Bases de données avancées

Recherche d'information

Auteurs : Raphaël Fournier-S'niehotta, Nicolas Travers, Philippe Rigaux
fournier@cnam.fr, nicolas.travers@cnam.fr, philippe.rigaux@cnam.fr

Département d'informatique
Conservatoire National des Arts & Métiers, Paris, France

Plan

Lecture des pages Web

Plan du cours

Classement par pertinence

Les requêtes Booléennes classent les documents en deux catégories : ceux qui satisfont la requête, et les autres. C'est 1, ou c'est 0.

Le classement par pertinence (*ranked search*) trie un résultat en fonction d'un "poids" (*weight*) mesurant le degré de pertinence d'un document pour une recherche.

Plusieurs approches complémentaires pour évaluer la pertinence d'un document d pour une recherche q :

- par similarité entre d et q ;
- par l'importance de d, évaluée par exemple par sa position dans une collection structurée (e.g., graphe, cf. PageRank) ;
- en prenant en compte l'utilisateur (profil, boucles de pertinence, etc.). Cf. l'analyse des clics !

Recherche par similarité

La requête q et le document d sont placés dans un même Espace métrique, doté d'une fonction de distance m_E.

Une fonction f produit un descripteur (le plus souvent sous la forme d'un vecteur appelé *features vector*) à partir d'un document.

La similarité, ou score, est l'inverse de la distance.

$$sim(q,d) = \frac{1}{m_E(f(q),f(d))}$$

Langage de requêtes ultra-simplifié

La requête est exprimée par un ensemble de mots-clés, et interprétée comme un "document" simplifié.

Quelques caractéristiques

Avec l'approche par similarité, le résultat (c.à.d. les documents qui ont un score non nul) est potentiellement très grand.

Il est impératif de classer le résultat par ordre de score croissant, et de présenter les k premiers à l'utilisateur (typ., $k \simeq 10 - 20$).

Souvenez-vous: rappel et précision

- La précision est la fraction du résultat qui est vraiment pertinente.

$$precision = \frac{|relevant| \cap |retrieved|}{|retrieved|}$$

- Le rappel est la fraction des documents pertinents présente dans le résultat.

$$recall = \frac{|relevant| \cap |retrieved|}{|relevant|}$$

Vocabulaire usuel

On place documents et requête dans un espace métrique, souvent un espace vectoriel.

L'essentiel: une fonction de similarité, définie à partir d'une distance, ou directement.

Les éléments de cet espace sont appelés descripteurs ou vecteurs de caractéristiques (*features vector*).

Le score $sim(q, d)$ mesure la pertinence d'un document d par rapport à une requête (besoin) q.

Le résultat est classé sur le score; les k premiers documents sont présentés.

Plan du cours

Soyons concrets: première approche pour la recherche plein texte

Supposons connu l'ensemble de tous les termes $V = \{t_1, t_2, \cdots, t_n\}$ de tous les termes utilisables pour la rédaction d'un document.

Exemple: $V =$ {"papa", "maman", "gateau", "chocolat", "haut", "bas"}

On définit $E = \{0,1\}^n$ comme l'espace de tous les vecteurs constitués de n coordonnées valant soit 0, soit 1. Ce sont nos descripteurs.

Exemple: vecteurs constitués de 6 coordonnées valant soit 0, soit 1.

Fonction f associant un document d à son descripteur $v = f(d)$.

$$v[i] = \begin{cases} 1 \text{ si } d \text{ contient le terme } t_i \\ 0 \text{ sinon} \end{cases}$$

Jusque là, rien de nouveau

C'est la représentation déjà vue pour les matrices d'incidences.

Exemple(s)

Rappel: on a $V = \{$"papa", "maman", "gateau", "chocolat", "haut", "bas"$\}$

Soit le document d_{maman}=`maman est en haut, qui fait du gateau`

alors $f(d_{maman}) = [0,1,1,0,1,0]$

Test

À vous de jouer: d_{papa}= `papa est en bas, qui fait du chocolat`

Deux remarques:

- On ignore certains mots (les "mots inutiles" ou stop words)
- L'ordre des mots dans le document est ignoré (approche "*bag of words*")

La distance

Dans un espace vectoriel, on peut penser à prendre la distance Euclidienne. Si v_1 et v_2 sont deux vecteurs:

$$E(v_1, v_2) = \sqrt{(v_1^1 - v_2^1)^2 + (v_1^2 - v_2^2)^2 + \cdots + (v_1^n - v_2^n)^2}$$

Et la similarité est l'inverse de la distance

$$sim(v_1, v_2) = \begin{cases} \infty \text{ si } v_1^i = v_2^i \text{ pour tout } i \\ \frac{1}{E(v_1, v_2)} \text{ sinon} \end{cases}$$

Exemple, pour $q =$ "maman haut chocolat", $v_q = [0,1,0,1,1,0]$

$$sim(v_q, d_{maman}) = \frac{1}{\sqrt{2}}$$

Test

À vous de calculer $sim(v_q, d_{papa})$

Quelques points à retenir

Important

Une différence concrète très sensible (illustrée ci-dessus) avec les requêtes Booléennes est qu'il n'est pas nécessaire qu'un document contienne tous les termes de la requête pour que son score soit différent de 0.

- "chocolat", un des mots-clés de q, n'apparaît pas dans le document d_{maman}, malgré tout classé en tête;

Approche présentée ici: simple mais nombreux inconvénients (exercices).

Ignore l'impact de : la taille des documents, la taille du vocabulaire, le nombre d'occurrences d'un terme dans un document et la rareté de ce terme dans la collection.

Exercices

Exercices, 1 à 4

Plan du cours

Pondération des termes

Le classement s'appuie sur l'idée qu'il est possible d'identifier l'importance des termes dans un document. Deux idées essentielles:

Plus un terme est fréquent dans un document, plus il est représentatif du contenu du document.

- Ex.: *contrepoint* apparaît 10 fois dans le document d, $\Rightarrow$ d est un document important pour une recherche sur "contrepoint".

- Normalisation : si d est très long, il est normal de trouver plus d'occurrences des termes ; on peut décider de *normaliser* cet indicateur pour éviter ce biais.

Plus un terme est rare dans la collection, plus sa présence dans un document est importante.

- Ex.: *musicologie* apparaît 100 fois dans une grande collection, 4 fois dans le document d $\Rightarrow$ d est un document important pour une recherche sur "musicologie".

Premier indicateur : la fréquence du terme

Soit un document d, t un terme.

La fréquence du terme t dans d, noté $n_{t,d}$, est simplement le nombre d'occurrences de t dans d.

Attention

On parle bien de termes, résultats de la normalisation lexicale (racinisation, élimination des mots inutiles, etc.)

Exemple : soit d le document suivant :

```
Spider Cochon Spider Cochon, il peut marcher au plafond,
Est ce qu'il peut faire une toile ? Bien sûr que non,
c'est un cochon. Prends garde ! Spider Cochon est là !
```

Donc $tf(\text{cochon}, \text{d}) = 4$

Deuxième indicateur : fréquence inverse des documents

Soit t un terme, une collection D. On mesure sa rareté de t par l'inverse de sa fréquence dans D.

- Le nombre total de documents est $|D|$
- le nombre de documents avec t est $|\{d' \in D, |, n_{t,d'} > 0\}|$

La rareté de t est donc mesurée par :

$$\frac{|D|}{|\{d' \in D \,|\, n_{t,d'} > 0\}|}$$

Ajustement. La valeur obtenu par la formule ci-dessus croît très vite avec la taille de la collection. On ajuste en prenant le logarithme et on obtient la fréquence inverse des documents (*inverse document frequency*, idf)

$$\mathrm{idf}(t) = \log \frac{|D|}{\left|\left\{d' \in D \,|\, n_{t,d'} > 0\right\}\right|}.$$

Pondération par TF-IDF

Le poids d'un terme dans un document est représenté par l'indicateur Term Frequency—Inverse Document Frequency (tf-idf)

$$\text{tfidf}(t,d) = n_{t,d} \cdot \log \frac{|D|}{\left|\left\{d' \in D \,|\, n_{t,d'} > 0\right\}\right|}$$

$n_{t,d}$	nombre d'occurrences de t dans d
D	ensemble de tous les documents

- le tf-idf décroît quand un terme est présent dans beaucoup de documents ;
- il décroît également quand il est peu présent dans un document ;
- il est maximal pour les termes peu fréquents apparaissant beaucoup dans un document particulier.

Descripteurs

Le tf.idf remplace l'indicateur 0/1 dans la matrice d'incidence.

Pondérons nos cochons

Voici des documents A, B et C.

```
Spider Cochon Spider Cochon, il peut marcher au plafond,
Est ce qu'il peut faire une toile ? Bien sûr que non,
c'est un cochon. Prends garde ! Spider Cochon est là !

Un petit cochon, pendu au plafond

Les Trois Petits Cochons est un conte traditionnel européen
mettant en scène trois jeunes cochons et un loup.
```

Test

Calculer le tf et l'idf pour les termes "cochon", "loup" et "plafond" et pour chaque document. En déduire la matrice d'incidence.

