

AAIA TP 3IF - PageRank

Florian Rascoussier

Christine Solnon

29 décembre 2023



Introduction

L'algorithme PageRank, développé par Larry Page et Sergey Brin à la fin des années 1990, est au cœur du moteur de recherche original de Google. Il a révolutionné la manière dont les informations sont récupérées sur le vaste espace qu'est l'Internet. L'idée fondamentale derrière PageRank est simple et élégante : mesurer l'importance d'une page web en fonction du nombre et de la qualité des liens qui y pointent. En pratique, cela revient à considérer le web comme un graphe, où chaque page est un sommet et chaque lien est une arête dirigée. L'algorithme effectue ensuite une marche aléatoire sur ce graphe pour déterminer la probabilité qu'un "surfeur" aléatoire arrive à une page donnée, cette probabilité servant d'indicateur de l'importance ou du "rang" de la page.

Toutefois, la mise en œuvre effective de PageRank dans un contexte réel est loin d'être triviale. Elle doit tenir compte de nombreux défis techniques et théoriques, notamment comment traiter les sommets absorbants ou comment garantir la convergence de l'algorithme vers une distribution stable. Ce document vise à explorer certains de ces défis, en particulier comment transformer une matrice de transition classique en une matrice stochastique ergodique, et démontre pourquoi et comment de telles transformations préservent l'essence de l'algorithme PageRank tout en garantissant ses propriétés mathématiques fondamentales.

Nous encourageons vivement tous les étudiants et lecteurs à participer activement à l'amélioration de cette correction et du code associé. Si vous trouvez des erreurs, des ambiguïtés ou des omissions, ou si vous avez des suggestions pour améliorer le contenu, n'hésitez pas à soumettre une Pull-Request sur le GitHub de la correction à l'adresse suivante : https://github.com/0nyr/AAIA_3IF_TP_PageRank. Votre contribution est précieuse et aidera à rendre ces ressources plus précises et utiles pour tous.

Table des Matières

1 Exercice 1 : Marche aléatoire et Matrice de Transition dans PageRank	2
1.1 Rappels et explications	2
1.1.1 Marche aléatoire sur le graphe	2
1.1.2 Matrice de transition M	2
1.1.3 Puissances de la matrice de transition M^k	2
1.2 Démonstration formelle par récurrence	3
1.2.1 Notations et Déclaration des Variables	3
1.2.2 Preuve par Induction sur k	3
1.2.3 Initialisation : $P(1)$	4
1.2.4 Récurrence : $P(k) \rightarrow P(k+1)$	4
2 Exercice 2 : Première implémentation de PageRank	4
3 Exercice 3 : Problème des nœuds absorbants	5
4 Exercice 4 : Modification de PageRank pour les nœuds absorbants	6
5 Exercice 5 : Matrice ergodique et convergence	7
5.1 Convergence de l'algorithme PageRank	7
5.2 Matrice ergodique	7
5.3 Démonstration de l'ergodicité et de la stochasticité de la matrice de transition M'' .	8
5.3.1 Notations et Déclaration des Variables	8
5.3.2 Définitions	8
5.3.3 Démonstration	8
6 Exercice 6 : PageRank avec téléportation (surfeur aléatoire)	9
7 Exercice 7 : Calcul de rangs après convergence	10
Acronymes	11
Références	11

Table des figures

1	Représentation du graphe G_1 fourni dans le fichier res/example_1.txt	5
2	Représentation du graphe G_2 fourni dans le fichier res/example_2.txt	6

Codes et programmes

1	Résultat de l'algorithme PageRank sur le graphe G_1 fourni, pour $k = 4$	5
2	Résultat de l'algorithme PageRank sur le graphe G_2 fourni, pour $k = 4$, avec la somme des valeurs du vecteur s_k à chaque itération k	5
3	Résultat de l'algorithme PageRank sur le graphe G_2 fourni, pour $k = 4$, avec la somme des valeurs du vecteur s_k à chaque itération k	6
4	Résultat de l'algorithme PageRank sur le graphe G_2 fourni, pour $k = 4$, avec la somme des valeurs du vecteur s_k à chaque itération k , pour PageRank modifié avec téléportation et $\alpha = 0.9$	9
5	Résultat de l'algorithme PageRank après convergence sur le graphe G_2 , avec $\alpha = 0.9$ et $\epsilon = 1.0 \times 10^{-10}$	10
6	Résultat de l'algorithme PageRank après convergence sur le graphe G_2 , avec $\alpha = 0.9$ et $\epsilon = 1.0 \times 10^{-10}$	10

1 Exercice 1 : Marche aléatoire et Matrice de Transition dans PageRank

Montrez que pour tout $k \geq 1$ et tout couple de sommets $(i, j) \in S \times S$, $M^k[i][j]$ est égal à la probabilité d'arriver sur la page j en k clics à partir de la page i .

