# Probabilités et statistiques

## *Chaînes de Markov : PageRank*

Becirspahic Lucas
Vu Huc

Université Pierre et Marie Curie - 20 avril 2017

# Introduction

Dans ce projet, nous étudions le modèle des chaînes de Markov à travers une application concrète : l'algorithme *PageRank* de Google. Il s'agit de définir des structures en Python permettant de représenter les structures de pages de l'algorithme, appelées nanoWeb, puis de simuler son fonctionnement.

# 1 Implémentation en Python

On dispose d'un certain nombre de fichiers :

- `datastructures.py` contient la structure de graphe (`SimpleWeb`) utilisée dans la suite
- `nanowebs.py` fournit les fonctions de génération des trois nanoWeb fournis dans l'énoncé
- `internautes.py, simulation.py` contiennent les classes de simulation
- les fichiers de test pour chaque type de simulation

Ainsi, un sommet du graphe représente une page et un arc un déplacement entre deux pages.

# 2 Simulation du comportement d'un internaute

Dans un premier temps, on souhaite simuler le déplacement d'un internaute dans un nanoWeb., *i.e.* de sommet en sommet, en fixant le sommet de départ. On compte ensuite le nombre de passages dans chaque nœud, pour obtenir une estimation de la distribution de probabilité $\pi_t = [p_1, ..., p_n]$ où $p_i$ correspond à la probabilité de finir sur le nœud $i$ à l'itération $t$ en partant d'un sommet initial donné. Il faut également définir une condition d'arrêt : on introduit un nombre d'itérations maximal et un seuil $\epsilon = \max |\pi_t(i) - \pi_{t+1}(i)|$. Trois situations se présentent alors :

– La chaîne de Markov est ergodique, *i.e.* irréductible et apériodique : quelque soit le nœud initial, on a la même probabilité de finir sur un nœud $i$. C'est le cas du nanoWeb 2 : en effet, on remarque que le PGCD des chaînes périodiques du graphe est égal à 1. Par exemple, on obtient une distribution grâce à la simulation dans la classe `Internaute` :

$$[0.025, 0.0375, 0.1625, 0.2125, 0.225, 0.1, 0.0875, 0.0875, 0.025, 0.0375]$$

– Il existe un sous-ensemble absorbant dans la chaîne de Markov : l'internaute se trouvera quasiment toujours dans les états de ce sous-ensemble à la fin du parcours. C'est le cas du nanoWeb 1, où le sous-ensemble {7, 8} est absorbant. Par exemple, en simulant un parcours depuis le nœud 3, on obtient la distribution suivante :

$$[0, 0, 0.002, 0, 0.002, 0.006, 0.006, 0.49, 0.49, 0.002]$$

La probabilité de finir sur le nœud 7 et le nœud 8 est significativement plus élevée. Si la chaîne de Markov n'est pas irréductible, la distribution de probabilité obtenue dépend donc du nœud de départ. Dans le cas du nanoWeb 3, qui ne vérifie pas la condition d'irréductibilité, on s'attend à avoir une probabilité nulle pour les sommets qui ne sont pas contenus dans la composante fortement connexe du sommet de départ.

En outre, notons que le calcul de seuil de convergence pose des problèmes d'arrondi lorsque $\epsilon$ prend des valeurs de plus en plus petites, proches de 0. On a cependant fixé un $\epsilon$ égal à 0.001, ce qui ne devrait pas poser de problème.