Plan du cours

Calcul de la similarité

À ce stade, on peut décrire un document par un vecteur composé des poids de tous ses termes.

$$\text{Descr(C)} = [\text{'cochon'}: 2, \text{'plafond'}: 0, \text{'loup'}: 4]$$

Le poids est à 0 pour les termes qui n'apparaissent pas dans le document. On a un espace vectoriel $E = \mathbb{N}^{|V|}$, V étant le vocabulaire.

$$[\text{'cochon'}: 0, ..., \text{'jaguar'}: 4, ..., \text{'loup'}: 0, ..., \text{'python'}: 2, ..., \text{'mouton'}: 0]$$

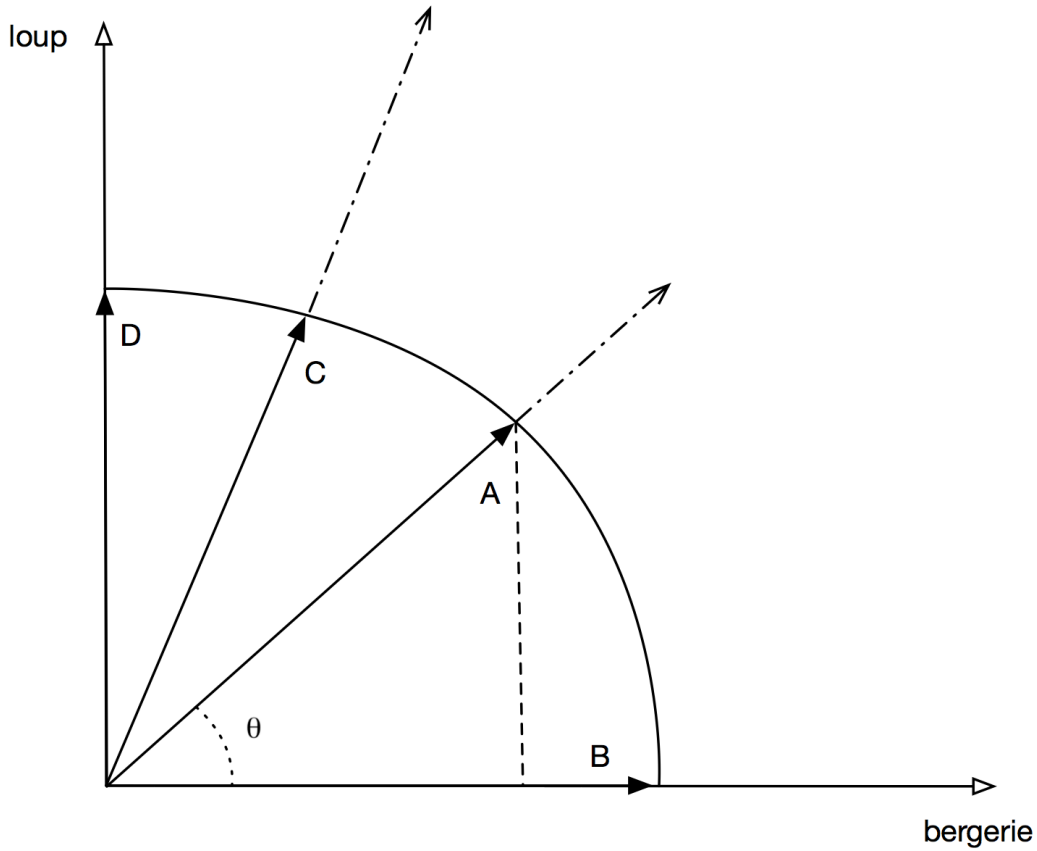
- L'espace a beaucoup de dimensions (des millions d'axes / termes).
- Chaque vecteur est principalement constitué de 0.

Distance Euclidienne?

Potentiellement très coûteuse à calculer, et introduit un biais lié à la longueur des documents.

Classement par cosinus

Plus deux documents sont “proches” l’un de l’autre, plus l’angle de leurs vecteurs descripteurs est petit.



Rappel : le cosinus est une fonction décroisante sur l’intervalle $[0, 90]$.

La similarité cosinus

La similarité cosinus est un bon candidat pour mesurer la proximité des vecteurs dans $E\mathbb{N}^{|V|}$

- Indifférent la la longueur (norme) des vecteurs.
- Maximal si même direction (angle $= 0$, cosinus $= 1$)
- Minimal si directions "orthogonales" (pas de terme en commun)
- Varie continuement entre 0 et 1.

En pratique

Calcul efficace car nécessite seulement les coordonnées non nulles.

Calcul du cosinus de deux vecteurs

Produit scalaire de deux vecteurs :

$$v_1.v_2 = ||v_1|| \times ||v_2|| \times cos\theta = \sum_{i=1}^{n} v_1[i] \times v_2[i]$$

où θ désigne l'angle entre les deux vecteurs et $||v||$ la norme d'un vecteur.

Donc:

$$cos\theta = \frac{\sum_{i=1}^{n} v_1[i] \times v_2[i]}{||v_1|| \times ||v_2||}$$

Normalisation

La division par la norme revient à éliminer le biais lié à la longueur des documents.

Calcul du cosinus, en pratique

Première étape : on calcule la norme du vecteur représentant chaque document, on la stocke.

Norme du vecteur $\vec{d}$:

$$||\vec{d}|| = \sqrt{\Sigma_i d_i^2}$$

Seconde étape : d le document, q la requête; on prend les vecteurs $\vec{q}$ (calculé à la volée) et $\vec{d}$ (stocké dans un index).

$$\frac{1}{||\vec{d}||} \times \frac{1}{||\vec{q}||} \times \Sigma_i d_i q_i$$

Approximation

On peut ignorer la norme de la requête (qui est constante), et la racine carrée pour la calcul des normes: c'est le classement qui nous intéresse.

Exercices

Exercices, 1 à 4

Plan du cours

PageRank : importance déduite du graphe des documents

Dans le cas du Web (et quelques autres systèmes), les documents sont liés par des hyperliens.

La structure de la collection est donc celle d'un graphe orienté.

Intuition

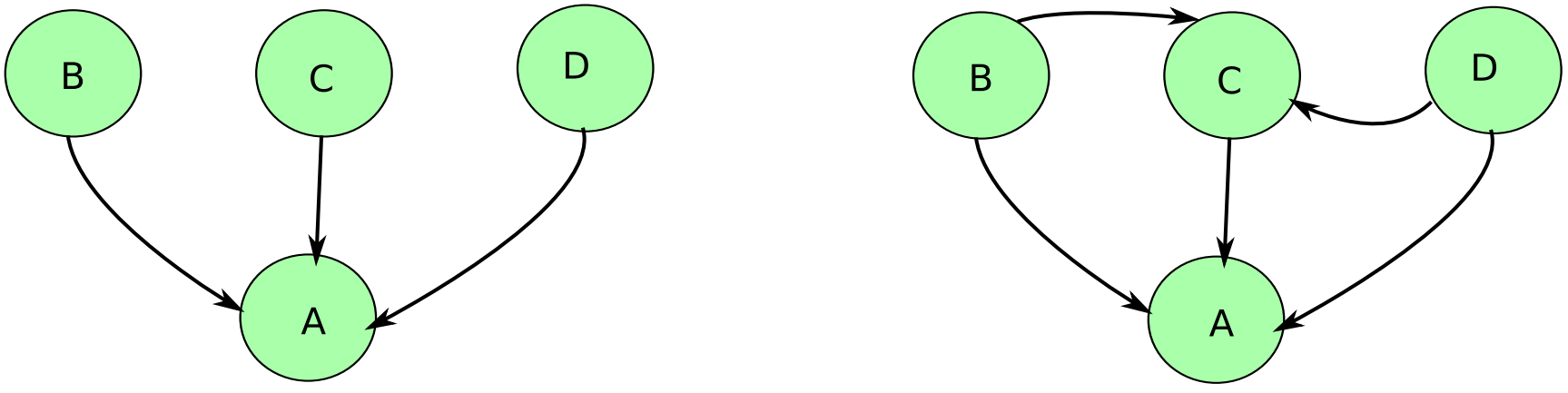
Un document vers lequel convergent beaucoup de chemins est un document important.

En combinant avec des mesures de pertinence (tf/idf), on obtient un moyen d'améliorer le classement.

PageRank : définition et exemples

Définition

L'indicateur *PageRank* (PR) d'une page p_i est la probabilité qu'un utilisateur suivant les liens de manière aléatoire arrive sur P_i.

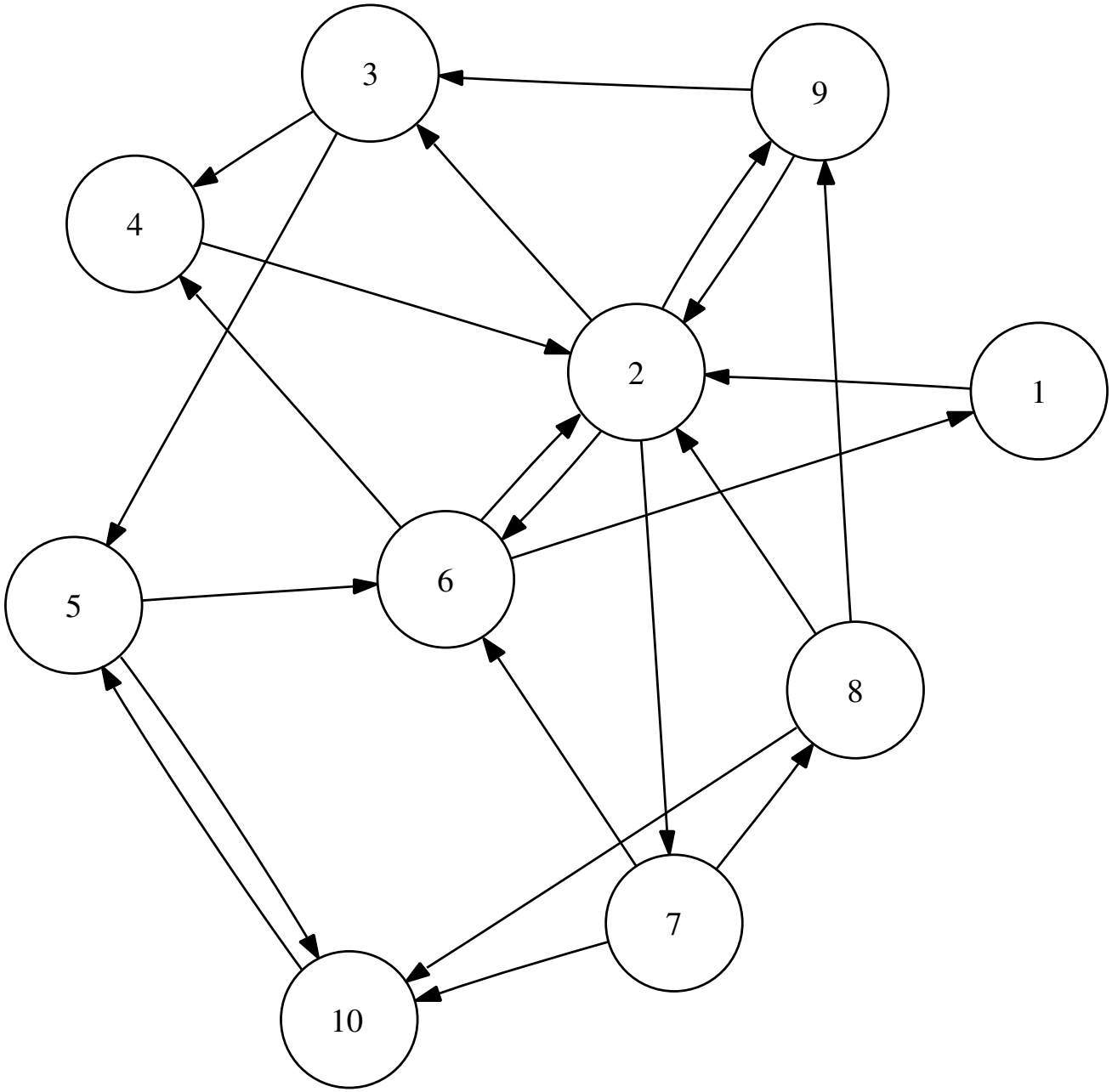


À gauche : la probabilité d'arriver en A en une étape est
$PR(A) = PR(B) + PR(C) + PR(D)$

À droite ?