Pour comprendre comment la matrice de transition M se rapporte à la probabilité d'arriver sur une page donnée en k clics, considérons d'abord ce que signifie une marche aléatoire sur le graphe du web $G = (S, A)$.

1.1 Rappels et explications

Les propriétés étudiées dans le cours¹ sur la théorie des graphes aborde les notions de marche aléatoire et de matrice de transition [1]. Ces notions sont essentielles pour comprendre l'algorithme PageRank.

1.1.1 Marche aléatoire sur le graphe

Dans le contexte de l'algorithme PageRank, la marche aléatoire représente un utilisateur naviguant sur le Web en cliquant sur des liens de manière aléatoire. À chaque étape :

- L'utilisateur est sur une page i (un sommet dans le graphe).
- Il clique sur un lien sortant au hasard, le menant à une page j (un autre sommet).

1.1.2 Matrice de transition M

La matrice de transition M reflète la probabilité de passer d'une page à une autre en un seul clic. Ainsi, $M[i][j]$ représente la probabilité de passer de la page i à la page j . Elle est définie comme :

- $M[i][j] = \frac{1}{d^+(i)}$ si un lien (arc) existe de i à j (c'est-à-dire $i \rightarrow j$).
- $M[i][j] = 0$ sinon.

Où $d^+(i)$ est le nombre de liens sortants de la page i (son demi-degré extérieur).

1.1.3 Puissances de la matrice de transition M^k

Lorsque l'on considère M^k , avec $k > 1$, on regarde le processus de navigation après k clics. Voici comment cela fonctionne :

- $M^1 = M$ représente la probabilité de passer d'une page à une autre en un seul clic.
- $M^2 = M \times M$ représente la probabilité de passer d'une page à une autre en deux clics.
- Et ainsi de suite, jusqu'à M^k , qui représente la probabilité de passer d'une page à une autre en k clics.

1. Voir le cours d'AAIA, section 3.1 p11. <http://perso.citi-lab.fr/csolnon/supportAlgoGraphes.pdf>

Pour montrer que $M^k[i][j]$ est égal à la probabilité d'arriver sur la page j en k clics à partir de la page i , on utilise la propriété des marches aléatoires et des matrices de transition. La démonstration informelle peut être faite par induction sur k :

- **Initialisation ($k=1$)** : Par définition, $M[i][j]$ est la probabilité d'aller de i à j en un clic.
- **Récurrence** : Supposons que $M^{k-1}[i][j]$ représente la probabilité d'aller de i à j en $k-1$ clics. M^k est alors le produit de M et M^{k-1} , ce qui signifie que chaque élément $M^k[i][j]$ est la somme des probabilités de passer de i à un certain intermédiaire m en un clic (selon $M[i][m]$) et puis d'aller de m à j en $k-1$ clics (selon $M^{k-1}[m][j]$). Cela reflète la nature cumulative des probabilités dans les marches aléatoires.

1.2 Démonstration formelle par récurrence

Nous cherchons à démontrer que pour une matrice de transition M d'une marche aléatoire sur le graphe du web, $M^k[i][j]$ représente la probabilité d'arriver sur la page j en k clics à partir de la page i .

Rappel En Mathématique, démontrer qu'une proposition $P(n)$ est vraie pour tout $n \in \mathbb{N}$ signifie démontrer que $P(1)$ est vrai quelle que soit la valeur de n . Dans ce cas, une preuve par récurrence sur k est appropriée.

