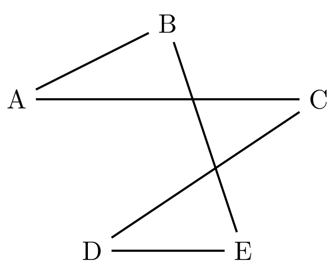
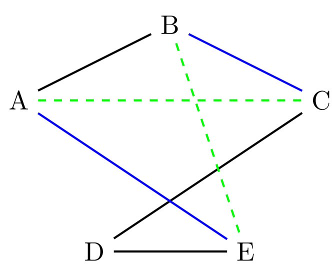


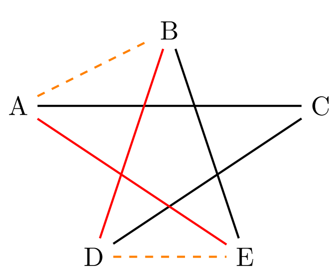
Diagramme initial Ãă optimiser



Exemple de proposition retenue par HM



Exemple de proposition rejetÃľe par HM



3.6 Application au cas du voyageur de commerce

Dans le cas du problÃĺme du voyageur de commerce (dit “TSP”), l'algorithme nÃľcessite une application adaptÃľe. Notre espace d'Ãľtat E correspond aux permutations des points. Par exemple, dans le cas de 3 points x_1, x_2, x_3, $E = \{(x_1, x_2, x_3), (x_2, x_1, x_3), (x_2, x_3, x_1), (x_1, x_3, x_2), (x_3, x_2, x_1), (x_3, x_1, x_2)\}$, ce qui, avec notre implÃľmentation actuelle, nÃľcessiterait de manipuler une matrice 6×6. Lorsque le nombre de villes augmente, nous aurions besoin d'une matrice carrÃľe de taille $n!$, oÃź n est le nombre de points, ce qui devient rapidement irrÃľaliste.

Nous dÃľfinissons donc des fonctions adaptÃľes sur l'espace d'Ãľtat E :

— Une fonction de distance totale sur les permutations des coordonnÃľes des villes $D : E \to \mathbb{R}$,

$$D(p) = \sum_{i=1}^{n} \|x_i - x_{i+1}\| : \forall p = (x_1, x_2, \dots, x_n) \in E, x_{n+1} = x_1$$

— Une distribution cible, avec $\beta \in \mathbb{R}$ un paramÃĺtre ajustable

$$\pi(p) = e^{-\beta \cdot D(p)}$$

— Une variable alÃľatoire $S : E \to E$, suivant une loi de chaÃőne de Markov. Nous pouvons modÃľliser cette transition par une fonction `permutation`($[x_1, \dots, x_n]$), qui renvoie la mÃłme liste avec seulement deux villes permutÃľes.

Nous pouvons donc redÃľfinir l'algorithme avec le pseudo code suivant :
Hasting Metropolis TSP

1. On prend p_0, une permutation donnÃľe
2. Ensuite nous itÃľrons sur t allant de 0 Ãă N (notre nombre d'itÃľrations voulu)
   - On tire p avec la variable alÃľatoireS_{p_t}, on peut utilisÃľ la fonction “permutation” dÃľfini plus tÃťt.
   - Calculer $\alpha \coloneqq \frac{\pi(p)}{\pi(p_t)}$ si la transition est symÃľtrique, c'est-Ãă-dire $\mathbb{P}(S_{p_t} = p) = \mathbb{P}(S_p = p_t)$.
   - Sinon $\alpha \coloneqq \frac{\pi(p)\mathbb{P}(S_p = p_t)}{\pi(p_t)\mathbb{P}(S_{p_t} = p)})$
   - On tire $u \in [0; 1]$, tel que $u \sim \mathcal{U}(0, 1)$
     - Si $u \leqslant \alpha$ alors : $p_{t+1} \coloneqq x$
     - Sinon : on conserve l'Ãľtat prÃľcÃľdent $p_{t+1} \coloneqq x$
3. La sÃľquence $\{p_0, p_1, \dots, p_{N-1}\}$ constitue donc l'Ãľchantillon obtenue Ãă partir de la chaÃőne de Markov associÃľ Ãă la loi π