Au départ, chaque page a un PR de 0,25. Quel est le PR après une itération ? Et après deux ?

Un exemple plus complet



On construit une matrice de transition

$$\begin{cases} g_{ij} = 0 & \text{s'il n'y a pas de lien entre les pages } i \text{ et } j; \\ g_{ij} = \frac{1}{n_i} & \text{sinon, } n_i \text{ étant le nombre de liens sortant de la page } i. \end{cases}$$

$$G = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{4} & 0 & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Calcul du PageRank, pas à pas

Je veux calculer la probablité d'être en N_2 à l'étape e. J'ai besoin :

- de la probabilité d'être sur chaque nœud N_i à l'étape $e-1$
  $\Rightarrow$ c'est le vecteur des PageRank, appelons-le v.

- de la probabilité d'arriver au nœud N_2 venant du nœud N_i
  $\Rightarrow$ c'est la seconde colonne de la matrice.

Allons-y. Au départ, le vecteur des PageRank est uniforme

$$v = [0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1]$$

La seconde colonne de la matrice, transposée est :

$$C_2 = [1, 0, 0, 1, 0, 1/3, 0, 1/3, 1/2, 0]$$

Ce qui donne la probabilité d'arriver en N_2 à la première itération

$$0.1 \times 1 + 0.1 \times 1 + 0.1 \times 1/3 + 0.1 \times 1/3 + 0.1 \times 1/2 = 0.317$$

Interprétation : j'ai 10% de chances d'être en N_1, 100% de chances, étant en N_1, d'aller en N_2, etc.

Calcul du PageRank, généralisé

On effectue le calcul précédent pour tous les nœuds, et autant de fois que nécessaire.

- On construit par itérations un vecteur contenant l'indicateur PR de chaque page du graphe.
  Appelons-le v ; il contient autant de coordonnées que de pages du Web...

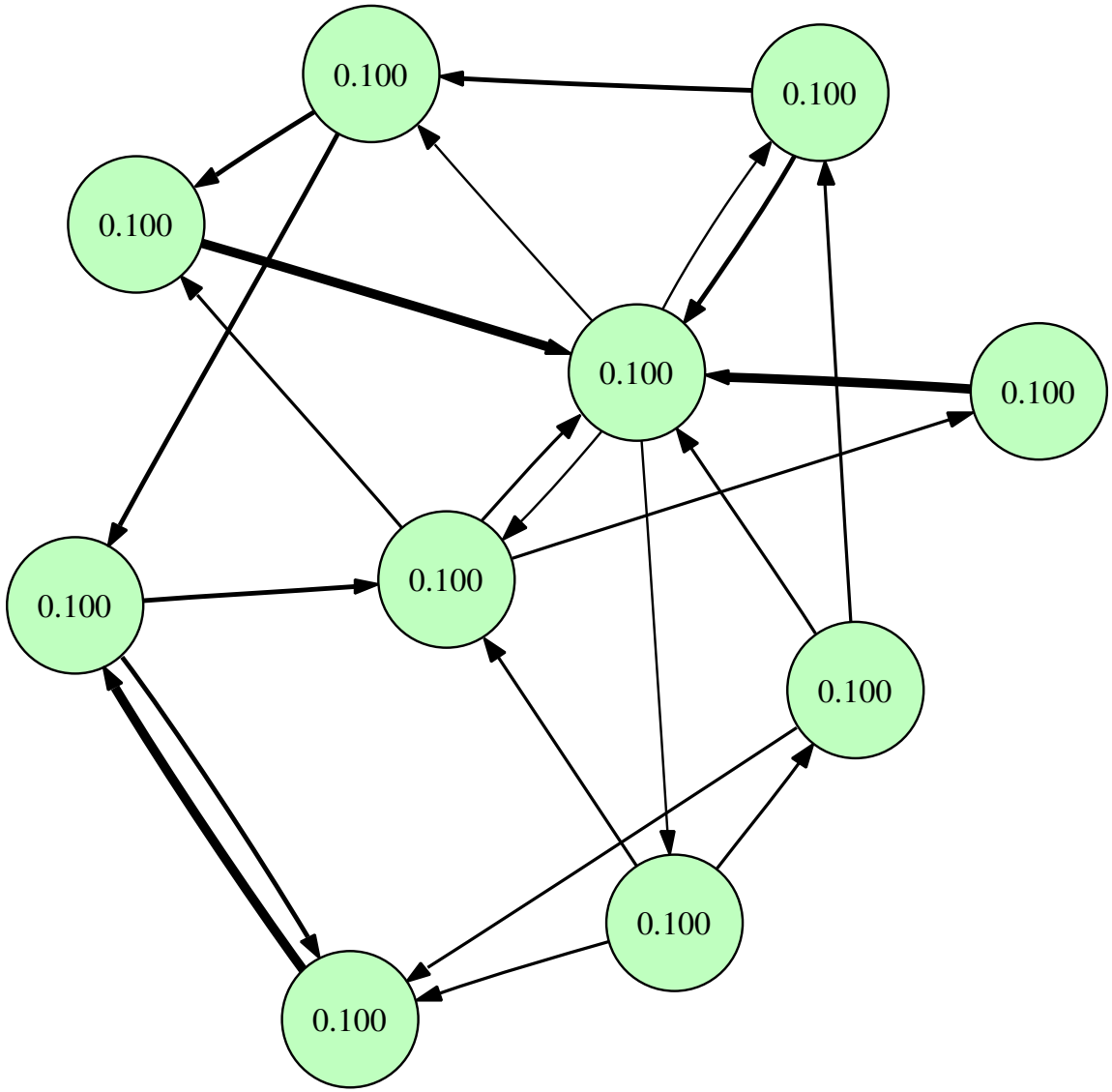
- v est initialisé avec une distribution uniforme ($v[i] = \frac{1}{|v|}$).
  Sur notre exemple, la valeur initiale est $1/10$.

- À chaque itération, on ajuste v on calculant la probabilité qu'un déplacement amène sur chaque nœud.
  On multiplie le vecteur v par la transposée de G (les colonnes donnent la probabilité d'arriver sur un nœud).

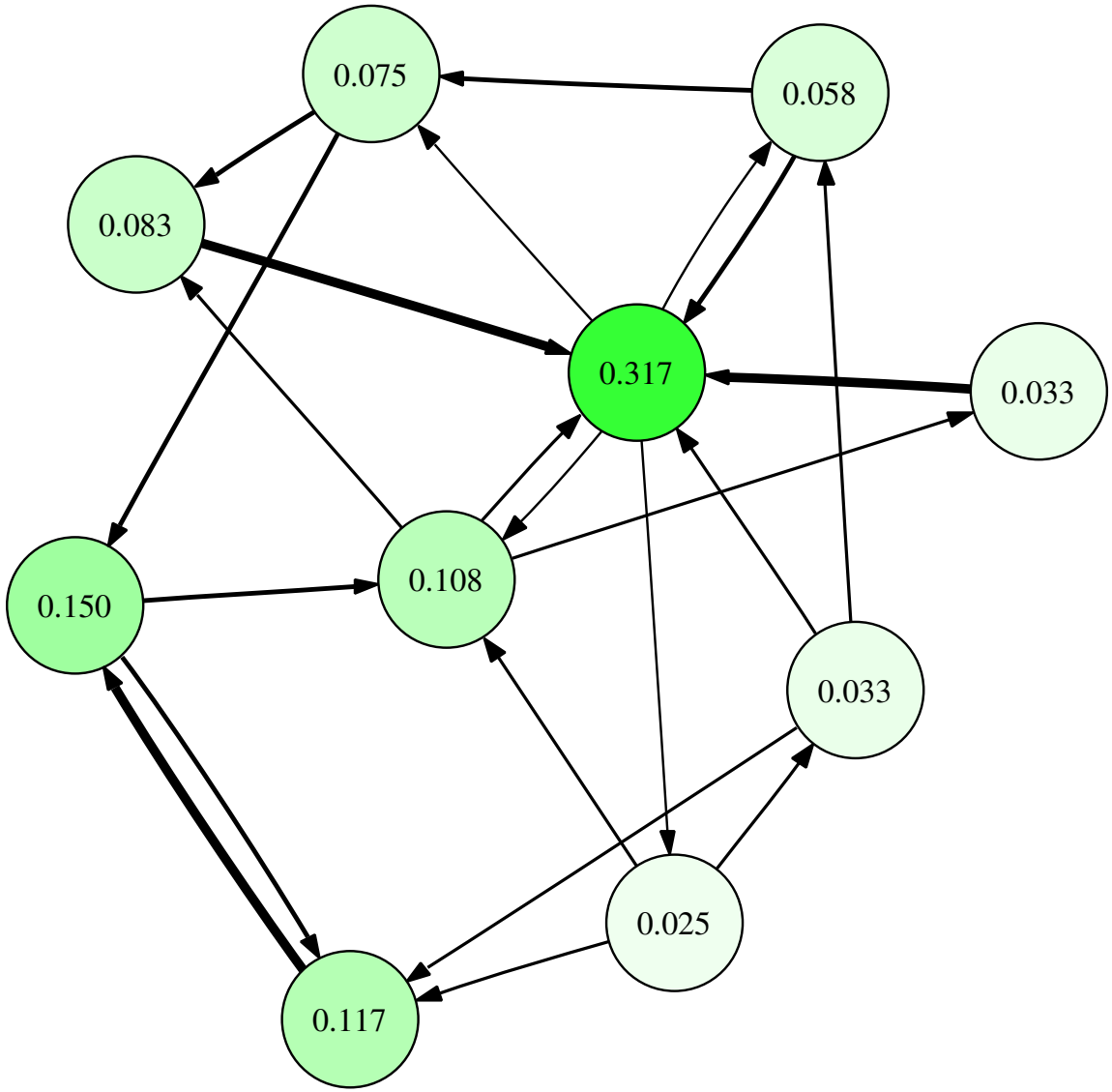
On peut montrer qu'il y a convergence du vecteur v vers une limite.