1.2.1 Notations et Déclaration des Variables

- S : Ensemble des sommets du graphe G .
- $M \in \mathbb{R}^{|S| \times |S|}$: Matrice de transition, où chaque élément $M[i][j]$ représente la probabilité de passer de la page i à la page j .
- $i, j \in \{1, 2, \dots, |S|\}$: Indices représentant des pages spécifiques dans le graphe, où $|S|$ est le nombre total de pages ou sommets dans le graphe.
- $k \in \mathbb{N}$: Le nombre de clics (ou étapes) dans la marche.
- $m \in \{1, 2, \dots, |S|\}$: Un sommet intermédiaire dans le graphe lors de la marche.
- S : L'ensemble des pages Web ou sommets du graphe. $S \in \mathbb{N}$ et représente le nombre total de sommets.
- $d^+(i) \in \mathbb{N}$: Le demi-degré extérieur du sommet i , c'est-à-dire le nombre de liens sortants de i dans un graphe orienté.
- $P(n)$: La proposition que nous cherchons à démontrer. Spécifiquement, $P(n)$ stipule que pour tout $n \in \mathbb{N}$ et pour tout couple de sommets $(i, j) \in S \times S$, $M^n[i][j]$ donne la probabilité d'une marche de longueur n de i à j .

1.2.2 Preuve par Induction sur k

Proposition $P(n)$: Pour tout $n \geq 1$ et tout couple de sommets $(i, j) \in S \times S$, $M^n[i][j]$ est égal à la probabilité d'arriver sur la page j en n clics à partir de la page i .

$P(n) : \forall n \in \mathbb{N}^*, \forall i, j \in \{1, 2, \dots, |S|\}, M^n[i][j] = \text{"probabilité d'une marche de } i \text{ à } j \text{ de longueur } n\text{"}$.

1.2.3 Initialisation : $P(1)$

Par définition :

- Cas 1 : $M[i][j] = \frac{1}{d^+(i)}$ si $(i, j) \in A$.
- Cas 2 : $M[i][j] = 0$ si $(i, j) \notin A$.

Dans les deux cas, $M^1[i][j]$ correspond aux probabilités définies pour un clic unique, donc $P(1)$ est vrai.

1.2.4 Récurrence : $P(k) \rightarrow P(k+1)$

Supposons que $P(k)$ soit vrai, c'est-à-dire que $M^k[i][j]$ représente la probabilité d'aller de i à j en k clics.

Pour tout chemin de longueur $k+1$ de i à j , il doit passer par un intermédiaire m . Ainsi, la probabilité d'une marche de i à j en $k+1$ clics est la somme sur tous les sommets intermédiaires possibles m des produits des probabilités d'aller de i à m en k clics et de m à j en un clic.

$$M^{k+1}[i][j] = \sum_{m=1}^{|S|} M^k[i][m] \cdot M[m][j]$$

Cela correspond à la définition du produit de matrices (dot-product). Par hypothèse de récurrence, $M^k[i][m]$ est la probabilité d'une marche de longueur k de i à m , et $M[m][j]$ est la probabilité d'un clic de m à j . Ainsi, l'induction est complète et $P(n)$ est démontrée².

En Conclusion

Pour démontrer formellement que $M^k[i][j]$ est la probabilité d'arriver sur la page j en k clics à partir de la page i , vous devrez utiliser des arguments de probabilité et la définition des marches aléatoires, en plus d'une preuve par récurrence sur la puissance de la matrice. C'est un concept fondamental en théorie des graphes et dans l'analyse des algorithmes de marche aléatoire, comme celui de PageRank.

2 Exercice 2 : Première implémentation de PageRank

Implémentez l'algorithme PageRank sur un graphe représenté par une liste d'adjacence. Donnez le résultat de l'algorithme sur le graphe G_1 fourni dans le fichier `res/example_1.txt` pour $k = 4$.

Dans cet exercice, nous allons implémenter l'algorithme PageRank sur un graphe représenté par une liste d'adjacence. L'implémentation C de cette liste d'adjacence est fournie.

L'étape la plus cruciale de l'algorithme PageRank est bien sûr le calcul d'une rangée de score (ou vecteur de probabilités de transition à l'étape k). Ce calcul se base sur la cardinalité des liens sortants du sommet considéré, ainsi que sur la valeur du score de ce sommet à l'étape $k-1$.