On peut retirer de la sÃľquence $\{p_0, p_1, \dots, p_{N-1}\}$ le meilleur rÃľsultat en prenant la $p_m \in E : D(p_m) = \min(D(E))$

3.7 Comparaison avec d'autres mÃľthodes d'Ãľchantillonnage

TODO

4 Conclusion

TODO

5 Annexe

5.1 Code source

Cette section contient des implÃľmentations des algorithmes abordÃľs. Le code source complet est disponible Ãă lâĂŹadresse suivante : `https://github.com/CorentinVaillant/ProjetMath-HastingMetropolis`

5.1.1 Code utilitaires

Voici le code source de ‘utilis.py‘ qui contient des fonctions utiles au reste du projet. Seul le code des utilitaires mathÃľmatiques est prÃľsentÃľ ici âĂŤ les fonctions utilitaires liÃľes aux graphiques sont disponibles directement dans le dÃľpÃťt.

```python
import numpy as np
import networkx as nx
import matplotlib.pyplot as plt

from math import factorial , comb

def poisson(l):
    return lambda k: ((l**k)/(factorial(k)))*np.exp(-l)

def binomial(n, p):
    return lambda k: comb(n, k) * p**k * (1-p)**(n-k)

def geo(p):
    return lambda k: p * ((1-p)**(k-1))
```

5.1.2 ImplÃľmentation de l'algorithme de Hastings-Metropolis

LâĂŹimplÃľmentation de lâĂŹalgorithme de Hastings-Metropolis, comme dÃľcrit en 3.2 et relativement idiomatique, voici son implÃľmentation, en diffÃľrentes fonctions :

La fonction `next_state_markov(x, transitions)` renvoie un Ãľtat y (un entier), avec une probabilitÃľ donnÃľe par $g_x(y)$:

```
def next_state_markov(x: int, transitions: np.array) -> int:
    return np.random.choice(len(transitions), p=transitions[x])
```

La fonction `markov_chain_from_tirage(tirage, nb_etats)` transforme un tirage en une matrice de transition, reprÃľsentant une chaÃőne de Markov :

```
def markov_chain_from_tirage(tirage, nb_etats) -> np.array:
    sommes = np.full((nb_etats, nb_etats), 0)
    derniere_etat = tirage[0]
    for col in tirage[1:]:
        sommes[derniere_etat][col] += 1
        derniere_etat = col
    result = []
    for i in range(nb_etats):
        denominateur = sum(sommes[i])
        result.append([
            sommes[i][j] / denominateur if denominateur != 0 else 0
            for j in range(nb_etats)
        ])
    return np.array(result)
```

La fonction `hasting_metropolis_tirage(x0, distribution_cible, proposition, iterations)` renvoie un tirage en suivant lâĂŹalgorithme de Hastings-Metropolis :

```
def hasting_metropolis_tirage(x0: int, distribution_cible,
                                  proposition: np.array, iterations
                                  : int = 1000) -> np.array:
    x_t = x0
    tirage = [x_t]

    for _ in range(iterations):
        x = next_state_markov(x_t, proposition)
        pi_xt = distribution_cible(x_t)
        pi_x = distribution_cible(x)
        g_xt_x = proposition[x_t][x]
        g_x_xt = proposition[x][x_t]

        if pi_xt * g_xt_x == 0:
            alpha = 1  # Evite la division par zero
        else:
            alpha = min(1, (pi_x * g_x_xt) / (pi_xt * g_xt_x))

        u = np.random.uniform(0, 1)

        if u <= alpha:
            x_t = x
        tirage.append(x_t)
```

```
    return np.array(tirage)
```

Enfin, la fonction `hasting_metropolis` combine tout pour gÃĺnÃĺrer directement une chaÃőne de Markov Ãă partir de l'algorithme :

```
def hasting_metropolis(x0: int, distribution_cible,
                       proposition: np.array, iterations: int =
                                                          1000) ->
                                                          np.array:
    tirage = hasting_metropolis_tirage(x0, distribution_cible,
                                       proposition, iterations)
    return markov_chain_from_tirage(tirage, proposition.shape[0])
```

