Algorithmes de Ranking et de Recommandations

J.M. Fourneau

Université de Versailles, DAVID, France

Motivation

- Ranking : donner un ordre aux pages du web pour répondre à une requête
- Comment traiter en temps réel sachant la taille du Web ?
- Comment donner une pertinence à une page ?
- Recommandations : connaissant vos achats dans le passé ainsi que tous les achats de tous les clients, comment vous proposer de nouveaux achats pour augmenter le chiffre d'affaires ?
- Recommandations plus sophistiquées : connaissant les notes de vos achats par le passé ainsi que toutes les notes données par les clients, comment vous proposer de nouveaux achats pour augmenter le chiffre d'affaires ?
- Même questions avec les contenus que vous regardez et que l'on vous propose.

Plan

- Ranking, l'approche de PageRank
- Gestion de grands objets en mémoire (matrice creuse)
- Convergence de PageRank
- Valeurs Propres, Vecteurs Propres
- Les alternatives à PageRank (SALSA, HITS)
- Agir sur PageRank
- Recommandations : similarité de clients et similarité d'objets
- Calcul d'une note virtuelle
- Singular Value Decomposition

Ranking

- Acteurs : les utilisateurs du web qui ont collectivement construit les pages du Web et un logiciel répondant, en temps réel, à une requête de mots clés par une liste ordonnée de pages pertinentes et liées à la requête.
- Noter les pages, puis trier sur les notes.
- Taille du Web indexé : environ 10^{10} pages
- Impossible d'explorer le Web pour répondre (problématique de la taille)
- Comment définir la pertinence ? (problématique de la pertinence)
- Comment gérer les synonymes (voiture et auto ?) (problématique du langage naturel)
- Comment gérer les homonymes (moule : pour la fonderie ?, la biologie ?)
- Phrases ambigues : "les poules du couvent couvent" (problématique de la grammaire d'un langage naturel).

On se concentre sur

- la gestion des objets en mémoire
- la définition de la pertinence
- le calcul spéculatif et la réponse temps réel
- les algorithme de calcul de la pertinence
- la boucle de retroaction (le google bombing)
- la personnalisation des réponses

On ne fera pas

- la gestion des robots qui indexent le web
- les techniques pour que votre page soit bien traitée
- les questions liées au langage naturel
- la détection de communautés
- l'algorithmique du texte

Hypothèses de départ

- Le WEB est visité par des robots qui indexent
  - les textes avec leurs propriétés typographiques
  - les URL (liens)
  - les images
  - les vidéos
- On construit deux objets
  - le graphe du WEB
  - le dictionnaire des mots avec une pondération qui indique leur importance pour la page

Graphe du WEB

- Multigraphe orienté
- Les noeuds sont les pages
- Les arcs sont les URL entre les pages
- Comme il peut y avoir plusieurs URL entre deux pages, il peut y avoir plusieurs arcs de la page i vers la page j: c'est un multigraphe et non pas un graphe.

Le dictionnaire

- Pour chaque page, on collecte les mots
- On ajoute les synonymes (voiture = automobile)
- On ajoute une note qui prend en compte l'importance dans la page :
  - nombre d'occurrences dans la page
  - attributs typographiques (couleur, fonte, gras, souligné, italique)
  - place dans le texte (titre, en-tête de paragraphe)
- Le calcul de cette note est peu documenté, il y a des conseils de Google pour qu'une page soit bien notée
- Attention aux ancres : on ajoute aux mots de la page, le texte des ancres associées aux URL qui ont menés à cette page.

Le dictionnaire-2

- Pour chaque mot du dictionnaire, on a donc la liste des pages WEB avec la note de ce mot pour cette page.
- On organise cette structure de dictionnaire pour avoir un accès très rapide aux mots les plus courants.
  - Mot 1 : page 1, note 1, page 2, note 2, page 3, note 3
  - Mot 2 : page 2, note 2, page 4, note 4
  - Algorithmique du texte (voir sur le web)

Aspect Temporel

- L'indexation du web est faite en continu.
- Le graphe est construit périodiquement (tous les mois ?) pour en tirer les notes de pertinence.
- Le dictionnaire peut être mis à jour plus souvent.
- Attention la construction du dictionnaire permet de censurer très vite et de faire disparaitre une partie du Web.

Construire la matrice du graphe

- $d^+(i)$: degré sortant de i (en prenant en compte les arcs multiples).
- Première étape : on construit une matrice d'adjacence du multigraphe
- $A[i,j] =$ nombre d'arcs depuis i vers j
- Matrice positive : $A[i,j] \geq 0$ pour tout i et j
- Attention : Positive Matrix (au sens anglais) signifie $A[i,j] > 0$. Attention aux erreurs possibles de traduction. La différence est importante pour les algorithmes.

Ranking par Pagerank

- Idée naive : Pour une requête, on calcule des notes pour toutes les pages du web, on trie et on rend les pages avec les meilleures notes en temps réel pour l'utilisateur.
- Comment faire sachant que l'on a indexé 10^{10} pages ?
- Séparer le calcul de la note en 2 notes :
- 1 note de contenu qui repose sur la requête et qui se calcule avec le dictionnaire
- 1 note de page qui repose sur la pertinence de la page dans le web et qui se calcule avec le graphe du Web
- Combinaison des deux notes : a priori multiplicative pour qu'une note nulle ou trop faible élimine la page

Note de contenu de la requête

- doit être fait à chaque fois et en temps réel.
- Pour chaque page du web
- On intersecte la liste de termes de la requête avec le dictionnaire
- On combine les notes des mots pour chaque page (a priori c'est une somme) pour obtenir la note de la page pour la requête
- On peut suggérer des corrections : un mot n'est pas dans la requête mais un mot approchant s'y trouve.
- Complexité mieux que linéaire dans la taille du dictionnaire

Note sur la pertinence

- C'est une note de la page liée à l'expertise de l'auteur
- l'expertise est reconnu par les pairs (même idée qu'en bibliographie)
- Les mots de la requête ne sont pas pris en compte
- Cette note n'est pas recalculée en temps réel.
- Elle tient compte des avis de tous les auteurs à propos de toutes les pages Web (au moins quadratique sur la taille du web).