L'implémentation de cet algorithme donne ainsi le résultat suivant pour le graphe G_1 fourni :

2. Adapté d'une preuve consultable sur math.stackexchange.com [2]

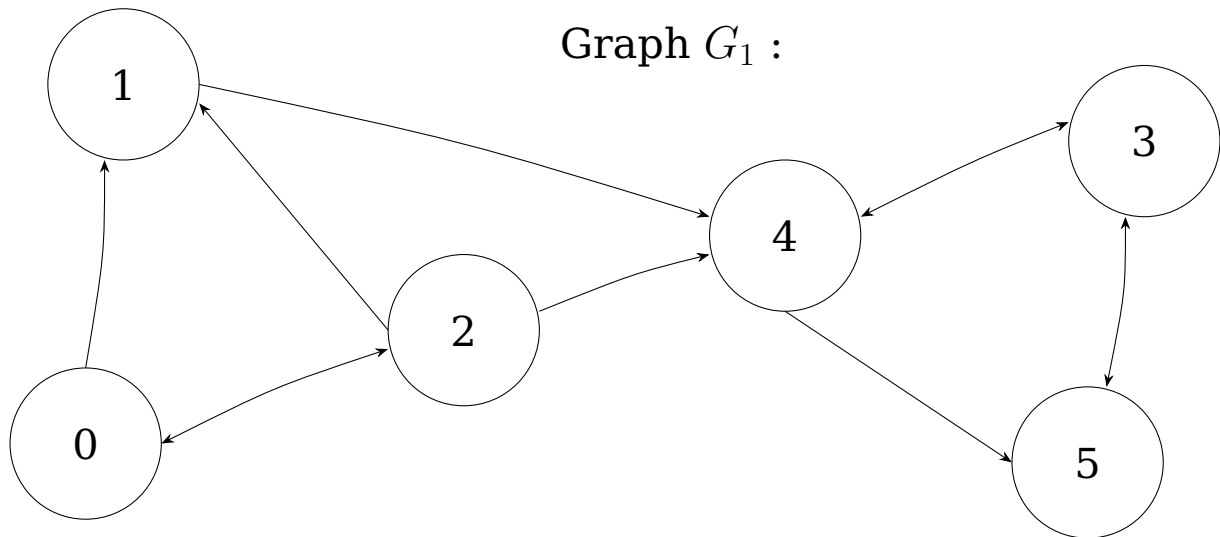


Figure 1 – Représentation du graphe G_1 fourni dans le fichier res/example_1.txt

1	Scores at step 0: [0.166667, 0.166667, 0.166667, 0.166667, 0.166667, 0.166667]
2	Scores at step 1: [0.055556, 0.138889, 0.083333, 0.250000, 0.305556, 0.166667]
3	Scores at step 2: [0.027778, 0.055556, 0.027778, 0.319444, 0.291667, 0.277778]
4	Scores at step 3: [0.009259, 0.023148, 0.013889, 0.423611, 0.224537, 0.305556]
5	Scores at step 4: [0.004630, 0.009259, 0.004630, 0.417824, 0.239583, 0.324074]

Code 1 – Résultat de l'algorithme PageRank sur le graphe G_1 fourni, pour $k = 4$.

Le résultat est donc, pour $k = 4$: [0.004630, 0.009259, 0.004630, 0.417824, 0.239583, 0.324074]. Le sommet 3 est donc le plus important, suivi des sommets 5 et 4 à l'issue de l'étape 4.

3 Exercice 3 : Problème des nœuds absorbants

Exécutez l'algorithme PageRank sur le graphe G_2 fourni dans le fichier res/example_2.txt pour $k = 4$. Calculez la somme des valeurs du vecteur s_k à chaque itération k . Qu'observe-t-on ? Comment évolue la somme des valeurs du vecteur s_k lorsque k augmente ?

Le graphe G_2 fourni dans le fichier res/example_2.txt est représenté dans la figure 2. Il s'agit du graphe G_1 dans lequel on a retiré le lien $1 \rightarrow 4$.

On relance l'algorithme PageRank sur le graphe G_2 . On obtient le résultat suivant :

1	S0: [0.166667, 0.166667, 0.166667, 0.166667, 0.166667, 0.166667] (sum: 1.000000)
2	S1: [0.055556, 0.138889, 0.083333, 0.250000, 0.138889, 0.166667] (sum: 0.833333)
3	S2: [0.027778, 0.055556, 0.027778, 0.236111, 0.152778, 0.194444] (sum: 0.694444)
4	S3: [0.009259, 0.023148, 0.013889, 0.270833, 0.127315, 0.194444] (sum: 0.638889)
5	S4: [0.004630, 0.009259, 0.004630, 0.258102, 0.140046, 0.199074] (sum: 0.615741)

Code 2 – Résultat de l'algorithme PageRank sur le graphe G_2 fourni, pour $k = 4$, avec la somme des valeurs du vecteur s_k à chaque itération k .

On remarque que la somme du vecteur s_k diminue à chaque itération. Hors, ce vecteur est censé donner la probabilité de se trouver sur chaque sommet du graphe à l'étape k . La somme de ces probabilités devrait donc être égale à 1.