5.1.3 Applications du code d'Hasting Metropolis

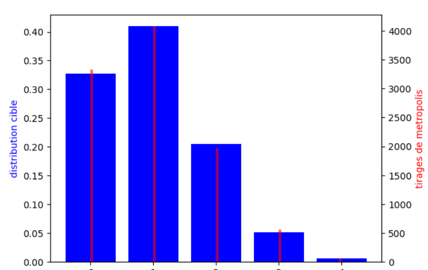
Dans cette section vous trouverais des exemples d'application des diffÃĺrents algorithmes implÃĺmentÃĺs

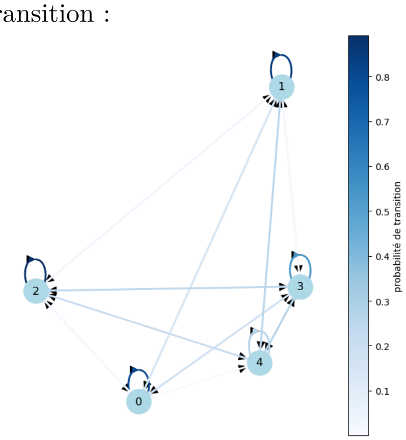
Exemples Hastings-Metropolis Voici les exemples d'application d'Hasting-Metropolis que nous avons effectuer, Ãă partir de la librairie mathplotlib, le code source est trouvable sur `https://github.com/CorentinVaillant/ProjetMath-HastingMetropolis`.

Application Ãă une loi binomial :

```
N = 5

distribution_cible = binomial(N,0.2)

#matrice de transition quelconque
proposition = np.array([ ligne / ligne.sum()
    for ligne in [np.random.rand(N) for _ in range(N)]
])


tirage = hasting_metropolis_tirage(0,distribution_cible,proposition
                                   ,iterations=10_000)
mk_c = markov_chain_from_tirage(tirage,N)
```

rÃĺsulat et graphe de la matrice de transition :

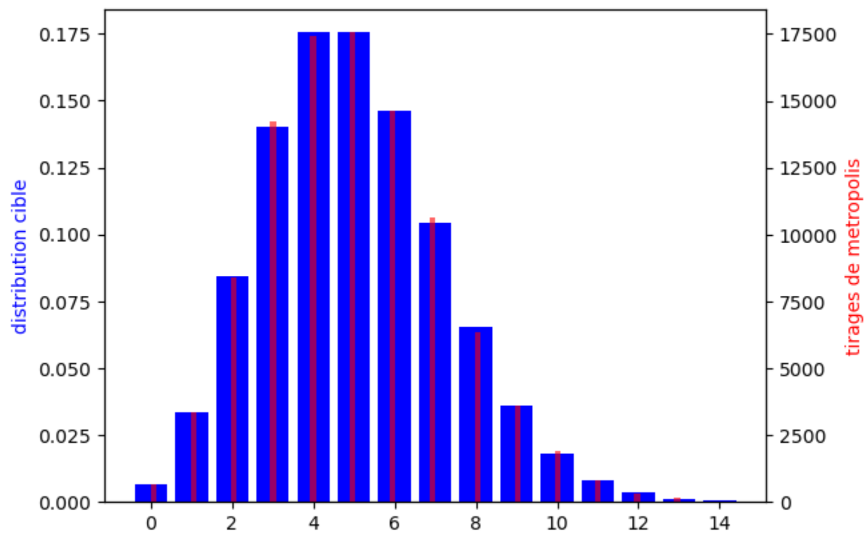




Application Ãă une loi de Poisson :

```
N = 15

distribution_cible = poisson(5)

#matrice de transition uniforme de taille NxN
proposition = np.full((N,N),1/N)

tirage = hasting_metropolis_tirage(0,distribution_cible,proposition
                                    ,iterations=100_000)
mk_c = markov_chain_from_tirage(tirage,N)
```