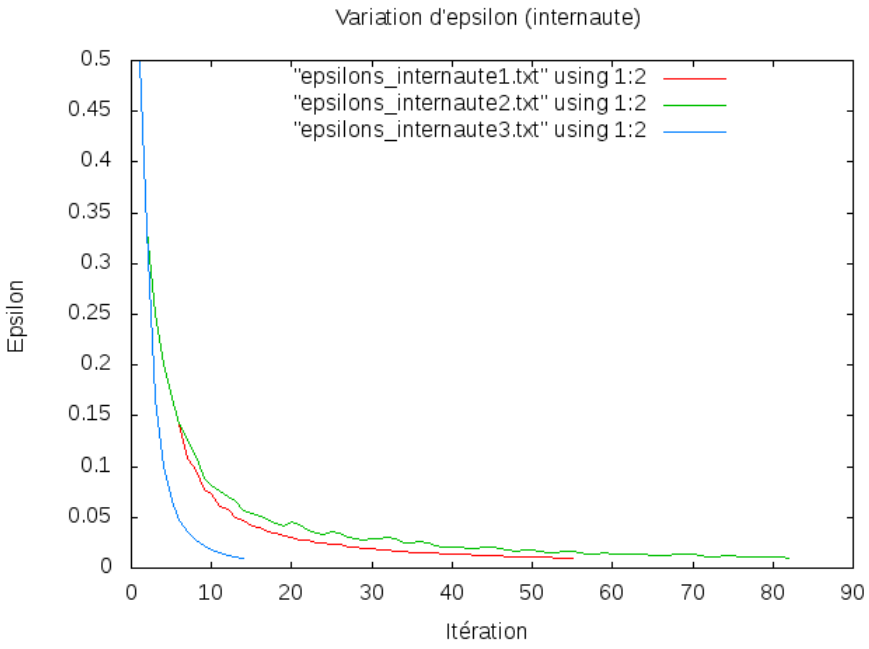


FIGURE 1 – $\epsilon$ Internaute

Le déplacement d'un internaute correspond à une seule situation possible lorsqu'on part d'un nœud initial donné. On aimerait cependant observer ce qu'il se passe si l'on répète l'expérience plusieurs fois. Par exemple, dans le nanoWeb 3, si l'on part du nœud 9 et que l'internaute reste coincé au nœud 3 en passant par le nœud 1, alors la probabilité d'aller sur le nœud 2 et le nœud 0 sera nulle pour cette séquence d'actions, alors qu'il est tout à fait possible d'aller sur ces nœuds depuis le nœud 9.

La simulation utilisant la classe `Internaute` a donc des limites, il faut alors trouver un moyen de simuler un déplacement sans obligatoirement fixer un seul point de départ, et ce sur plusieurs itérations.

## 3 Vecteurs-matrices

Soit $\pi_t$ la probabilité de se trouver dans chacun des nœuds du graphe à l'instant $t$ ($\pi_t$ est une matrice-ligne). Soit $P$ la matrice de transition, composée des probabilités de transition $(p_{ij})$, avec $p_{ij} = P(X_n = j \mid X_{n-1} = i)$ la probabilité de passer du sommet $i$ au sommet $j$. D'après la formule des probabilités totales, on sait que la probabilité d'être au nœud $k$ à l'instant $t+1$ est :

$$P(k) = \sum_{\substack{i=0 \\ i \neq k}}^{n} P(X_n = k \mid X_{n-1} = i) P(X_{n-1} = i)$$

On sait que $P(X_{n-1} = i)$ correspond à $\pi_t(i)$ et $p_{ik} = P(X_n = k \mid X_{n-1} = i)$. Ainsi, la probabilité d'être au noeud $k$ à l'instant $t+1$ est égal à $\pi_t$ multiplié par la $k$-ième colonne de la matrice de transition $P$. Comme on réitère pour chaque sommet, on obtient $\pi_{t+1} = \pi_t \cdot P$.