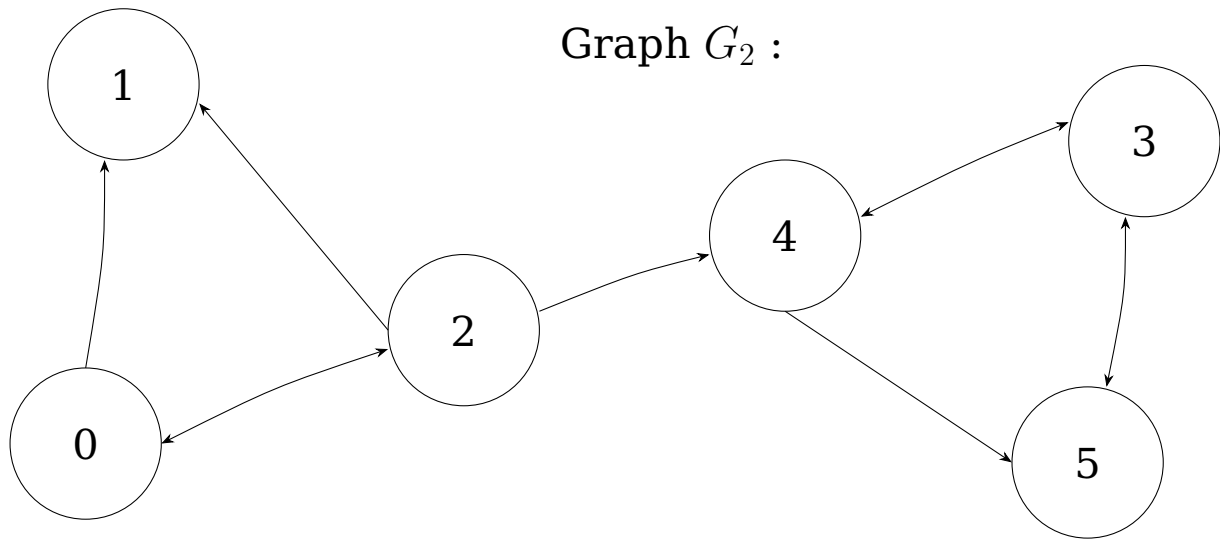


Figure 2 – Représentation du graphe G_2 fourni dans le fichier res/example_2.txt

4 Exercice 4 : Modification de PageRank pour les nœuds absorbants

On définit un sommet comme étant *absorbant* s'il ne possède aucun lien sortant, c'est-à-dire si son demi-degré $d^+(i)$ extérieur est égal à 0. Dans le graphe G_2 fourni, le sommet 1 est absorbant.

Dans ce cas, l'algorithme PageRank actuel ne fonctionne pas correctement. En effet, la somme des valeurs du vecteur s_k à chaque itération k n'est pas forcément égale à 1. Cela est dû au fait que la matrice de transition M n'est pas stochastique, c'est-à-dire que la somme des valeurs de chaque ligne n'est pas égale à 1. Ainsi, lors du calcul des valeurs de s_k , la somme des valeurs du vecteur s_{k-1} n'est pas forcément égale à 1 non-plus.

Pour corriger ça, on va modifier la matrice de transition M pour la rendre stochastique. Pour cela, on va ajouter une probabilité de transition de chaque sommet vers tous les autres sommets du graphe. Ainsi, la somme des valeurs de chaque ligne de la matrice sera égale à 1. Le fait d'ajouter ces connexions virtuelles vers tous les autres sommets conserve le fait que le sommet absorbant ne discrimine aucun autre sommet. En effet, dans le cas de PageRank, un sommet qui n'est connecté à aucun autre sommet ou qui est connecté à tous les autres sommets du graphe est équivalent puisqu'il ne discrimine aucun autre sommet par rapport à un autre dans la marche aléatoire.

Après modification de l'algorithme, on obtient les résultats suivants :

1	S0: [0.166667, 0.166667, 0.166667, 0.166667, 0.166667, 0.166667] (sum: 1.000000)
2	S1: [0.083333, 0.166667, 0.111111, 0.277778, 0.166667, 0.194444] (sum: 1.000000)
3	S2: [0.064815, 0.106481, 0.069444, 0.305556, 0.203704, 0.250000] (sum: 1.000000)
4	S3: [0.040895, 0.073302, 0.050154, 0.369599, 0.193673, 0.272377] (sum: 1.000000)
5	S4: [0.028935, 0.049383, 0.032665, 0.381430, 0.213735, 0.293853] (sum: 1.000000)

Code 3 – Résultat de l'algorithme PageRank sur le graphe G_2 fourni, pour $k = 4$, avec la somme des valeurs du vecteur s_k à chaque itération k .