Formellement pour PageRank

- 2 axiomes
- Une page pertinente pointe vers des pages pertinentes
- Une page pertinente est pointée par des pages pertinentes.
- 3 autres idées sur les combinaisons de pertinence
- Chaque page distribue de la pertinence et en reçoit (Equilibre)
- La pertinence d'une page est divisée équitablement sur les pages vers lesquelles elle pointe
- La pertinence d'une page est la somme des parts de pertinence qui pointe vers elle.

Critique

- Toute citation est positive
- Dire qu'une page est nulle ou qu'elle contient des erreurs renforce sa pertinence (pour PageRank).
- La pertinence est additive (beaucoup de soutiens de pages peu pertinentes peut être plus valorisant qu'un avis positif de l'expert du domaine: effet réseaux sociaux). C'est le principe qui permet le "Google bombing".

Implémentation des échanges de pertinence

- La pertinence est un réel positif $\pi(i)$ pour la page i.
- On supposera qu'il est compris entre 0 et 1 pour utiliser des notions de probabilité (le surfer aléatoire)
- On construit une matrice P de distribution de la pertinence
- la seconde règle sur les combinaisons se traduit par

$$\pi(j) = \sum_i \pi(i) P[i,j]$$

- la première règle sur les combinaisons dit que $P[i,j]$ vaut $k/d^+(i)$ si il y a k arcs de i vers j.

$$P[i,j] = A[i,j]/d^+(i)$$

  si $d^+(i) > 0$
- et $P[i,j] = 0$ pour tout j si $d^+(i) = 0$

Définition implicite

- Il faut que $d^+(i) > 0$. Toute page doit avoir au moins un successeur dans le graphe du WEB
- C'est loin d'être vrai en réalité.
- On verra plus tard comment régler les problèmes.
- Supposons dans un premier temps que $d^+(i) > 0$ soit vrai.

$$\pi(j) = \sum_i \pi(i) A[i,j] / d^+(i)$$

- On pose π : vecteur LIGNE contenant les $\pi(i)$.
- Comment résoudre une équation où le vecteur π est défini implicitement en fonction du vecteur π ?

Plus formellement

- $$\pi(j) = \sum_i \pi(i) P[i,j]$$
- Vous devriez reconnaitre (?) le produit à gauche du vecteur π par la matrice P
- Vous avez plus certainement fait des produits à droite en math avec des vecteurs COLONNES
- On va écrire plus simplement

$$\pi = \pi P$$

2 familles simples de solution pour $\pi = \pi P$

1. Résolution de l'équation $xA = b$ (ou $x = bA^{-1}$) si il existe une inverse de A
2. Recherche du spectre de A : couple x et λ tel que $xA = \lambda x$

- Une base commune : l'algèbre linéaire
- Un choix basé sur la complexité en temps et en espace

Algèbre Linéaire

- Pour les fondamentaux, voir un livre de mathématiques.
- Je vais uniquement survoler quelques opérations plus ou moins élémentaires et surtout travailler sur les matrices associées au graphe du Web
- Les éléments sont des réels positifs (≥ 0), ou sont parfois associés à des probabilités.
- Malgré cela, on verra apparaitre des complexes.
- On verra en TD les aspects d'implémentation pour de grandes matrices.
- Telecharger Scilab pour faire des tests sur de petites matrices.

Rappel Algèbre Linéaire -, Opérations

- Somme matrice $D = A + B$ avec A, B et C matrices $N \times M$.
- $D[i,j] = A[i,j] + B[i,j]$
- Produit matrice $C = AB$ avec A matrice $N \times K$, B matrice $K \times M$ et C matrice $N \times M$.
- $C[i,j] = \sum_{k=1}^{K} A[i,k]B[k,j]$
- Attention à la compatibilité des tailles
- Les matrices peuvent ne pas être carrées ($N \neq M$)

Property

Soient A, B et C trois matrices (taille compatible) , $C(A+B) = CA + CB$ (distributivité), et $A(BC) = (AB)C$ associativité.

Rappel Algèbre Linéaire - 3 - Notations

- Notation : on appelle e un vecteur rempli de 1. Id est la matrice Identité (composée de 1 sur la diagonale).

Property

Soit A matrice carrée , $A\ Id = Id\ A = A$.

- Attention, le produit de matrice n'est pas commutatif, $AB \neq BA$ en général.
- Un vecteur ligne à N valeurs est une matrice $1 \times N$
- Un vecteur colonne à N valeurs est une matrice $N \times 1$
- Donc on définit les produits à gauche vecteur (ligne)-matrice et les produits à droite matrice-vecteur (colonne).
- Vous avez utilisé les produits à droite plus classiquement. C'est le sens utilisé en physique en général.

Rappel Algèbre Linéaire - 4 Opérations

Definition

Soit A une matrice $N \times M$, A^t est la matrice $M \times N$ telle que $A^t[i,j] = A[j,i]$. C'est la matrice transposée.

Property

Soit A matrice , $(A^t)^t = A$.

- La transposée d'un vecteur ligne est un vecteur colonne et la transposée d'un vecteur colonne est un vecteur ligne.

Property

Soit A et B 2 matrices de taille cohérente , $(AB)^t = B^t A^t$.

Ligne et Colonne

- Soit C un vecteur colonne $N \times 1$ et L un vecteur ligne $1 \times N$
- LC est un scalaire (c'est le produit scalaire que vous connaissez)
- CL est une matrice $N \times N$ et

$$(CL)[i,j] = C[i]L[j]$$

- Attention, en général une matrice n'est pas inversible, A^{-1} peut ne pas exister.

Definition

on dit que A est une matrice non singulière si A^{-1} existe.

Property

Si A est non singulière, $AA^{-1} = A^{-1}A = Id$.

Property

Si A et B sont non singulières, AB est non singulière, et $(AB)^{-1} = B^{-1}A^{-1}$.

Spectre, Valeurs Propres-Vecteurs Propres

Definition

Soit P une matrice carrée, on dit que λ scalaire et x vecteur ligne sont respectivement valeur propre et vecteur propre à gauche si

$$xP = \lambda x$$

Definition

Soit P une matrice carrée, on dit que λ scalaire et c vecteur colonne sont respectivement valeur propre et vecteur propre à droite si $Pc = \lambda c$.

- les valeurs propres sont les mêmes pour les vecteurs propres à gauche et ceux à droite quand la matrice est réelle.
- L'ensemble des valeurs de λ s'appelle le spectre.
- Par contre les vecteurs propres à gauche sont en général différents des vecteurs propres à droite.