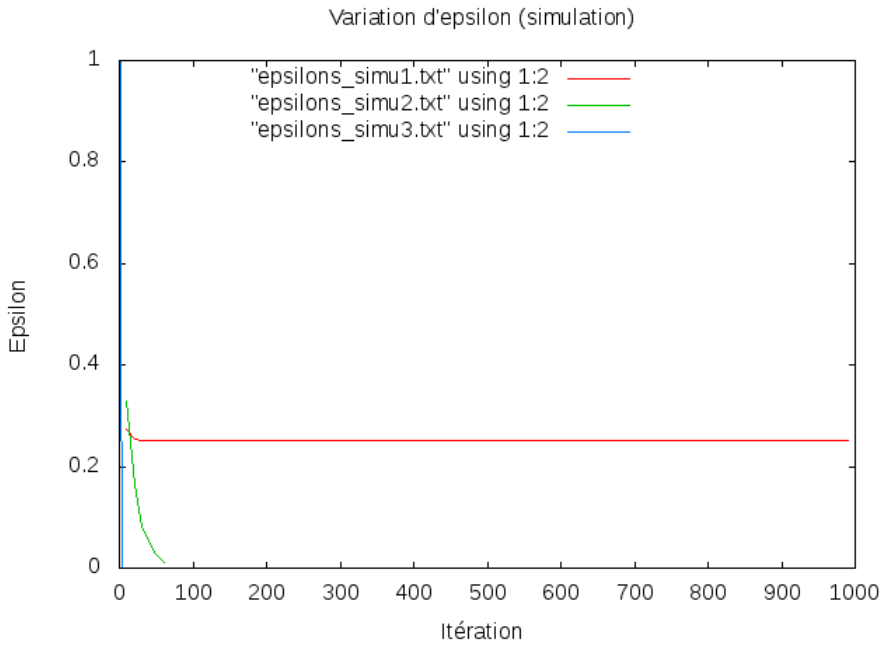


FIGURE 2 – $\epsilon$ Simulation

Pour le nanoWeb 1, on remarque qu'après un certain nombre d'itérations, la valeur de $\epsilon$ se stabilise à 0.25. Cela correspond à la différence $|\pi_t(7) - \pi_t(8)|$, 7 et 8 constituant les sommets du sous-ensemble absorbant. A l'instant $t$, on a une probabilité de 0.625 d'être sur le nœud 7 et 0.375 d'être sur le nœud 8. A l'instant $t+1$, on est forcément dans le nœud 8 si on était dans le nœud 7, la probabilité d'être sur le nœud 8 à l'instant $t+1$ est alors 0.625. Réciproquement, la probabilité d'être dans le nœud 7 à l'instant $t+1$ est 0.375.

Dans cette implémentation, on choisit manuellement la distribution de probabilité initiale $\pi_0$, ce qui permet de ne pas se cantonner à un seul point de départ, mais de positionner des probabilités sur certains sommets initiaux.

Comme $\pi_{t+1} = \pi_t \cdot P$, on peut montrer que $\pi_n = \pi_0 \cdot P^n$. On peut ainsi étudier la convergence de la suite des puissances de P.

## 4 Puissances de matrices

Dans ce cadre, $P^n$ correspond à la matrice de transition en $n$ étapes, $\lim_{n\to\infty} P^n$ est donc la distribution stationnaire (si convergence). Pour vérifier la convergence de la suite des puissances de P, on définit

$$\epsilon = \max |P_{ij}^n - P_{ij}^{n+1}|$$

En particulier, pour les nanoWeb 1 et 3 dont les chaînes de Markov ne sont pas ergodiques et comportent des sous-ensembles absorbants, la suite des puissances de P converge très rapidement, elle converge en réalité vers les états absorbants. En revanche, la chaîne du nanoWeb 2 étant ergodique, la suite des puissances de P met plus de temps pour converger, pour obtenir une matrice dont toutes les lignes sont égales.

Dans tous les cas, la convergence est beaucoup plus rapide qu'avec les simulations pour les nanoWeb fournies dans l'énoncé (petit nombre de nœuds).