$$\mathrm{pr}(i) = \left(\lim_{k \to +\infty} (G^T)^k v \right)_i$$

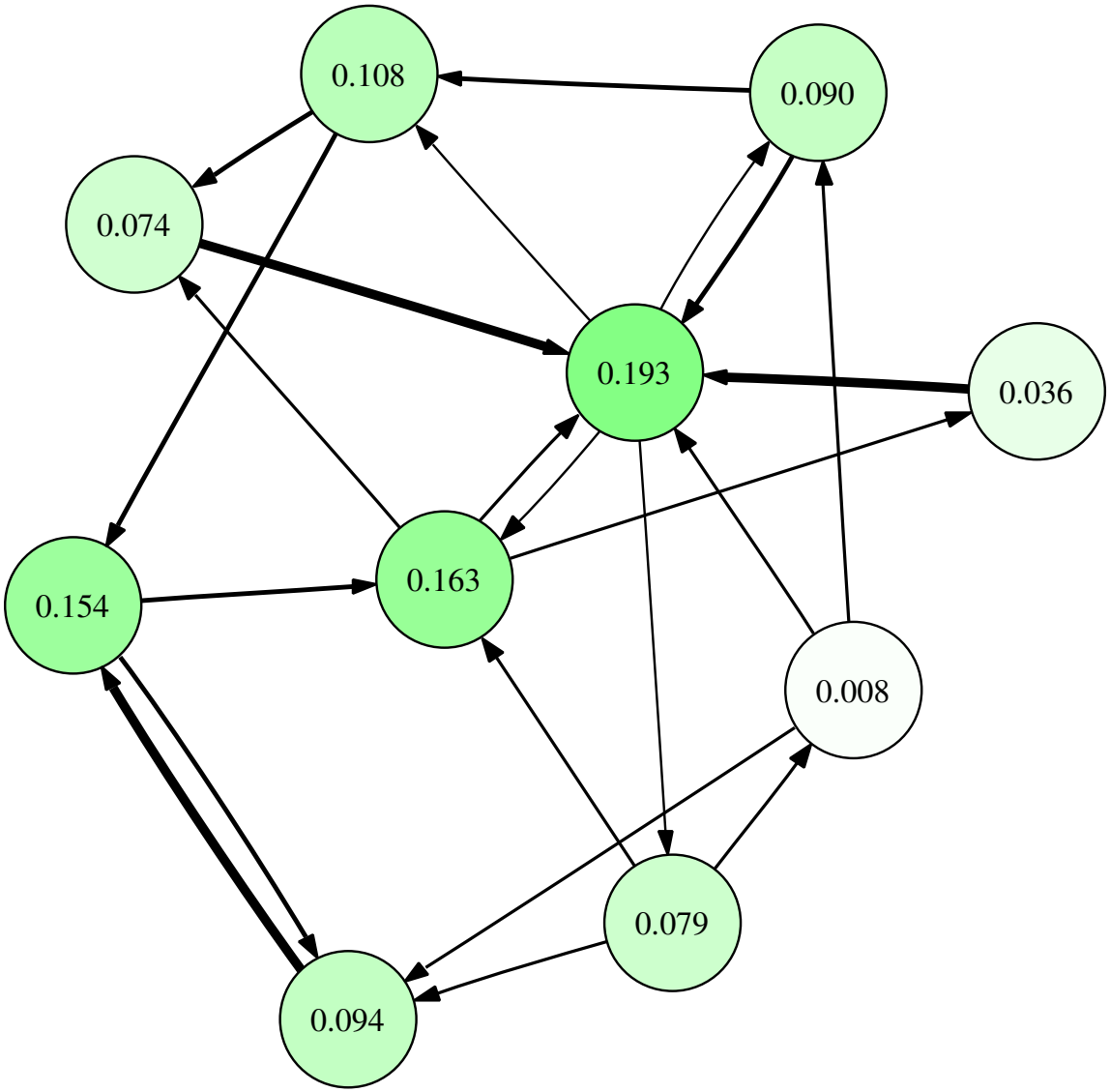
Queques itérations PageRank



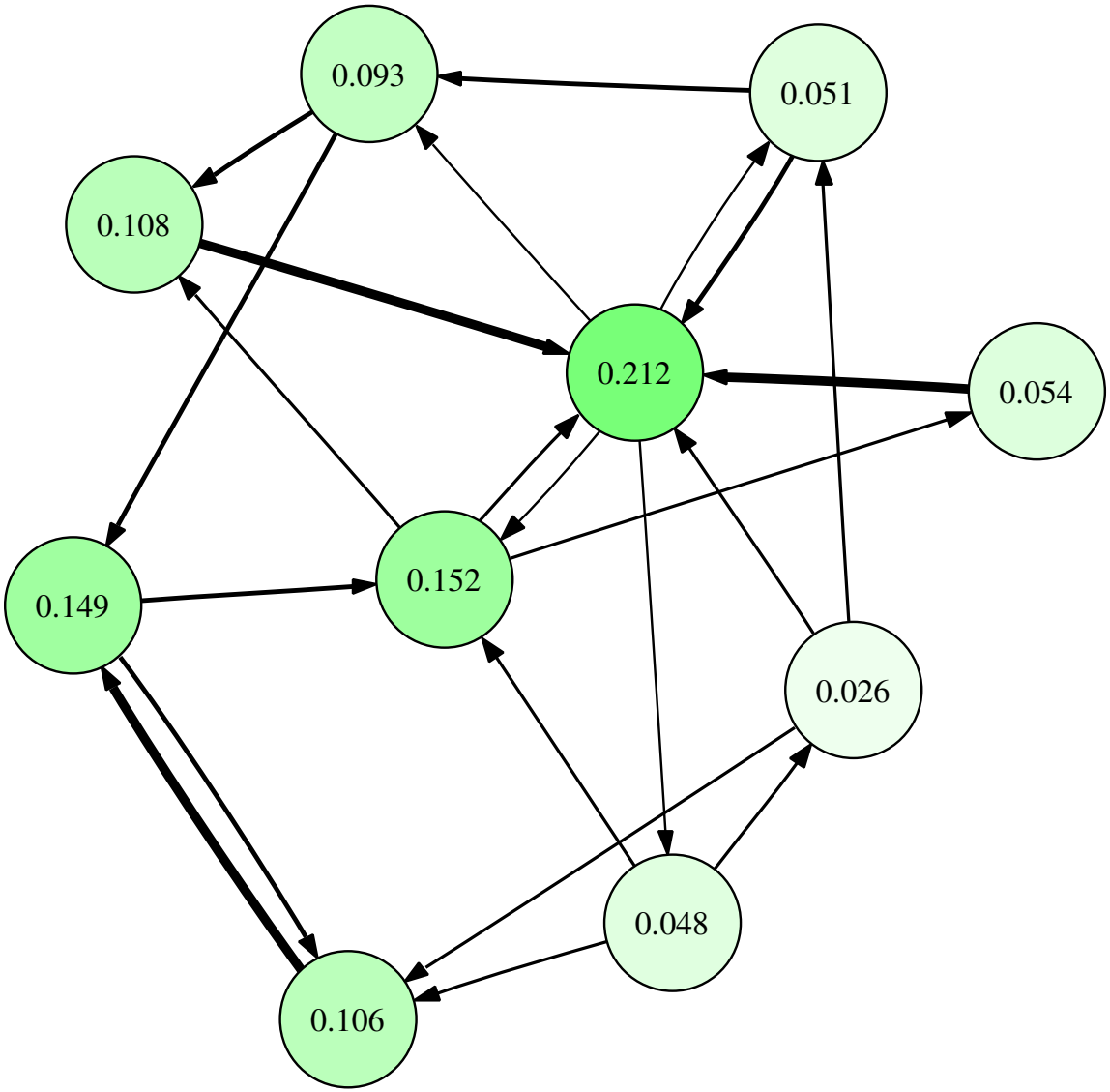
Queques itérations PageRank



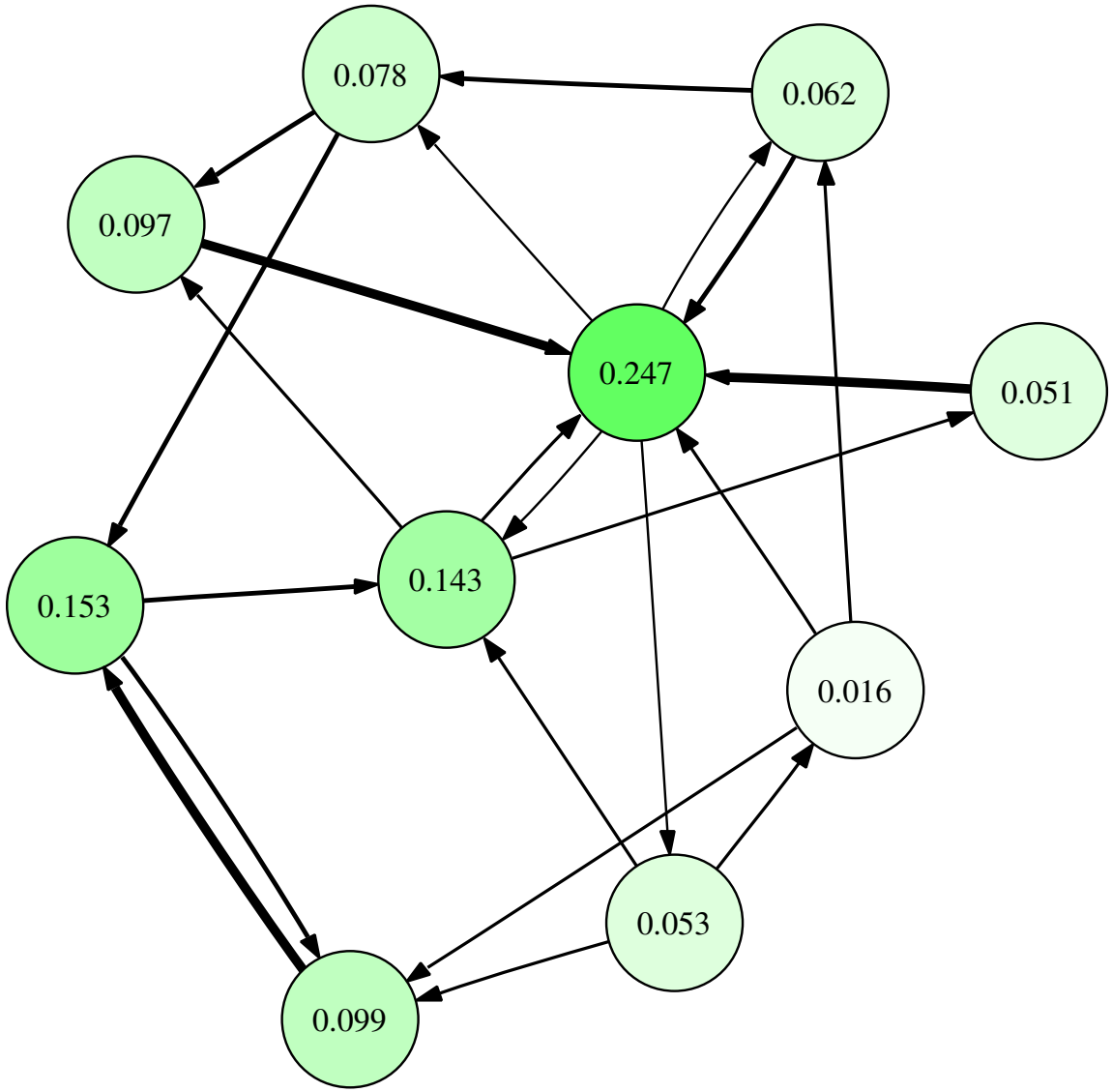
Queques itérations PageRank



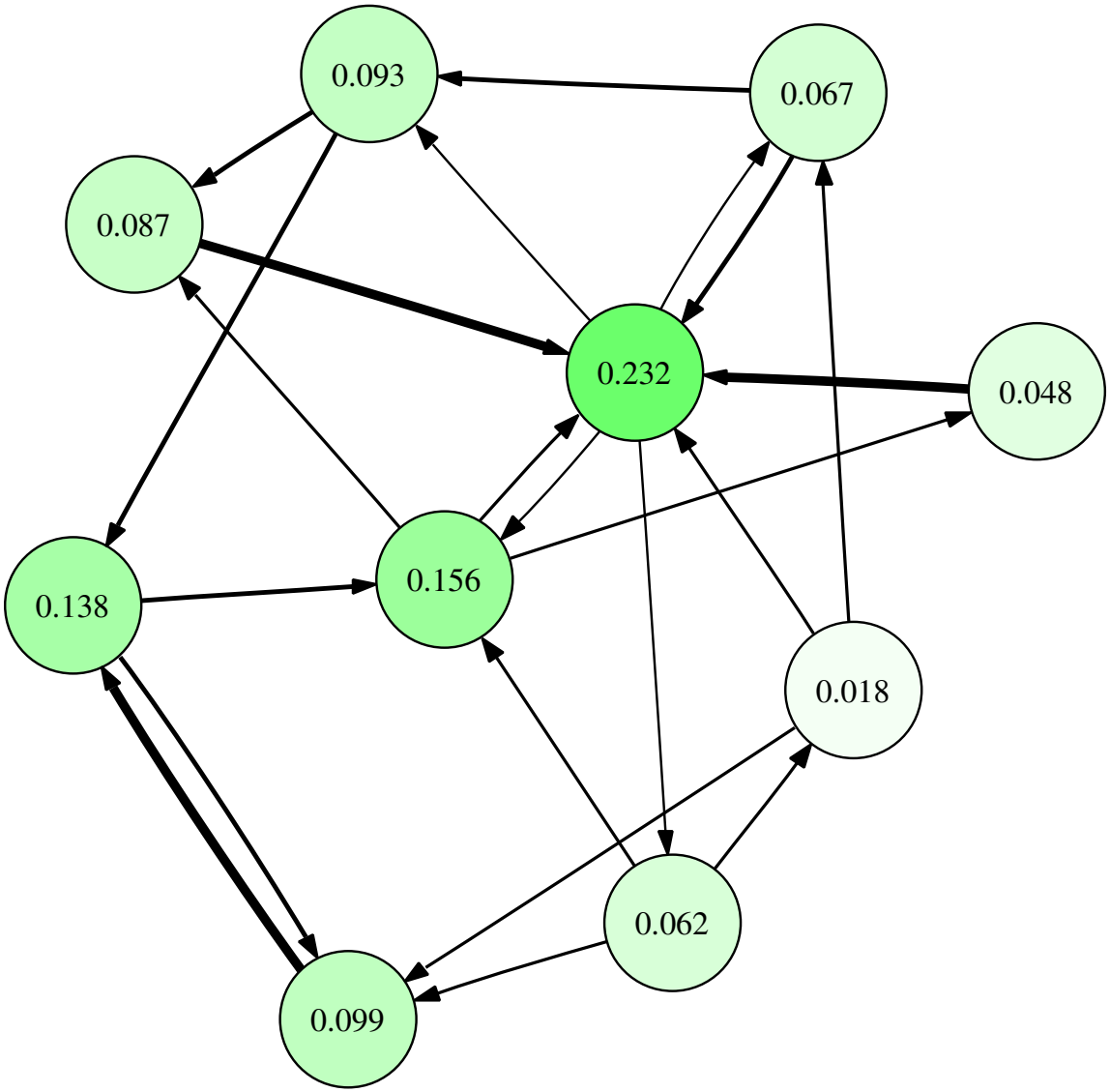
Queques itérations PageRank



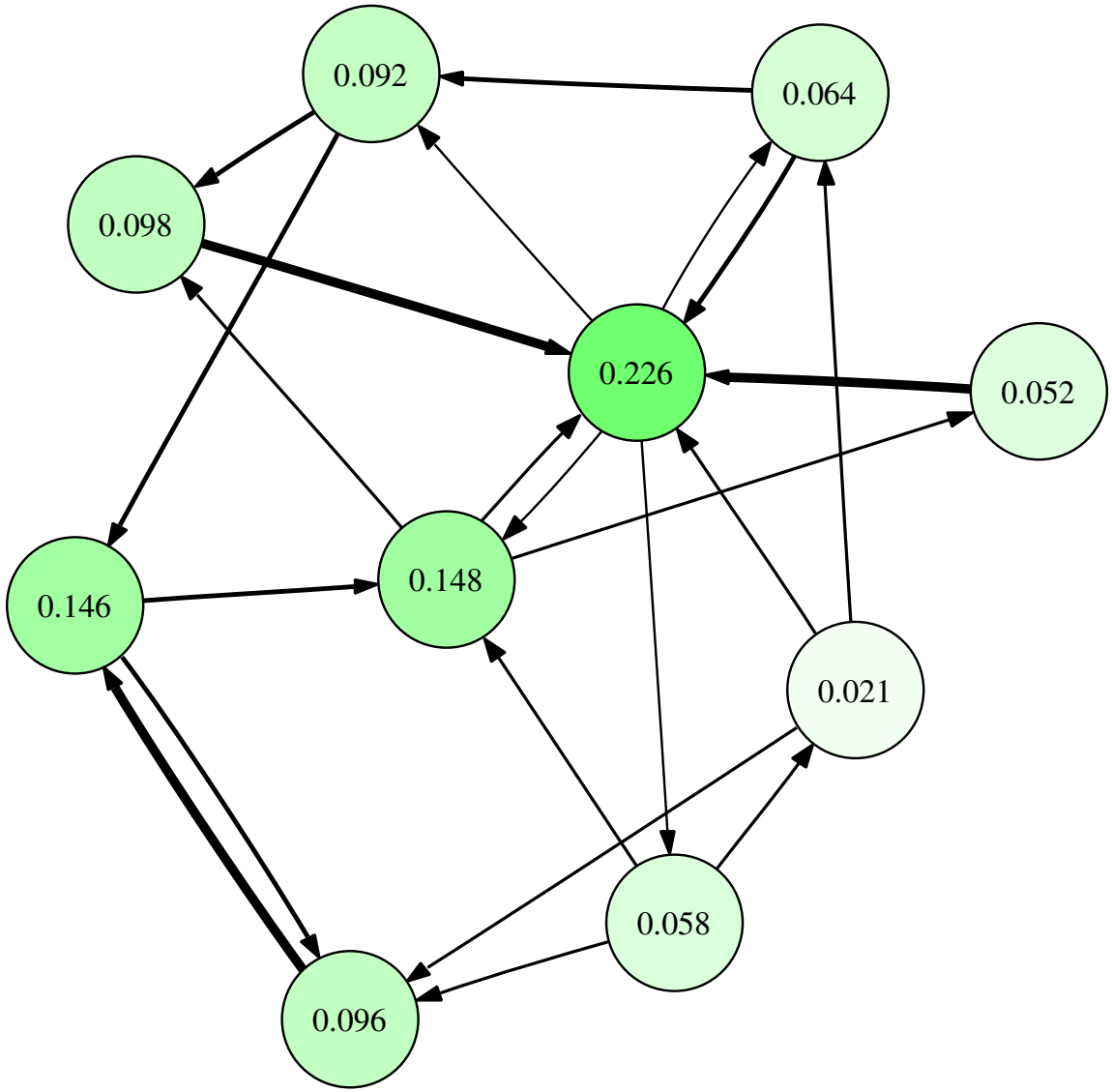
Queques itérations PageRank



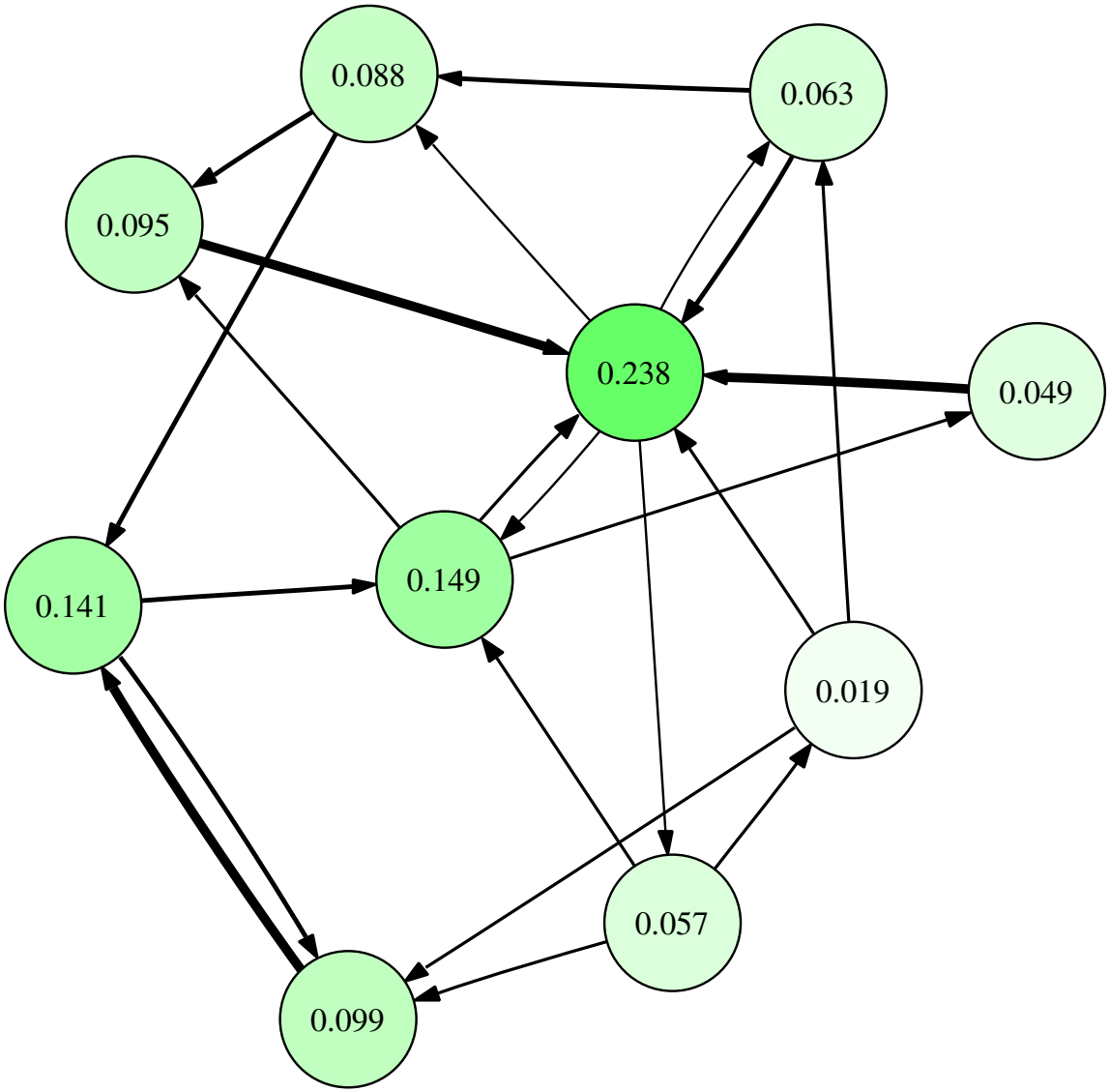
Queques itérations PageRank



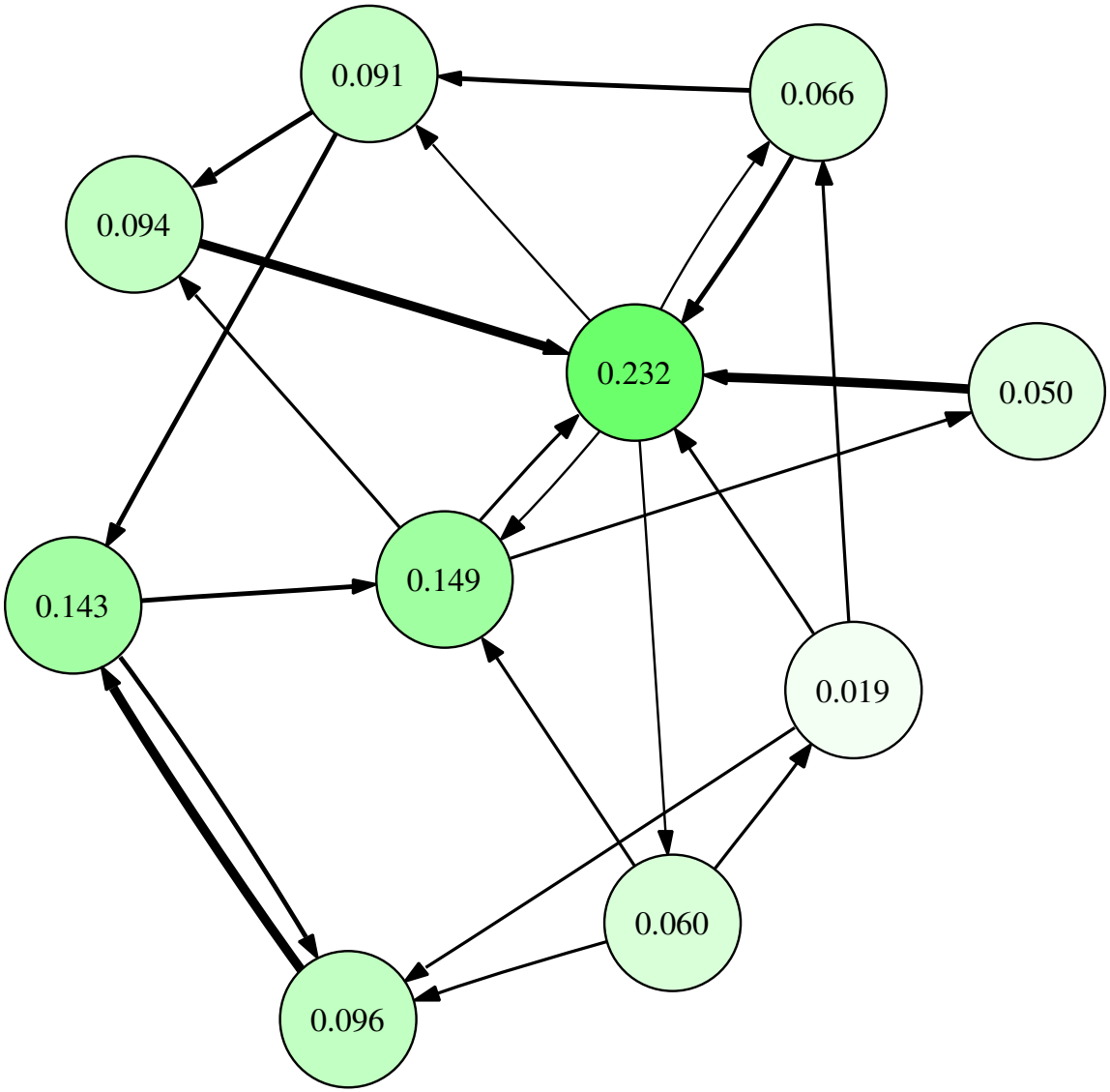
Queques itérations PageRank



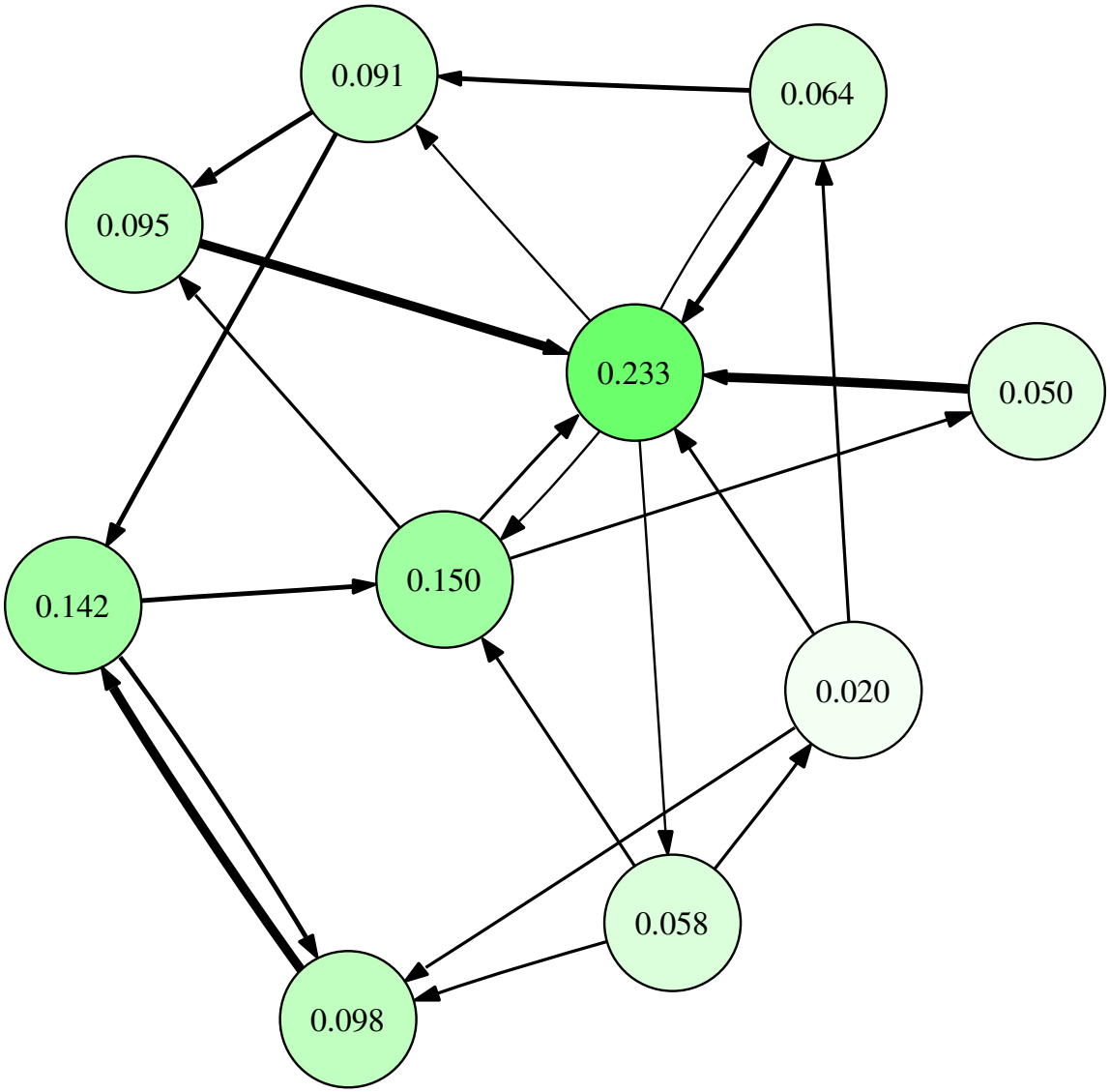
Queques itérations PageRank



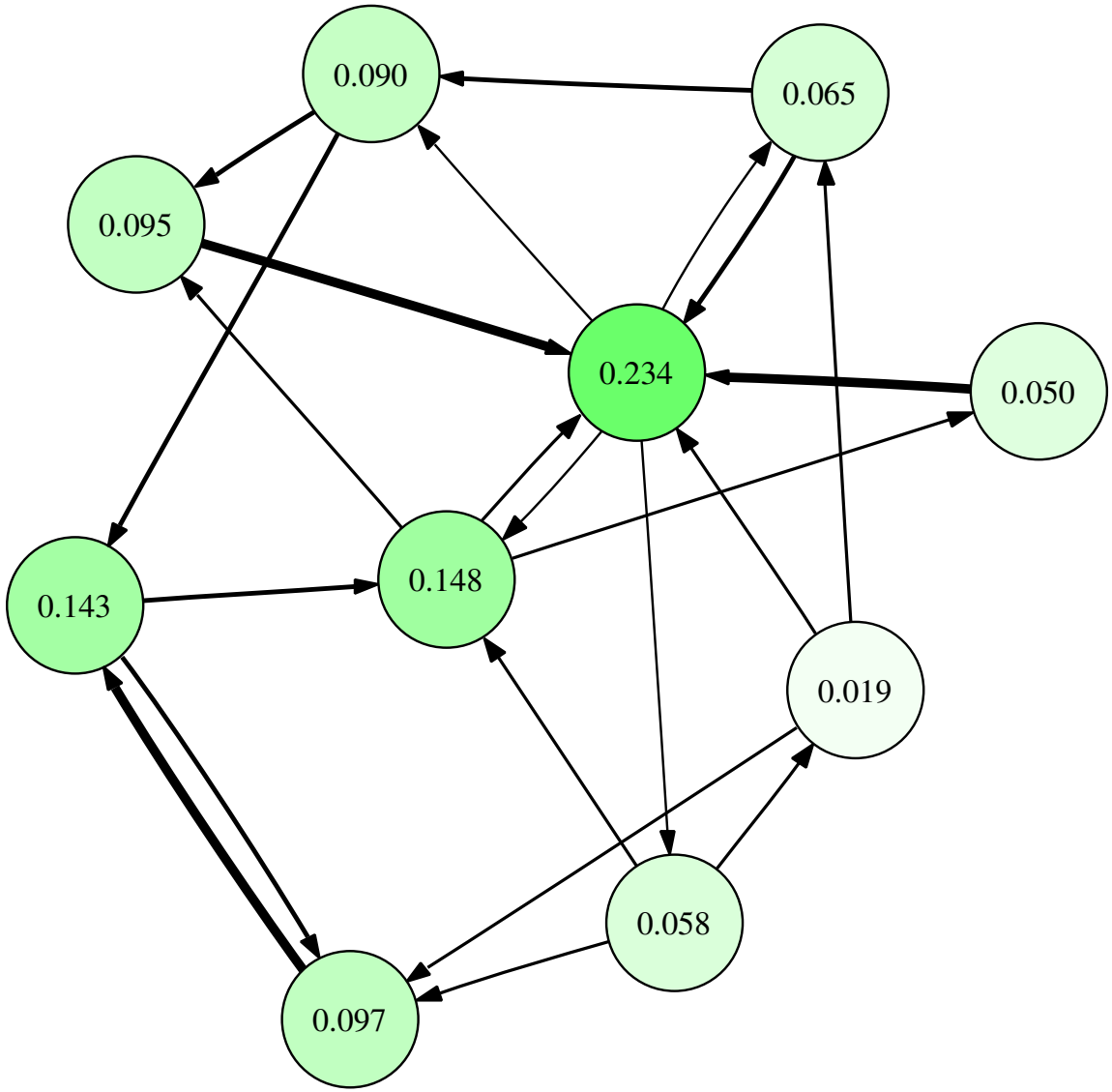