Exemple

- $P = \begin{bmatrix} 0.6 & 0.4 \\ 0.8 & 0.2 \end{bmatrix}$
- Vérifiez que 1 est valeur propre à gauche et à droite
- Calculez les vecteurs propres à gauche et à droite associée à 1.

Property

Soit A une matrice, les valeurs propres de A sont les zeros du polynome caractéristique :

$$P(\lambda) = det(A - \lambda Id)$$

Le degré de $P(\lambda)$ est N la taille de la matrice et du graphe du Web.

- Il est difficile en général de calculer exactement les valeurs propres
- Il existe des algorithmes itératifs pour approcher les solutions
- On peut avoir très rapidement des bornes par les cercles de Geshgorin.
- Les valeurs propres peuvent être complexes mêmes si les matrices sont réelles.

Exemple

- pour la matrice $P = \begin{bmatrix} 0.5 & 0.1 & 0.4 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0.5 \\ 0.1 & 0.6 & 0.1 & 0.2 \end{bmatrix}$. le spectre vaut $(1, -0.3, 0.5i, -0.5i)$
- Le 1er vecteur propre à gauche vaut $(-0.8227736, -0.3894462, -0.3510501, -0.2194063)$.
- Le second vecteur propre à gauche vaut $(0.3162278, -0.3162278, -0.6324555, 0.6324555)$
- Le troisième vecteur propre à gauche vaut $(0.6611074, -0.2908872 - 0.3173315i, 0.0264443 + 0.449553i, -0.3966644 - 0.1322215i)$
- Le premier vecteur propre à droite vaut $(1, 1, 1, 1)^t$.
- Calculé par la commande spec(P) de Scilab.

Cercles de Gershgorin

- On considère le plan complexe.

Definition

A une matrice A de taille $N \times N$, on associe

- pour chaque ligne i, un cercle centré sur $(A[i,i],0)$ et de rayon $\sum_{j \neq i} |A[i,j]|$
- pour chaque colonne i, un cercle centré sur $(A[i,i],0)$ et de rayon $\sum_{j \neq i} |A[j,i]|$

Theorem

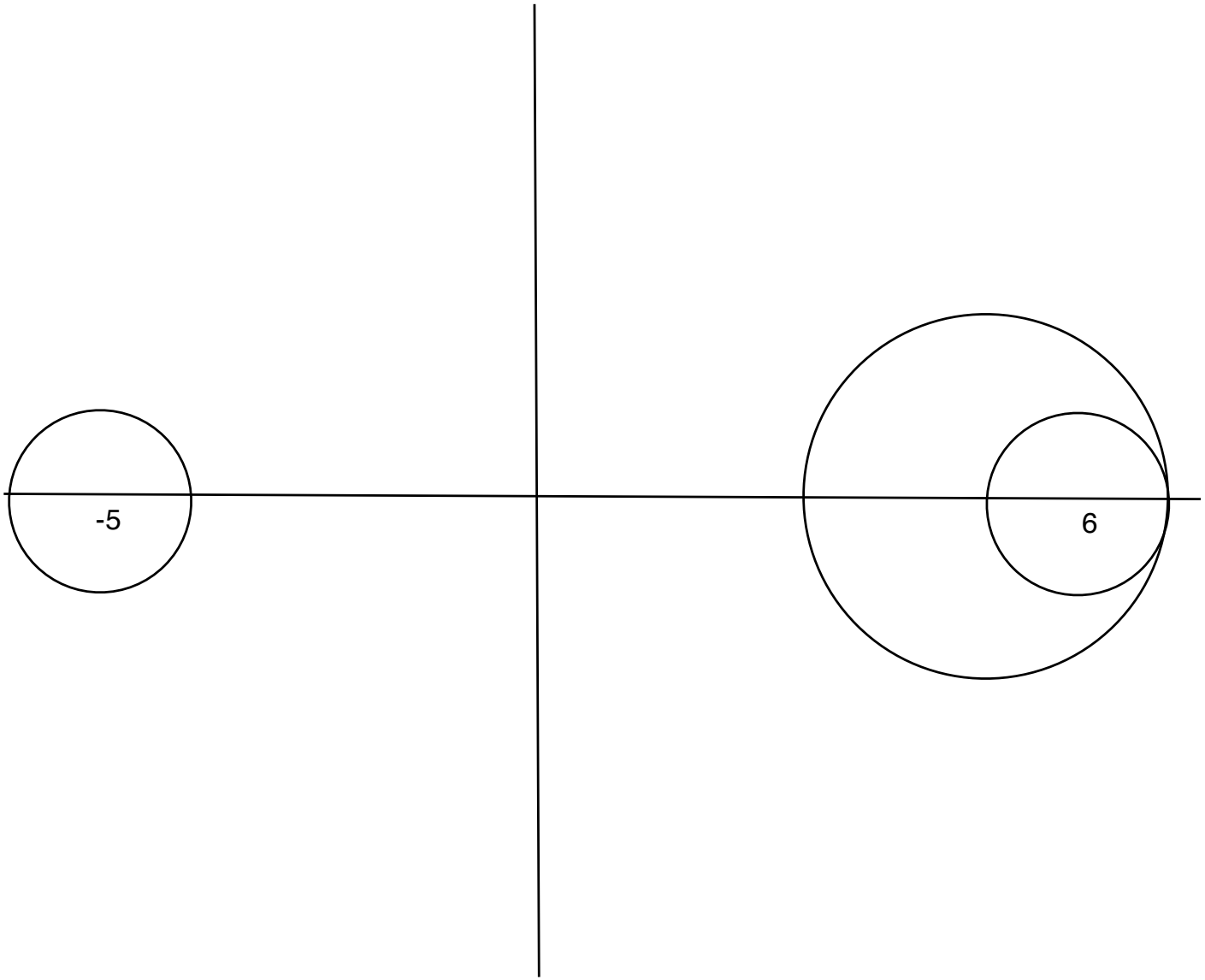
La valeur propre i de A est inclus dans l'intersection des cercles de Gershgorin associés à la ligne i et la colonne i.

- Le cercle associé à la ligne i et celui associé à la colonne i ont le même centre. Donc il faut regarder le plus petit rayon.