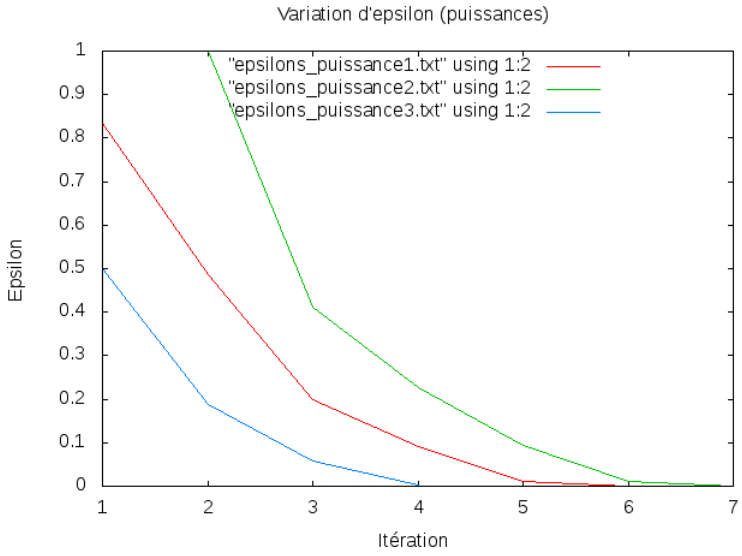


FIGURE 3 – $\epsilon$ Puissances

## 5 Génération de Webs

On souhaite générer des `SimpleWeb` ergodiques, *i.e* irréductibles et apériodiques.Pour obtenir une chaîne de Markov apériodique, on ajoute un cycle de $n-1$ nœuds. Il suffit en effet que le PGCD des périodes des états soit égal à 1. Pour la propriété d'irréductibilité, il faut que le graphe n'ait qu'une seule composante fortement connexe, sans sous-ensemble absorbant. Pour cela, on adopte des graphes en forme d'anneau. En outre, on vérifie que la chaîne de Markov générée est ergodique en calculant la matrice obtenue au terme de la convergence de la suite des puissances de la matrice de transition.

Une fois les graphes générés, on peut visualiser les temps de calcul des différentes algorithmes d'estimation de la distribution stationnaire.

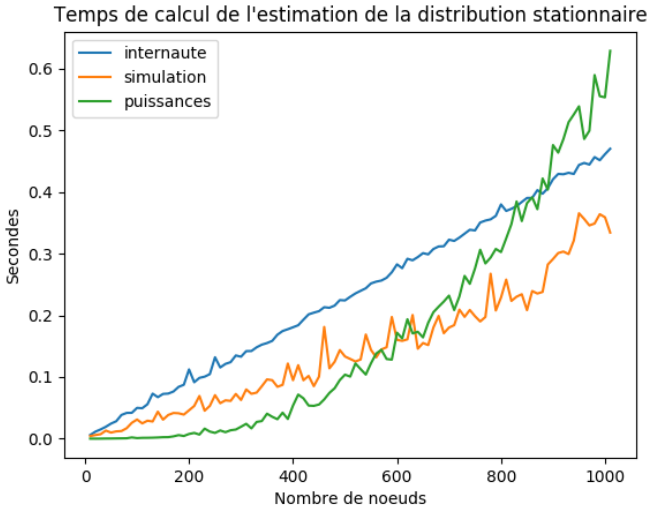


FIGURE 4 – Estimation de la distribution stationnaire

On observe que le calcul de la distribution stationnaire, en fonction du nombre de nœuds, a une complexité linéaire pour l'internaute et la simulation, et en $O(n^2)$ pour la suite des puissances de la matrice de transition (on multiplie une matrice par une autre matrice, ce qui est cohérent à la complexité théorique). Pour des petites valeurs, l'estimation de la distribution est plus rapide avec la suite des puissances : 6 itérations contre 100 itérations pour la simulation. Cela s'explique par la convergence plus rapide pour ces valeurs. Au contraire, pour des grandes valeurs (à partir d'environ 600), le calcul de la distribution est de plus en plus longue en raison de la multiplication entre deux matrices de transition, qui possèdent une dimension de plus en plus grande.

## 6 Conclusion

Le *PageRank* est une méthode efficace pour classer les pages Web, mais présente néanmoins de nombreuses subtilités. Si la chaîne de Markov n'est pas irréductible, il y aura forcément un sous-ensemble absorbant, dont les états auront un poids conséquent par rapport aux autres nœuds à l'extérieur de ce sous-ensemble. Parmi les méthodes à notre disposition pour calculer la distribution de probabilités, on préférera utiliser les puissances de matrices pour les chaînes dont le nombre de nœuds n'excède pas 600. Sinon, la classe de simulation se révèle très efficace : pour une chaîne de Markov ergodique, toutes les lignes de la matrice de transition sont identiques, il suffit d'en calculer une seule, peu importe le point de départ.