Queques itérations PageRank



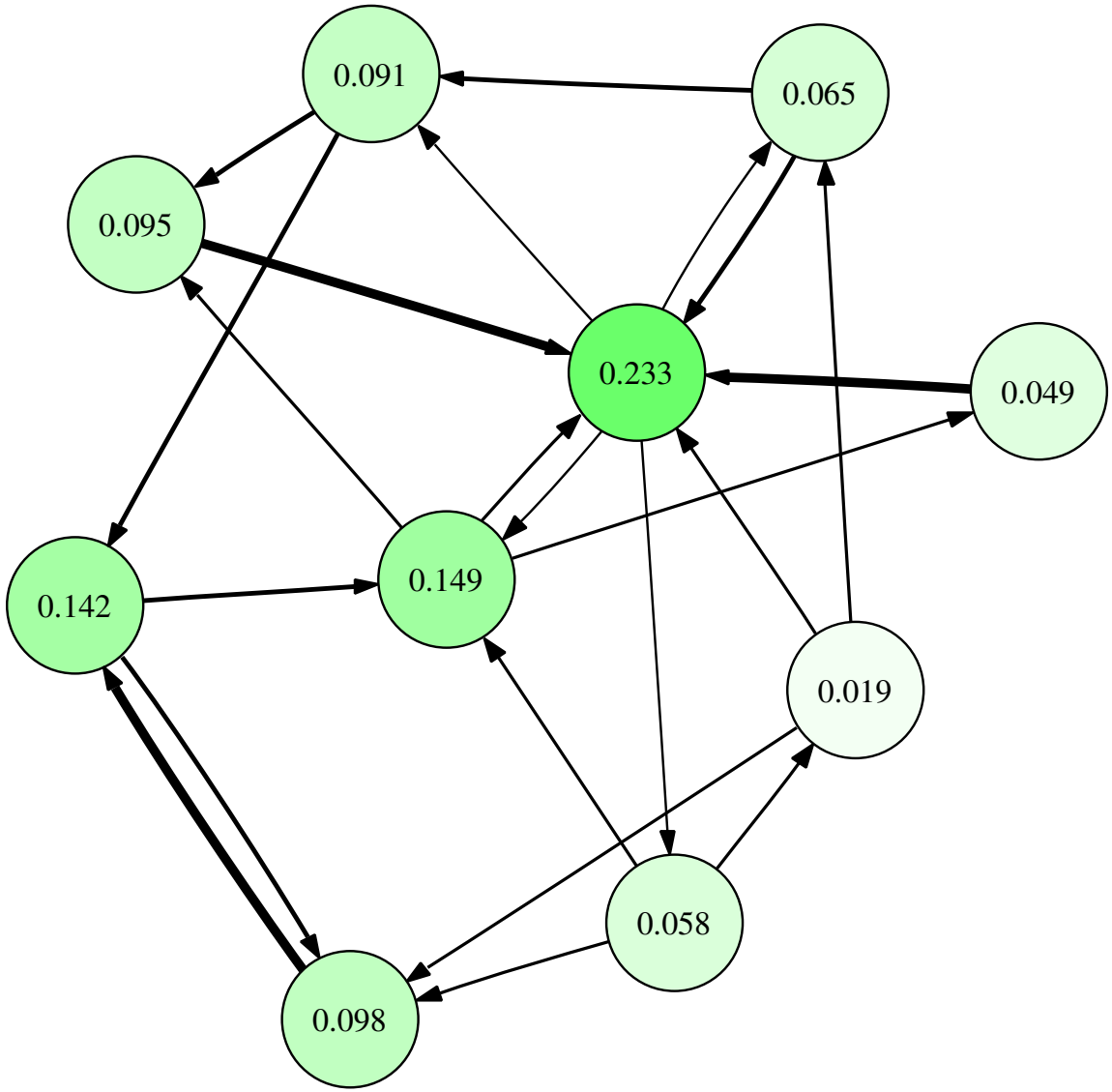
Queques itérations PageRank



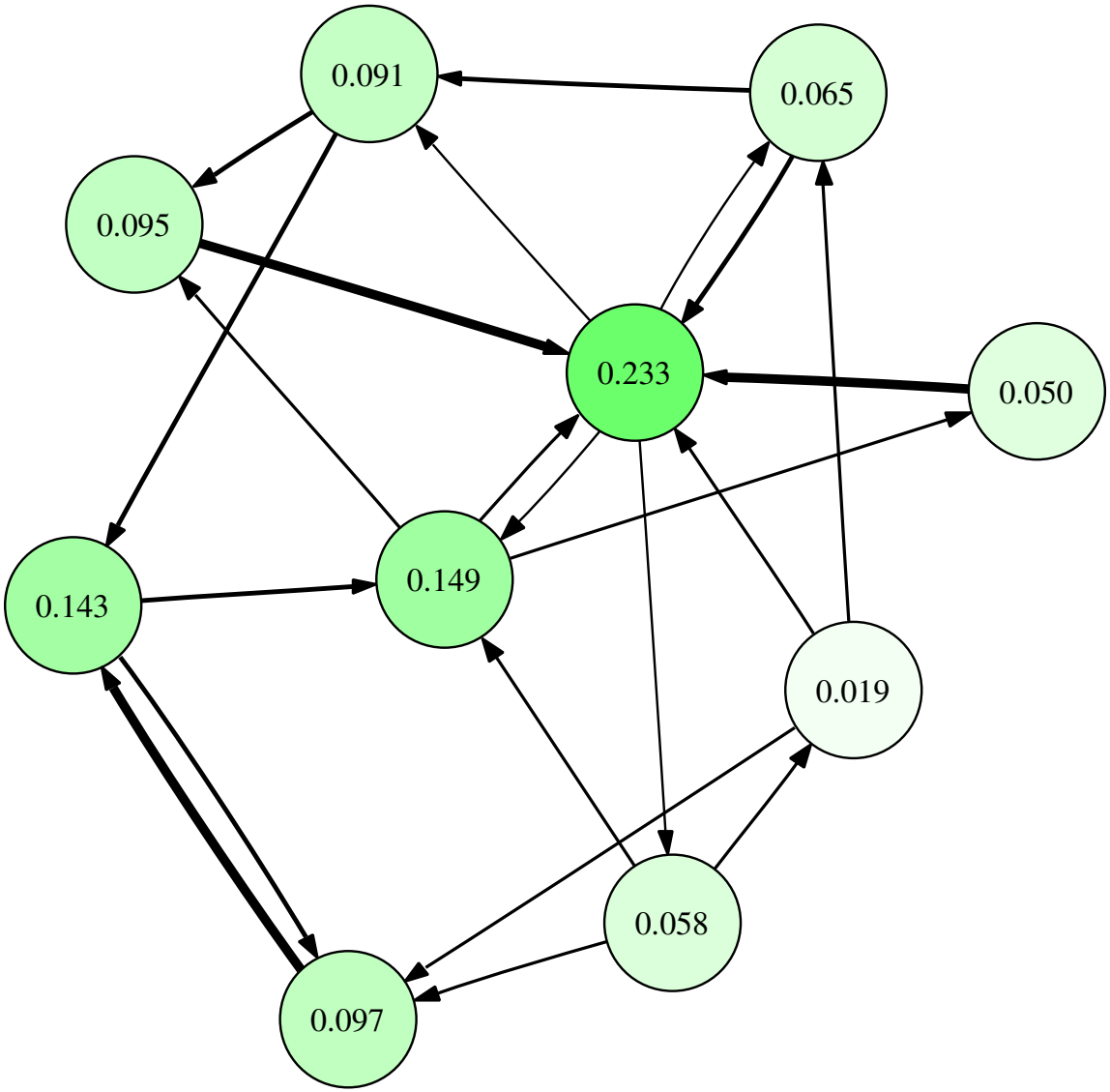
Queques itérations PageRank



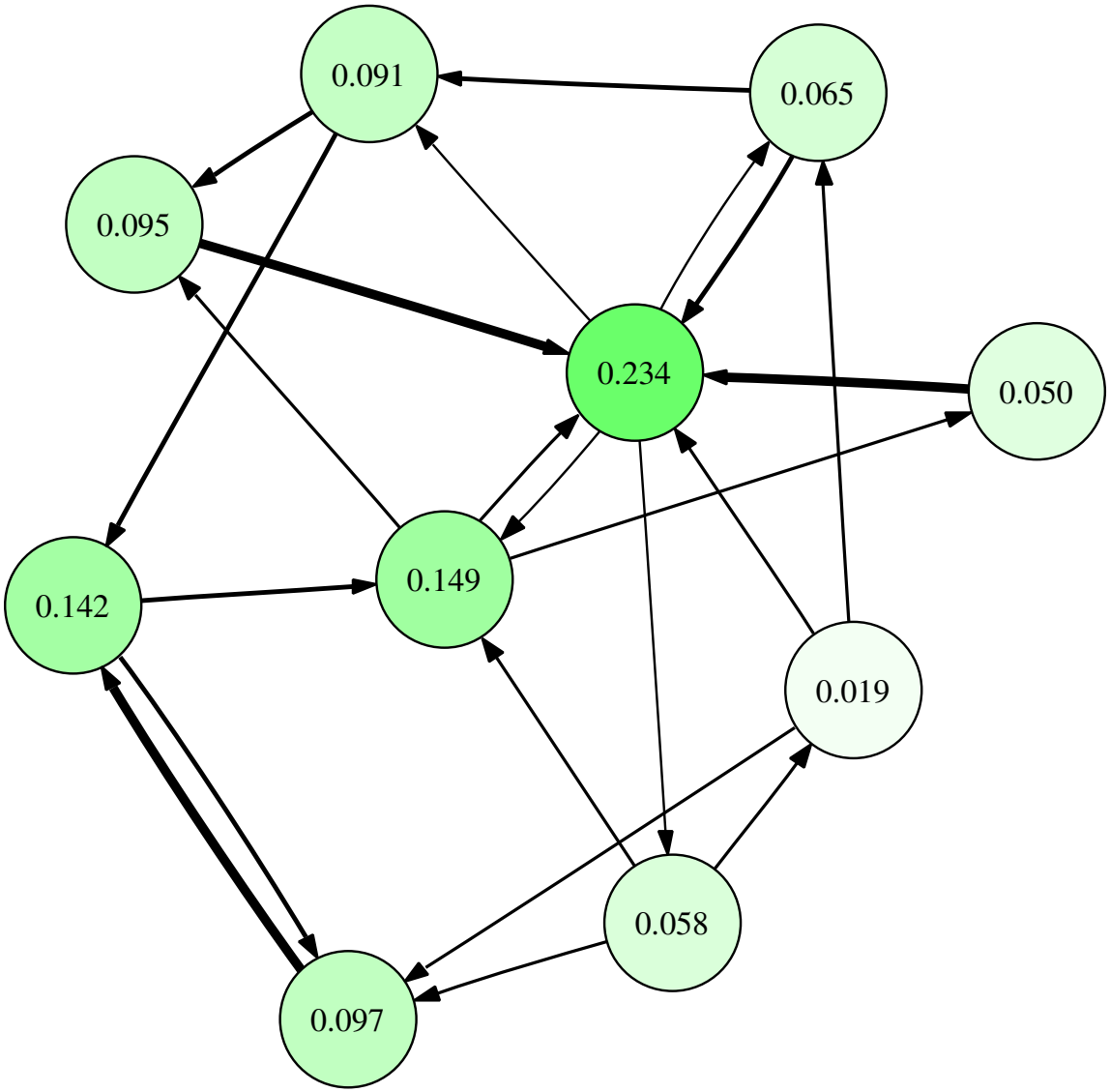
Queques itérations PageRank



Queques itérations PageRank



Queques itérations PageRank



Petite extension pratique

Pour mieux modéliser le comportement d'un utilisateur, on s'autorise des sauts d'une page à une autre, sans qu'il y ait nécessairement de lien.

À chaque étape, on prend en compte la possibilité d'un tel saut avec une probabilité d ($1-d$: damping factor). Ce qui donne :

$$\mathrm{pr}(i) = \left(\lim_{k \to +\infty} ((1-d)G^T + dU)^k v \right)_i$$