Pour s_4 , on obtient ainsi : [0.028935, 0.049383, 0.032665, 0.381430, 0.213735, 0.293853]. On obtient donc bien une somme de 1 à chaque itération k . On remarque également que les valeurs de s_k sont

différentes de celles obtenues précédemment. En effet, le sommet 1 est maintenant connecté à tous les autres sommets du graphe, ce qui fait qu'il est moins discriminant que les autres sommets. Ainsi, les valeurs de s_k sont plus équilibrées entre les différents sommets du graphe.

5 Exercice 5 : Matrice ergodique et convergence

5.1 Convergence de l'algorithme PageRank

Lorsque l'on observe les valeurs de s_k obtenues à chaque itération k , on remarque que ces valeurs convergent vers une valeur stable. En effet, à partir de l'itération $k = C | C \in \mathbb{R}^*$, les valeurs de s_k ne changent plus pour $k \geq C$.

On remarque également que les scores de certains sommets ont convergé vers 0. On peut expliquer ce phénomène par le fait que le graphe en entrée contient des cycles absorbants (c'est-à-dire des cycles dans lesquels on ne peut pas sortir). Ainsi, les sommets qui ne sont pas dans ces cycles absorbants ne peuvent plus être atteints par la marche aléatoire après un certain nombre d'étapes. Ainsi, leur score converge vers 0.

Une matrice stochastique est dite *ergodique* si, pour deux sommets quelconques du graphe associé, la probabilité de rejoindre le second en partant du premier en un nombre fini d'étapes n'est pas nulle.

5.2 Matrice ergodique

Pour qu'une matrice de transition stochastique M' associée à un graphe G soit ergodique, le graphe doit satisfaire à une condition nécessaire et suffisante. Spécifiquement :

1. **Irréductibilité** : Un graphe est dit irréductible si pour toute paire de sommets i et j dans G , il existe un chemin de i à j . Formellement, pour tous les sommets $i, j \in G$, il existe une séquence de sommets commençant par i et se terminant par j où chaque paire consécutive est connectée par un arc. Cela garantit que le graphe est fortement connexe donc qu'il n'y a pas de sous-ensemble isolé de sommets.
2. **Apériodicité** : Un graphe est dit apériodique si le plus grand commun diviseur de la longueur de tous les cycles du graphe est 1. Autrement dit, il n'existe pas de nombre entier $k > 1$ tel que toutes les marches retournant à un sommet donné ont une longueur qui est un multiple de k . Cela garantit qu'il n'y a pas de comportement cyclique régulier dans les marches aléatoires sur le graphe.

Lorsqu'un graphe est à la fois irréductible et apériodique, la matrice de transition stochastique associée à ce graphe est dite ergodique. Cela signifie que, indépendamment de l'état initial, la marche aléatoire sur le graphe convergera vers une distribution stationnaire unique. En termes mathématiques, la matrice de transition stochastique M' aura une unique distribution de probabilité limite vers laquelle elle converge, peu importe la distribution de départ.

Ces propriétés garantissent que la marche aléatoire sur le graphe se stabilisera sur une distribution qui ne dépend pas de l'état initial, ce qui est crucial dans de nombreuses applications telles que l'algorithme de PageRank où l'on souhaite que la probabilité de visiter les pages convergent vers une distribution unique reflétant leur "importance" ou leur "rang" dans le réseau.

5.3 Démonstration de l'ergodicité et de la stochasticité de la matrice de transition M''

5.3.1 Notations et Déclaration des Variables

Nous avons déjà défini une partie des notations dans une section précédente (voir 1.2.1). Voici les notations supplémentaires utilisées dans cette section :

- $M \in \mathbb{R}^{|S| \times |S|}$: Matrice de transition originale du graphe G .
- $M' \in \mathbb{R}^{|S| \times |S|}$: Matrice de transition stochastique modifiée à partir de M .
- $M'' \in \mathbb{R}^{|S| \times |S|}$: Matrice de transition ergodique et stochastique modifiée à partir de M' .
- $p = |A|$: Nombre total d'arcs dans G .
- $n = |S|$: Nombre total de sommets dans G .
- $d^+(i) \in \mathbb{N}$: Degré sortant du sommet i .
- $\alpha \in (0, 1)$: Paramètre du surfeur aléatoire utilisé dans la modification de M' à M'' .
- $\vec{s}_k \in \mathbb{R}^{|S|}$: Vecteur de probabilités à l'étape k .
- q_{abs} : Quantité redistribuée par les sommets absorbants à chaque itération.