Exemple (non stochastique

- $P = \begin{bmatrix} 5 & 1 & 1 \\ 0 & 6 & 1 \\ 1 & 0 & -5 \end{bmatrix}$
- Ici pour définir un cercle, on donne son centre et son rayon.
- Les cercles associés aux lignes sont (5,0,2), (6,0,1) et (-5, 0 , 1)
- Les cercles associés aux colonnes sont (5,0,1), (6,0,1) et (-5, 0 , 2)
- Les valeurs propres sont dans les cercles (5,0,1), (6,0,1) et (-5, 0 , 1)

Figure Cercle de Gershgorin



Base

- Sur un espace vectoriel de dimension N, une base $v1,...vN$ est un ensemble de n vecteurs indépendants.
- Si les v_i sont des vecteurs ligne, alors $\begin{bmatrix} v1 \\ v2 \\ \\ \\ vN \end{bmatrix}$ est une matrice non singulière D.
- Sur une base $v1,...vN$ tout vecteur x peut se décomposer $x = \sum_i \alpha_i vi$

Property

En changeant de base, on change les vecteurs propres mais on ne change pas les valeurs propres. L'opération de changement de base pour une matrice P se traduit par $D^{-1}PD$.

Definition

on appelle matrice stochastique P une matrice carrée telle que $P[i,j] \geq 0.0$ pour tout i et j et $\sum_j P[i,j] = 1$ pour tout i.

- On peut aussi noter $Pe^t = 1$.

Property

Soit P et R deux matrices stochastiques de meme taille, et $0 < \alpha < 1$, alors $\alpha P + (1-\alpha)R$ est stochastique et PR est stochastique.

Matrice Stochastique Irréductible

- On construit le graphe $G = (V, E)$ orienté associé à la matrice P comme suit:
  - les sommets de G sont associés au lignes de la matrice P
  - $(i, j) \in E$ si et seulement si $P[i, j] > 0$.

Definition

Soit P une matrice stochastique, P est irréductible si son graphe G associé est fortement connexe.

Matrice Positive et Matrice Primitive

- Il existe une théorie pour une "Positive Matrix" (au sens anglais)
- Attention aux contresens avec l'anglais. Positive (en anglais) signifie > 0. Positif (en francais) signifie ≥ 0.
- Par exemple, le graphe associé à une matrice positive (au sens anglais) est le graphe complet alors que le graphe associé à une matrice positive (au sens francais) est un graphe quelconque, qui peut ne pas être connexe.

Definition

Soit P une matrice stochastique, P est primitive si il existe n tel que P^n soit positive (au sens anglais). C'est à dire, $\exists n, P[i,j]^n > 0, \forall i,j$.

Matrice Primitive et Irréductible

Property

Il existe des matrices irréductibles qui ne sont pas primitives.

- Preuve : $P = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
- $P^2 = Id$, $P^3 = P$
- Par induction $P^{2n} = Id$ et $P^{2n+1} = P$
- Donc aucune puissance n de P vérifie $P^n[i,j] > 0$ pour tout i et j

Matrice apériodique -1-

- Soit P une matrice stochastique irréductible.
- On utilise la matrice P pour décrire une marche aléatoire.
- Si à la date t on est à l'état i, on est en j à la date $t+1$ avec la probabilité $P[i,j]$.
- On définit f_i^n comme la probabilité de premier retour en i en exactement n étapes.
- Si $\sum_{n=1}^{\infty} f_i^n = 1$, on dira que la matrice (et la marche aléatoire) est récurrente, sinon on dira qu'elle est transiente.

Matrice apériodique -2-

- S_i est l'ensemble des valeurs de n telles que $f_i^n > 0$.
- Si $f_i^n > 0$ il y a un cycle de longueur n passant par i (départ de i, retour en i, pas d'autres passages en i).
- On calcule $\gamma_i = PGCD(S_i)$
- Si $\gamma_i = 1$ alors la matrice et la marche sont dits apériodiques,
- Si $\gamma_i = \gamma > 1$, la matrice et la marche sont dits périodiques de période γ.

Theorem

Si P est finie, irréductible et apériodique, alors P est primitive.

Exemple

- $P = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
- $f_1^{(1)} = 0$ et $f_1^{(2)} = 1$ et $f_1^{(3)} = 0$ et $f_1^{(4)} = 0$.
- Premier retour....
- Même séquence pour $f_2^{(i)}$.
- Donc période égale à 2.

Matrice sous stochastique

Definition

P est une matrice sous stochastique, si

1. $P[i,j] \geq 0$
2. $\sum_j P[i,j] \leq 1$ pour toute valeur de i
3. il existe au moins une valeur k, telle que $\sum_j P[k,j] < 1$.

Property

Si P est sous stochastique et qu"il n' y a pas de composantes récurrentes autres que des points isolés, alors $Id - P$ est non singulière et

$$(Id - P)^{-1} = \sum_{i=0}^{\infty} P^i$$

- Lien avec la série géométrique ? $\sum_{i=0}^{\infty} \rho^i = \frac{1}{1-\rho}$.