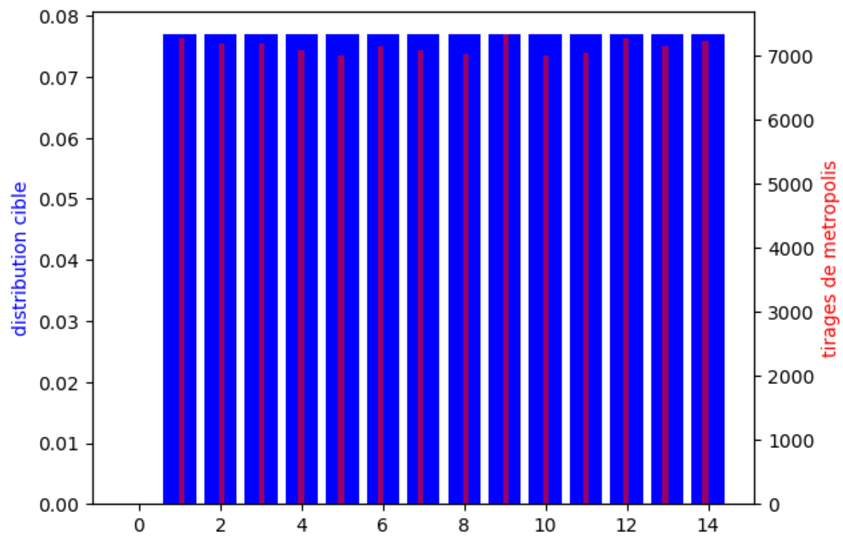
rÃľsulats :



Application Ãă une loi uniform :

```
N = 15

distribution_cible = lambda k : 1/(N-2) if 0<k and k<N else 0

#matrice de transition aleatoire
proposition = np.array([ ligne / ligne.sum()
    for ligne in [np.random.rand(N) for _ in range(N)]
])


tirage = hasting_metropolis_tirage(0,distribution_cible,proposition
                                   ,iterations=100_000)
mk_c = markov_chain_from_tirage(tirage,N)
```

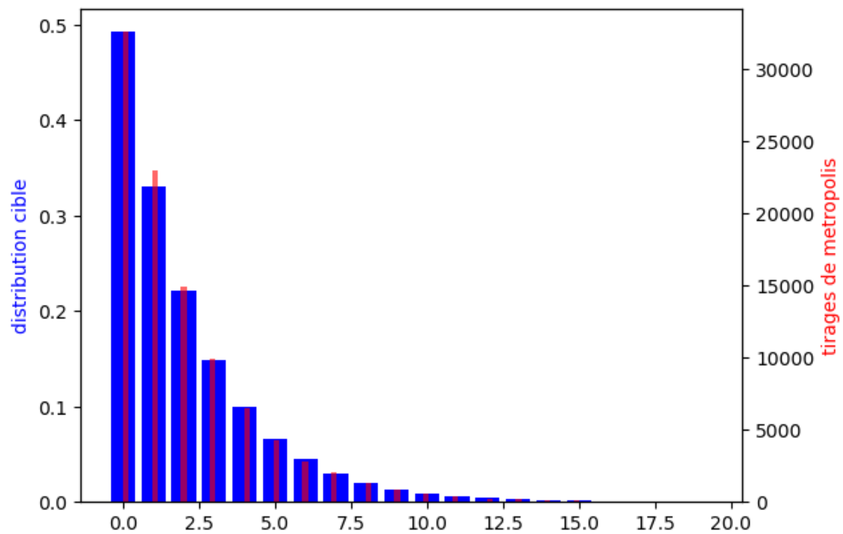
rÃľsulats :



Application Ãă une loi gÃľomÃľtrique :

```
N = 20

distribution_cible = geo(0.33)

# matrice de transition aleatoire
proposition = np.array([ ligne / ligne.sum()
    for ligne in [np.array([abs(np.random.normal(i,1)) for _ in
                                      range(N)]) for i in range(N)]
])

tirage = hasting_metropolis_tirage(0,distribution_cible,proposition
                                  ,iterations=100_000)
mk_c = markov_chain_from_tirage(tirage,N)
```