5.3.2 Définitions

Soit M la matrice de transition d'un graphe G . Nous transformons M en une matrice stochastique M' où les sommets absorbants redistribuent la même importance à chaque sommet du graphe. Cette transformation est réalisée en s'assurant que chaque sommet a au moins un lien sortant. La quantité redistribuée par les sommets absorbants à chaque itération est calculée par la formule suivante :

$$q_{abs} = \frac{1}{n} \cdot \sum_{\{i \in S | d^+(i)=0\}} \vec{s}_k[i]$$

Ensuite, pour rendre M' ergodique, nous introduisons la matrice M'' en incorporant la métaphore du surfeur aléatoire. Pour chaque couple de sommets $(i, j) \in S \times S$:

$$M''[i][j] = \begin{cases} \alpha \cdot M'[i][j] + \frac{1-\alpha}{n} & \text{si } (i, j) \in A \\ \frac{1-\alpha}{n} & \text{si } (i, j) \notin A \end{cases}$$

5.3.3 Démonstration

Stochasticité de M'' : Nous devons montrer que M'' est une matrice stochastique, c'est-à-dire que chaque ligne de M'' somme à 1. Pour chaque sommet $i \in S$, la somme des probabilités de transition vers tous les autres sommets j est :

$$\sum_{j=1}^{|S|} M''[i][j] = \alpha \sum_{j=1}^{|S|} M'[i][j] + (1 - \alpha) = \alpha + (1 - \alpha) = 1$$

Cela est vrai, car M' est déjà stochastique, et donc la somme des probabilités sur toutes les destinations j pour un sommet donné i est 1. L'ajout du terme $(1 - \alpha)$ réparti uniformément entre tous les sommets assure que chaque ligne de M'' somme également à 1.

Ergodicité de M'' : Pour démontrer que M'' est ergodique, nous devons montrer que le graphe est irréductible et apériodique.

- *Irréductibilité* : Par la définition de M'' , même si un sommet était initialement absorbant (aucun lien sortant), il a maintenant une probabilité $(1 - \alpha)/n$ de se connecter à chaque autre sommet, garantissant qu'il y a toujours un chemin de n'importe quel sommet i à n'importe quel sommet j . Ainsi, le graphe associé à M'' est irréductible.
- *Apériodicité* : Supposons par l'absurde qu'il existe un entier $k > 1$ qui soit le plus grand commun diviseur des longueurs de tous les cycles dans le graphe. Cela signifierait que chaque cycle dans le graphe a une longueur qui est un multiple de k . Considérons un cycle de longueur k et un sommet i dans ce cycle. Dans M'' , chaque sommet, y compris i , a une probabilité non nulle de transitionner vers lui-même en une étape, grâce à la téléportation du surfeur aléatoire ($\frac{1-\alpha}{n}$). Cela introduit un cycle de longueur 1 (boucle i vers i) pour chaque sommet, en contradiction avec notre supposition initiale que tous les cycles ont une longueur qui est un multiple de $k > 1$. Par conséquent, notre supposition doit être fausse et le plus grand commun diviseur des longueurs des cycles doit être 1. Ainsi, le graphe associé à M'' est apériodique.

En combinant ces deux propriétés, nous concluons que M'' est à la fois stochastique et ergodique. Cela garantit la convergence de l'algorithme PageRank vers une distribution stationnaire unique, indépendamment de l'état initial.

6 Exercice 6 : PageRank avec téléportation (surfeur aléatoire)

Dans cet exercice, nous allons implémenter l'algorithme PageRank avec téléportation (surfeur aléatoire) sur un graphe représenté par une liste d'adjacence. Cela revient à faire les modifications de la matrice de transition M pour la rendre ergodique et stochastique, comme nous l'avons vu dans l'exercice précédent.

Pour cela, on remarque qu'à chaque itération k , la valeur minimale d'un élément de s_k est $(\alpha \times q_{abs}) + \frac{1-\alpha}{n}$. Cela correspond à sortir le terme q_{abs} dans le terme $M'[i][j]$. Il ne reste plus qu'à propager les valeurs des successeurs de chaque sommet pondéré par α .