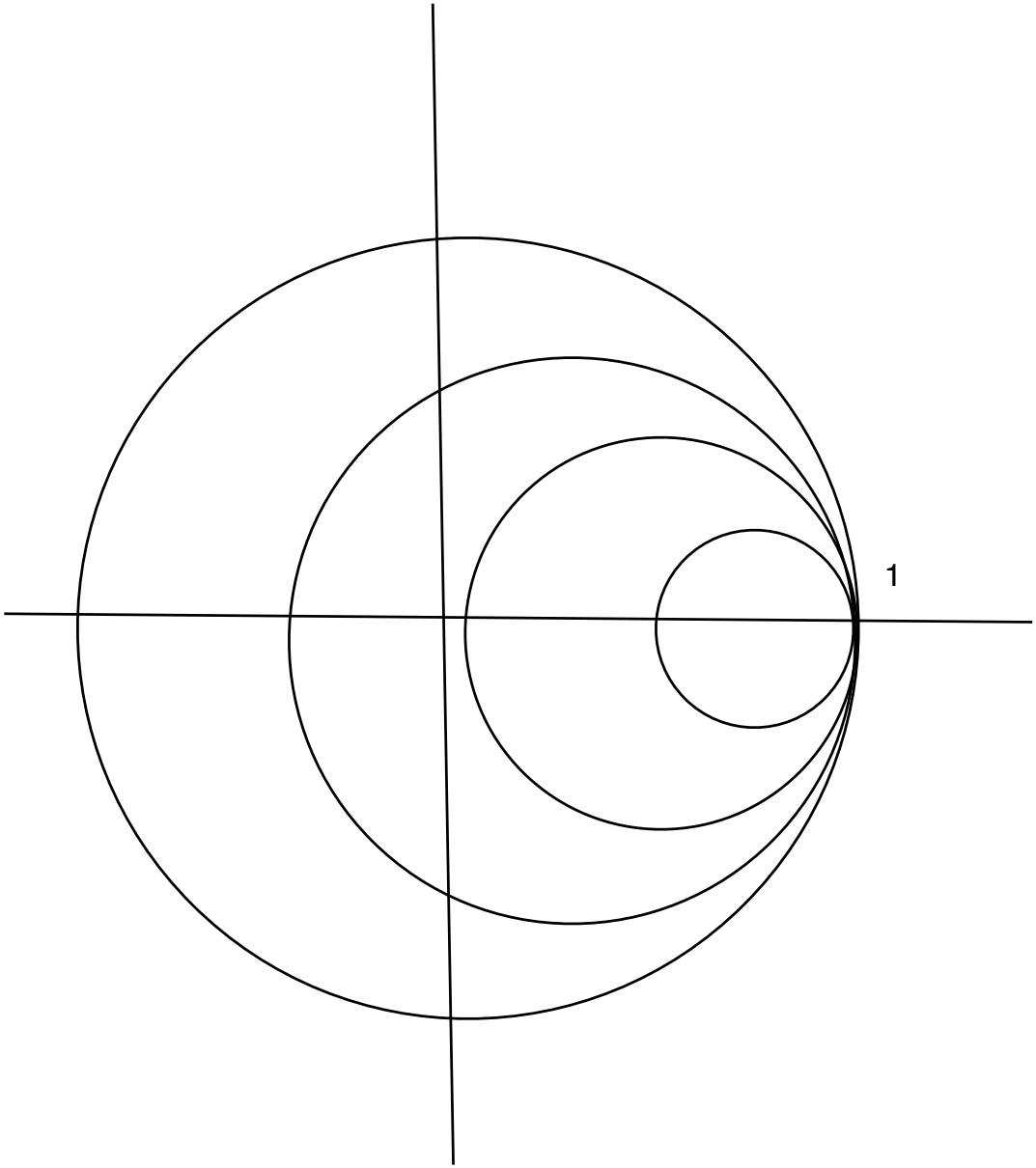
Cercles de Gershgorin pour une matrice stochastique

- Soit P une matrice stochastique quelconque.
- La ligne i de la matrice est associée au cercle de centre $P[i,i]$ et de rayon $1 - P[i,i]$.
- Donc tous les cercles passent par le point $(1,0)$ et sont inclus dans le cercle de rayon 1 et de centre 0
- 1 est une valeur propre car $Pe^t = e^t$ (la somme en ligne est 1)

Property

Toutes les valeurs propres sont de module inférieur ou égal à 1.

Figure Gershgorin et Stochastique



Retour à PageRank

Revenons à la résolution de

$$\pi = \pi P \quad et \quad \pi e^t = 1$$

Approche système linéaire

- Comment passer de

$$\pi P = \pi \quad et \quad \pi e^t = 1.$$

- à $xA = b$
- Remarque $\sum_i \pi(i) = 1$ est équivalent à $\pi.e^t = 1$
- $\pi P = \pi$ et $\pi e^t = 1$ est un système à $n+1$ équations et n inconnues.
- Mais les équations ne sont pas indépendantes.

Equation redondantes

- Toutes les lignes de P sont de somme égale à 1
- Exemple : $P = \begin{bmatrix} 0.5 & 0.1 & 0.4 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0.5 \\ 0.1 & 0.6 & 0.1 & 0.2 \end{bmatrix}$.

$$\pi[1]0.5 + \pi[2] + \pi[4]0.1 = \pi[1]$$

$$\pi[1]0.1 + \pi[3]0.5 + \pi[4]0.6 = \pi[2]$$

$$\pi[1]0.4 + \pi[4]0.1 = \pi[3]$$

$$\pi[3]0.5 + \pi[4]0.2 = \pi[4]$$

- Donc il y a une redondance entre les équations
- Dans l'exemple, en additionnant les 3 premières équations on obtient

$$\pi[1] + \pi[2] + \pi[3](0.5) + \pi[4](0.8) = \pi[1] + \pi[2] + \pi[3]$$

- Donc en simplifiant

$$\pi[4](0.8) = \pi[3](0.5) + \pi[4]$$

- Ce qui est équivalent à la dernière équation.
- On élimine une équation,
- On élimine une colonne (sauf meilleures idées, la dernière)

- transformation de P en $\bar{P}$ (matrice de N lignes et $N-1$ inconnues).
- Ajout d'une colonne supplémentaire égale à e^t pour intégrer l'équation $\pi e^t = 1$

$$\pi \left[\begin{array}{c|c} & \\ \bar{P} & e^t \\ & \end{array}\right] = \begin{bmatrix} 0 \\ \\ \\ 0 \\ 1 \end{bmatrix}$$

- Donc un système linéaire $xA = b$ avec N équations et N inconnues.
- On peut avoir 0 solution, 1 solution ou une infinité de solution.

Elimination de Gauss

- Passer d'un système à N équations et N inconnues à un système à $N-1$ inconnues et $N-1$ équations.
- Idée : Combiner les équations pour rendre le système triangulaire (l'équation i utilise les variables i à N
- Donc l'équation N ne repose que sur la variable N (facile à résoudre).
- Puis on remonte à l'équation $N-1$ qui utilise les variables N et $N-1$ mais la variable N est connue. Donc il est facile de trouver la variable $N-1$. Induction.
- Opérations de combinaison possibles (elles ne changent pas la solution)
  - Echanger les équations E_i et E_j
  - multiplier l'équation i par une constante non nulle a
  - Remplacer l'équation E_i par $E_i + aE_j$

Elimination-Suite

- Chaque étape se fait en choisissant une équation (dite équation de pivot)
- Le terme diagonal de l'équation de pivot s'appelle le pivot
- Le pivot doit être non nul.
- C'est pour cela qu'on considère l'opération d'échange d'équation.
- Remplacer l'équation E_i par $E_i + E_j$ peut conduire à une equation E_i avec beaucoup de termes non nuls. C'est cette opération qui rend le système triangulaire.