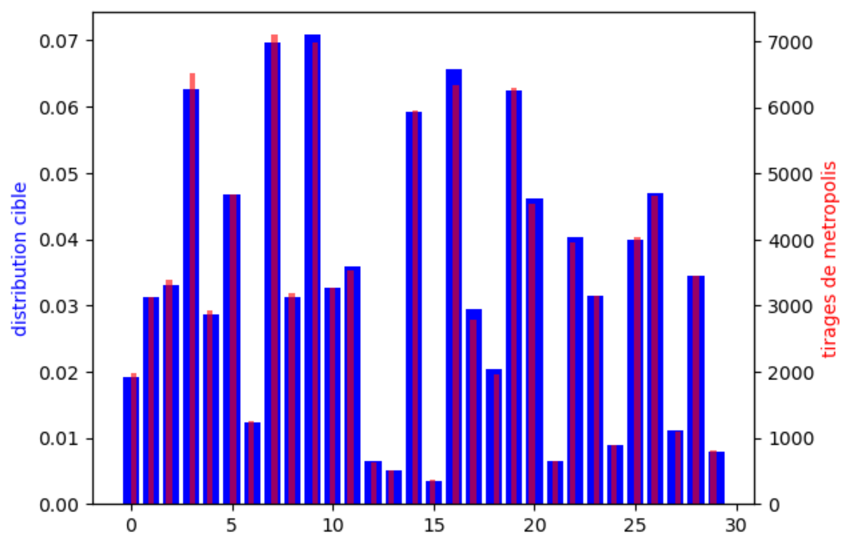
rÃľsulats :



Application Ãă une loi quelconque :

```
N = 30

distribution = np.random.rand(N)
distribution = distribution / distribution.sum()

distribution_cible = lambda k : distribution[k] if k < N and 0<=k
                                  else 0

#matrice de transition aleatoire
proposition = np.array([ ligne / ligne.sum()
    for ligne in [np.array([abs(np.random.normal(i,1)) for _ in
                                      range(N)]) for i in range(N)]
])

tirage = hasting_metropolis_tirage(0,distribution_cible,proposition
                                  ,iterations=100_000)
mk_c = markov_chain_from_tirage(tirage,N)
```

rÃľsulats :



5.2 Sources

Documents textuels :

- https://fr.wikipedia.org/wiki/ChaÃőne_de_Markov
- https://dms.umontreal.ca/~bedard/BergeronL_rapport_final.pdf
- https://www.math.univ-paris13.fr/~tournier/fichiers/agreg/2014/cours_markov.pdf
- https://www.math.u-bordeaux.fr/~mchabano/Agreg/ProbaAgreg1213-COURS5-CM.pdf
- https://www.imo.universite-paris-saclay.fr/~pierre-loic.meliot/agreg/markov.pdf
- https://www.college-de-france.fr/fr/agenda/cours/apprentissage-et-generation-par-echantillonnage-aleatoire/algorithme-de-metropolis-hasting
- https://www.math.u-bordeaux.fr/~mibonnef/mimse-markov/recurence-transience.pdf
- https://perso.univ-rennes1.fr/jean-christophe.breton/agreg/AGREG/COURS/ch-mark2.pdf
- https://fr.wikipedia.org/wiki/Processus_stochastique
- https://fr.wikipedia.org/wiki/MÃľthode_de_Monte-Carlo_par_chaÃőnes_de_Markov
- https://www.math.u-bordeaux.fr/~mchabano/Agreg/ProbaAgreg1314-COURS5-CM.pdf
- https://fr.wikipedia.org/wiki/Algorithme_de_Metropolis-Hastings#cite_note-2
- https://www.radcliffe.harvard.edu/news-and-ideas/flash-of-genius
- https://fr.wikipedia.org/wiki/Probl%C3%A8me_du_voyageur_de_commerce
- https://www.cambridge.org/core/journals/mathematical-proceedings-of-the-cambridge-philosophical-society/article/abs/shortest-path-through-many-points/F1C28B5730B94887F4659FCBF8A1F2BB

Documents vidÃľos :

- https://www.youtube.com/watch?v=yCv2N7wGDCw
- https://www.youtube.com/watch?v=MxI78mpq_44
- https://www.youtube.com/watch?v=e0ZHDK4DSEI&list=PLWoShwK0FEjovcc32x9LbpDTf8pquPimV
- https://www.youtube.com/playlist?list=PLWoShwK0FEjovcc32x9LbpDTf8pquPimV