Après modification de l'algorithme, on obtient les résultats suivants :

1	S0: [0.166667, 0.166667, 0.166667, 0.166667, 0.166667, 0.166667] (sum: 1.000000)
2	S1: [0.091667, 0.166667, 0.116667, 0.266667, 0.166667, 0.191667] (sum: 1.000000)
3	S2: [0.076667, 0.117917, 0.082917, 0.289167, 0.196667, 0.236667] (sum: 1.000000)
4	S3: [0.059229, 0.093729, 0.068854, 0.335854, 0.189354, 0.252979] (sum: 1.000000)
5	S4: [0.051382, 0.078035, 0.057379, 0.343617, 0.202517, 0.267070] (sum: 1.000000)

Code 4 – Résultat de l'algorithme PageRank sur le graphe G_2 fourni, pour $k = 4$, avec la somme des valeurs du vecteur s_k à chaque itération k , pour PageRank modifié avec téléportation et $\alpha = 0.9$.

On obtient pour s_4 , $\alpha = 0.9$: [0.051382, 0.078035, 0.057379, 0.343617, 0.202517, 0.267070].

7 Exercice 7 : Calcul de rangs après convergence

Calculez les rangs des sommets du graphe G_2 fourni dans le fichier `res/example_2.txt` après convergence de l'algorithme PageRank.

On modifie l'algorithme pour qu'il s'arrête lorsque la différence entre les valeurs de s_k et s_{k-1} est inférieure à 10^{-6} . Plutôt que de vérifier que toutes les valeurs sont bien inférieures en valeur absolue à la valeur limite ϵ , on vérifie si au moins une valeur est supérieure en valeur absolue à ϵ . Cela permet d'éviter de parcourir toutes les valeurs du vecteur s_k à chaque itération.

Après modification de l'algorithme, on obtient les résultats suivants :

```
1      Convergence reached: stopping at step 43
2      Last scores provided at step 42
3      +----- Ranks [k=42] -----
4      + Rank 1: node 3 (score: 0.375081)
5      + Rank 2: node 5 (score: 0.286246)
6      + Rank 3: node 4 (score: 0.205998)
7      + Rank 4: node 1 (score: 0.053957)
8      + Rank 5: node 2 (score: 0.041506)
9      + Rank 6: node 0 (score: 0.037212)
```

Code 5 – Résultat de l'algorithme PageRank après convergence sur le graphe G_2 , avec $\alpha = 0.9$ et $\epsilon = 1.0 \times 10^{-10}$.

Retester l'algorithme sur le gros graphe $G_{genetic}$. Donnez les 5 sommets les plus importants une fois la convergence atteinte.

On réeffectue le même test sur le graphe $G_{genetic}$ fourni dans le fichier `res/genetic_network.txt`. On obtient les résultats suivants :

```
1      Convergence reached: stopping at step 161
2      Last scores provided at step 160
3      +----- Ranks [k=160] -----
4      + Rank 1: node 491 (score: 0.015354)
5      + Rank 2: node 492 (score: 0.015227)
6      + Rank 3: node 2790 (score: 0.014818)
7      + Rank 4: node 1182 (score: 0.014485)
8      + Rank 5: node 1188 (score: 0.014446)
```

Code 6 – Résultat de l'algorithme PageRank après convergence sur le graphe G_2 , avec $\alpha = 0.9$ et $\epsilon = 1.0 \times 10^{-10}$.

Les 5 meilleurs sommets sont donc les sommets 491, 492, 2790, 1182 et 1188, une fois la convergence atteinte après 160 itérations.

Acronymes

AAIA Algorithmique Avancée pour l'Intelligence Artificielle et les graphes. 2

Références

- [1] Christine Solnon. *Première partie : Algorithmique avancée pour les graphes*. AAIA. CITI Lab, INSA Lyon dépt. Informatique. 2016. url : <http://perso.citi-lab.fr/csolnon/supportAlgoGraphes.pdf> (visité le 28/12/2023).
- [2] Simón Ramírez Amaya et BMBM. *Proof - raising adjacency matrix to n -th power gives n -length walks between two vertices*. 2017. url : <https://math.stackexchange.com/questions/2434064/proof-raising-adjacency-matrix-to-n-th-power-gives-n-length-walks-between> (visité le 28/12/2023).