Exemple

- Exemple : $\left[\begin{array}{rcr} 2x+y+z & = & 1 \\ 6x+2y+z & = & -1 \\ -2x+2y+z & = & 7 \end{array}\right.$
- On utilise la première équation pour le pivot
- On remplace E_2 par $E_2 - 3E_1$. Le terme 6x disparait.

$$\left[\begin{array}{rcr} 2x+y+z & = & 1 \\ -y-2z & = & -4 \\ -2x+2y+z & = & 7 \end{array}\right.$$

Exemple

- On remplace $E3$ par $E3 + E1$. Le terme $-2x$ disparait.

$$\left[\begin{array}{rcl} 2x + y + z & = & 1 \\ -y - 2z & = & -4 \\ -3y + 2z & = & 8 \end{array}\right.$$

- On choisit l'équation $E2$ pour nouveau pivot.
- On remplace $E3$ par $E3 + 3E2$.

$$\left[\begin{array}{rcl} 2x + y + z & = & 1 \\ -y - 2z & = & -4 \\ -4z & = & -4 \end{array}\right.$$

- Choix du pivot
- Complexité en temps : chaque pivot permet d'éliminer jusquà $N-1$ inconnue
- Chaque pivot coute jusqu'à N^2 opérations
- Chaque pivot peut ajouter jusqu'à $2N$ nouveaux termes dans la matrice.

- Donc algorithme en complexité $N^3/3$
- et remplissage éventuel de la matrice jusqu'à $N^2/2$.
- Rappel $N = 10^{10}$.
- Et le graphe du Web est très creux : le nombre de liens est petit comparé à N^2.
- Donc solution impraticable pour le WEB général
- On peut l'envisager pour des sous problèmes de petite taille.

Autre Solution

- Cherchons numériquement x et λ tels que $xP = \lambda x$.
- Pour notre problème, on sait que $\lambda = 1$ (résultat sur les cercles de Gershgorin)
- Les valeurs propres sont en général des complexes.
- On note le spectre λ_1, λ_2, ..., λ_N en supposant un ordre décroissant en module sur les valeurs propres.
- On suppose $\lambda_1 = 1$ et $|\lambda_k| < 1$ pour tout $k > 1$.

Algorithme des puissances

- Itératif.
- Initialisation : choix d'une heuristique
- Il faudrait démontrer que la solution ne dépend pas de la valeur initiale.
  1. Initialisation $x^{(0)}$
  2. Boucle $x^{(n+1)} = x^{(n)} P$
  3. Jusqu'à réussite d'un test de convergence entre $x^{(n+1)}$ et $x^{(n)}$

Complexité

- en temps : nombre de passages dans la boucle avant convergence $\times$ nombre d'opérations pour le produit xP.
- en espace : la matrice + 2 vecteurs
- Et la matrice reste identique pendant toutes les itérations (elle ne se remplit par comme lors d'une élimination).
- Complexité en espace pour les matrice creuses: O(nombre d'éléments non nuls)
- Et aussi sur le nombre d'opérations pour xP (ca dépend du stockage de la matrice) mais on peut atteindre O(nombre d'éléments non nuls)

Test de Convergence

- Rappel $||x||_1$: norme 1 du vecteur $x = \sum_i |x(i)|$ si x est réel.
- $||x||_2$: norme euclidienne : $\sqrt{\sum_i x^2(i)}$.
- $||x||_\infty$: norme infinie : $Max_i|x(i)|$
- En général, pour vérifier la convergence, on teste $||x^{(n+1)} - x^{(n)}||_1 < \epsilon$.
- Mais ce n'est pas une preuve que la limite exacte π vérifie $||\pi - x^{(n+1)}||_1 < \epsilon$. On a plein d'exemple où c'est faux.

Preuve de convergence

- Hypotèses :
  - la matrice P a une décomposition en valeur propres / vecteur propres.
  - Les vecteurs propres $(v_1, v_2, ..., v_N)$ forment une base.
  - On suppose $\lambda_1 = 1$ et $|\lambda_k| < 1$ pour tout $k > 1$.
- On décompose $x^{(0)}$ sur cette base:

$$x^{(0)} = \sum_{i=1}^{N} \alpha_i v_i$$

- Donc

$$x^{(1)} = \sum_{i=1}^{N} \alpha_i v_i P$$

- Mais comme $v_i P = \lambda_i v_i$ puisque v_i est le vecteur propre associé à λ_i, on a

$$x^{(1)} = \sum_{i=1}^{N} \alpha_i \lambda_i v_i$$

Induction

- Donc
$$x^{(2)} = \sum_{i=1}^{N} \alpha_i \lambda_i v_i P = \sum_{i=1}^{N} \alpha_i \lambda_i^2 v_i$$
- en général $x^{(k)} = \sum_{i=1}^{N} \alpha_i \lambda_i^k v_i$
- On se souvient que $\lambda_1 = 1$ et $|\lambda_i| < 1$ pour $i > 1$. Donc
$$x^{(k)} = \alpha_1 v_1 + \sum_{i=2}^{N} \alpha_i \lambda_i^k v_i$$
- $\sum_{i=2}^{N} \alpha_i \lambda_i^k v_i$ tend vers 0 quand k tend vers ∞.
- Si on évite les cas où $\alpha_1 = 0$ (pas de chance), on trouve que la limite de $x^{(k)}$ est v_1 après renormalisation.
- Donc l'algorithme itératif converge vers le vecteur propre associé à la valeur propre 1.

Vitesse de Convergence

- Vitesse pour que $\sum_{i=2}^{N} \alpha_i \lambda_i^k v_i$ se rapproche de 0 ?
- Rappel : les λ_i sont de plus en plus petits (en module) quand i augmente.
- Donc l'élément le plus important de la somme est le terme en λ_2 (si α_2 est non nul).

Property

La vitesse de convergence de la méthode des puissances est une géométrique de ratio égal à la deuxième valeur propre.