où U est une matrice contenant $\frac{1}{N}$ dans chaque cellule.

Le PR réel : un secret bien gardé

Un nombre important de facteurs est pris en compte dans le PageRank.

- Ces facteurs sont très nombreux (plus de 200 d'après Google).
- Leur nature et leur pondération sont secrets pour limiter les chances de manipulations (et la concurrence des autres moteurs de recherche).
- Le terme "PageRank" est une marque déposée et a été l'objet de brevets, à commencer par (U.S. Patent 6,285,999). Le brevet appartient à Stanford University et Google en a l'usage exclusif, mais l'algorithme a beaucoup évolué depuis le dépôt en 98.
- Beaucoup de spéculations sur ce sujet, voyons quelques- uns des paramètres connus. . .

Quelques paramètres

- Sur la page (“ onpage ”)
  - Ancienneté / Fréquence d'actualisation
  - Texte = visible sur la page / Code = Meta tags = non visibles sur la page
- Sur le site (“ onsite ”)
  - Lien internes, arborescence, fil d'ariane (“ Breadcrumbs ”)
  - Paramétrage sur Google outils pour les webmasters (Sitemap)
- Hors du site (“ offsite ”)
  - Liens entrants en (petite) partie visibles via une recherche Google(PageRank, Âge, TrustRank de la page, Social bookmaking, tweets. . .)
- Un débat : Google utilise t-il les données qu'il stocke sur le comportement des internautes pour le calcul du PageRank ?
  - Temps passé sur le site, statistiques renvoyées par la barre d'outil google, annotations sidewiki, citations d'URL dans gmail, requêtes avec l'URL du site, marque-pages Google, âge/sexe/localisation des internautes, leurs recherches précédentes . . . . les licences de ces services précisent souvent que non.

SERP Rank

C'est l'ordre de présentation des liens lorsque l'on entre des mots-clés dans un moteur de recherche

- La page de résultats présente une liste ordonnée de liens vers des pages/images/vidéos, associés à des textes courts (snippets)
- Le SERP Rank est fonction du PageRank, mais aussi de facteurs liés aux mots-clés.
- SERP = Search Engine Results Page

La semaine prochaine

Début des TP :

- MongoDB