Généralisation à une matrice non stochastique

- On en aura besoin plus tard pour les alternatives à PageRank.
- L'algorithme fonctionne pour des matrices plus générales (on suppose que λ_1 est quelconque, non nul).
- Il faut ajouter une étape de renormalisation de $x^{(n+1)}$
- Si λ_1 est connu, il faut diviser $x^{(n+1)}$ par λ_1 (la preuve est triviale)
- Sinon, il faut diviser par la norme $||x^{(n+1)}||_1$ (preuve plus complexe)

Retour à PageRank

- Il faut que $d^+(i) > 0$. Toute page devrait avoir au moins un successeur dans le graphe du WEB
- Si $d^+(i) > 0$ alors.

$$P[i,j] = A[i,j]/d^+(i)$$

$$\pi(j) = \sum_i \pi(i) P[i,j]$$

- Si $d^+(i) > 0$, alors P est une matrice stochastique.
- Mais en réalité, le graphe du Web contient de nombreuses pages qui n'ont pas de lien de sortie (et donc $d^+(i) = 0$ pour de nombreuses pages i)

Solution PageRank

- Le Surfer Aléatoire
- Le graphe du web est le support d'une marche aléatoire d'un usager.
- Quand il arrive sur une page qui a des url de sortie, il en choisit une au hasard avec la même probabilité pour tous les liens (c'est la marche selon la matrice P) pour sa prochaine étape.
- Quand un utilisateur arrive sur une page sans lien de sortie, il se dirige au hasard sur une page quelconque avec une probabilité identique pour toutes les pages, en tapant une adresse quelconque dans la barre d'adresse du navigateur.
- Quand il n'y pas de lien de sortie, et si il y a N pages dans le graphe du web, on peut aller vers chaque page avec une probabilité $1/N$
- On peut donc rester sur la même page avec la probabilité $1/N$ (donc on ajoute des boucles dans le graphe).
- D'autres stratégies sont possibles (Projet).

Conséquence

- Les lignes nulles de $A[i,j]$ sont remplacées par des lignes $(1/N)\ e$.
- Donc, M est défini comme suit :
  - Si $d^+(i) > 0$ alors $M[i,j] = A[i,j]/d^+(i)$.
  - Sinon $M[i,j] = 1/N$.
- Et on doit résoudre

$$\pi = \pi M \quad et \quad \pi e^t = 1$$

- et M est une matrice stochastique

Mathématiquement

- On définit un vecteur ligne f, tel que $f(i) = 1$ si $\sum_j P(i,j) = 0$ et $f(i) = 0$ sinon

$$M = P + (1/N)\ f^t\ e$$

- Stockage de M = stockage de P + le vecteur f (taille N)

Conditions d'existence de la limite

- et de convergence de l'algorithme des puissances.
- 4 étapes :
- Donner les conditions sur M d'existence de π
- Examiner les conditions de convergence
- Trouver des conditions sur la valeur initiale pour le début de l'algorithme
- Voir les choix de PageRank pour être sûr que le graphe du Web modifié par le surfer aléatoire satisfasse aux conditions de convergence.

Existence de π

Definition

Une matrice stochastique M est associée à une chaine de Markov en temps discret (voir le cours de simulation, ou bibliographie)

- Le problème de l'existence de la solution de $\pi M = \pi$ et $\pi e^t = 1$ est l'existence d'une solution stationnaire pour une chaine de Markov en temps discret
- Et en plus ici cette chaine est finie.
- Les conditions sont connues...

Theorem

Si M est finie, irréductible et primitive alors il existe une solution de $\pi M = \pi$ et $\pi e^t = 1$ et cette solution de dépend pas de l'état initial de la marche et $\lambda_2 < 1$ (la deuxième valeur propre).

Point d'étape

- Si la matrice du graphe du Web est finie, primitive et irréductible, alors
- l'algorithme des puissances marche mais on ne sait pas avec quelle vitesse (on ne sait rien de précis sur λ_2)
- Le graphe du Web est fini (et donc la matrice P aussi).
- Comment prouver que la matrice du graphe du Web est irréductible ???
- Comment prouver que la matrice du graphe du Web est primitive
- Alternative : comment prouver que le PGCD des cycles du graphe du Web est 1.

Echec et expériences

- Le graphe du Web n'est pas fortement connexe.
- Il n'est même pas connexe.
- Petit Monde (voir la théorie des graphes de terrain)
- Et pourtant PageRank marche....

Le retour du surfer aléatoire

- A chaque étape, l'usager a le choix entre taper une page aléatoire par la barre du navigateur ou (avec une probabilité constante α) suivre les url contenues dans la page (et donc suivre P).
- Donc la marche aléatoire sur le graphe du Web s'effectue selon une nouvelle matrice G (guess why ?)

$$G = \alpha M + (1-\alpha)(1/N)e^t e = \alpha P + \alpha(1/N)\, f^t\, e + (1-\alpha)(1/N)e^t e$$

- en effet $e^t e$ est une matrice $N \times N$ dont tous les éléments sont égaux à 1
- PageRank initial $\alpha = 0.85$.

Le surfer aléatoire gagne sur tous les tableaux

Theorem

G est finie, irréductible et primitive, et $\lambda_2(G) < \alpha = 0.85$

- G est finie car le graphe du web est fini.
- G est irréductible car $G[i,j] > 0$ pour tout i,j.
- G est primitive car il est apériodique. En effet $G[i,i] > 0$ pour tout i, donc il y a un cycle de longueur 1 passant par chaque sommet, donc le PGCD vaut 1.
- ce qui assure une vitesse de convergence suffisante.
- Si on diminue α on converge plus vite...

Choix de α

- Si $\alpha = 0$ alors $G = (1/N)e^t e$
- Dans ce cas $\pi = e(1/N)$
- En effet, $\pi e^t = (1/N)e\ e^t = N/N = 1$ et

$$\pi G = (1/N)e(1/N)e^t e = (1/N)^2(e\ e^t)\ e = (1/N)^2 N\ e$$

et donc

$$\pi G = (1/N)\ e = \pi$$

- Donc toutes les pages ont la même note.
- Si on veut avoir une certaine crédibilité dans le ranking, il que α soit proche de 1. Mais cela ralentit la convergence.

Borne sur λ_2

- La borne repose sur un résultat plus général qui relie directement les valeurs propres de G à α.

Theorem

Soit P une matrice dont le spectre est $\{1, \lambda_2, \lambda_3, ..., \lambda_N\}$, alors le spectre de $G = \alpha P + (1-\alpha)e^t v$ est $\{1, \alpha\lambda_2, \alpha\lambda_3, ..., \alpha\lambda_N\}$ pour tout vecteur de probabilit é v.
Rappel : pour PageRank, $v = 1/N\ e$.

Plan de Preuve : on va calculer les deux spectres après changement de base.

- G est stochastique, car P est stochastique et $e^t v$ aussi.
- Donc 1 est valeur propre.

Preuve - suite

Lemma (Technique)

On considère une matrice carrée Q non singulière dont la première colonne est e^t.

$$Q = \left[\begin{array}{c|c} e^t & X \end{array}\right].$$

On cherche la forme de son inverse. On décompose Q^{-1} en une première ligne y et le reste de la matrice Y. Y est une matrice à $N-1$ lignes et N colonnes.

$$Q^{-1} = \left[\begin{array}{c} y \\ \hline Y \end{array}\right].$$

On a $ye^t = 1$, $yX = 0$, $Ye^t = 0$, et $YX = Id_{N-1}$.

Preuve- Lemme

- On fait le produit $Q^{-1}Q$ par bloc. Par définition c'est la matrice identité. Et

$$Q^{-1}Q = \left[\begin{array}{c|c} ye^t & yX \\ \hline Ye^t & YX \end{array}\right] = \left[\begin{array}{c|c} 1 & 0 \\ \hline 0 & Id \end{array}\right]$$

- Les égalités s'en déduisent simplement .

Preuve- Théorème

- On considère le produit PQ (Q est définie dans le lemme).

$$PQ = P * \left[\begin{array}{c|c} & \\ e^t & X \\ & \end{array}\right] = \left[\begin{array}{c|c} & \\ Pe^t & PX \\ & \end{array}\right].$$

- On calcule $Q^{-1}PQ$.

$$Q^{-1}PQ = \left[\begin{array}{c|c} yPe^t & yPX \\ \hline & \\ YPe^t & YPX \end{array}\right].$$

- Puisque P est stochastique, $Pe^t = e^t$ et on opère les simplifications prouvées par le lemme:

$$yPe^t = ye^t = 1 \quad et \quad YPe^t = Ye^t = 0.$$

Preuve- suite

- Et donc après simplifications

$$Q^{-1}PQ = \left[\begin{array}{c|c} 1 & yPX \\ \hline 0 & YPX \end{array}\right].$$

- $Q^{-1}PQ$ avec Q non singulière correspond à un changement de base et cela conserve le spectre.
- 1 est valeur propre à gauche de $Q^{-1}PQ$ associé au vecteur propre $(1, 0, 0, ..., 0)$.
- Donc les valeurs propres $\lambda_2, ..., \lambda_N$ de P sont celle de YPX

Preuve- suite

- On refait le même calcul pour $Q^{-1}GQ$ avec G défini dans le théorème.

$$Q^{-1}(\alpha P + (1-\alpha)e^t v)Q = \alpha Q^{-1}PQ + (1-\alpha)Q^{-1}e^t vQ.$$

- en utilisant les résultats précédents

$$Q^{-1}GQ = \left[\begin{array}{c|c} \alpha & \alpha yPX \\ \hline 0 & \alpha YPX \end{array}\right] + (1-\alpha)\left[\begin{array}{c} y \\ \hline Y \end{array}\right] e^t\, v \left[\begin{array}{c|c} e^t & X \end{array}\right]$$

- Grâce à l'associativité on ordonne les calculs en regroupant e^t avec le terme de gauche et v avec celui de droite.

Preuve- suite

$$Q^{-1}GQ = \left[\begin{array}{c|c} \alpha & \alpha yPX \\ \hline 0 & \alpha YPX \end{array}\right] + (1-\alpha)\left[\begin{array}{c} ye^t \\ \hline Ye^t \end{array}\right]\left[\begin{array}{c|c} ve^t & vX \end{array}\right]$$

- Mais d'après les résultats du Lemme Technique, $ye^t = 1$ et $Ye^t = 0$.
- Et $ve^t = 1$ car v est un vecteur de probabilit/'es.
- Donc

$$Q^{-1}GQ = \left[\begin{array}{c|c} \alpha & \alpha yPX \\ \hline 0 & \alpha YPX \end{array}\right] + (1-\alpha)\left[\begin{array}{c} 1 \\ \hline 0 \end{array}\right]\left[\begin{array}{c|c} 1 & vX \end{array}\right]$$

Preuve- Fin

- On effectue le produit.

$$Q^{-1}GQ = \left[\begin{array}{c|c} \alpha & \alpha yPX \\ \hline 0 & \alpha YPX \end{array}\right] + \left[\begin{array}{c|c} 1-\alpha & (1-\alpha)vX \\ \hline 0 & 0 \end{array}\right]$$

- On fait la somme.

$$Q^{-1}GQ = \left[\begin{array}{c|c} 1 & \alpha yPX + (1-\alpha)vX \\ \hline 0 & \alpha YPX \end{array}\right].$$

- Donc le spectre de $Q^{-1}GQ$ contient 1 et le spectre de αYPX. Mais le spectre de YPX est $\lambda_2, ..., \lambda_N$ (les autres valeurs propres de P). Donc le spectre de $Q^{-1}GQ$ est $\{1, \alpha\lambda_2, ..., \alpha\lambda_N\}$.

Complexité en temps et en espace

$$G = \alpha P + (1 - \alpha)(1/N)e^t e$$

- Donc la matrice G est pleine, elle a N^2 éléments non nuls
- Donc le produit vecteur-matrice a une complexité N^2.
- Mais $(1 - \alpha)(1/N)e^t e$ a une forme très particulière qui permet de faire des calculs numériques avec un stockage linaire.
- Numérique et Symbolique.
- Ne pas stocker explicitement G en mémoire.

Algorithme pour xG

$$\begin{aligned} xG &= x(\alpha P + \alpha(1/N)f^t\ e + (1-\alpha)(1/N)e^t e) \\ &= \alpha xP + \alpha(1/N)x\ f^t\ e + (1-\alpha)(1/N)x\ e^t e) \\ &= \alpha xP + \alpha(1/N)(x\ f^t)\ e + (1-\alpha)(1/N)x\ e^t e \\ &= \alpha xP + \alpha(1/N)(x\ f^t)\ e + (1-\alpha)(1/N)(x\ e^t)e \end{aligned}$$

x est un vecteur de proba, donc $(xe^t) = 1$ et on recalcule $(x\ f^t)$ à chaque itération (c'est une scalaire)

$$xG = \alpha xP + [(1-\alpha)(1/N) + \alpha(1/N)(x\ f^t)]e$$

Algorithme pour xG

- xP est calculé numériquement.
- P est stocké en mémoire (pas G)
- f est stocké en mémoire (surcout linéaire)
- On ajoute une contante $\frac{(1-\alpha)}{N}$ à chaque terme.
- On recalcule $(x\ f^t)$ à chaque itération puisque x change.
- On ajoute $\alpha(1/N)(x\ f^t)$ à chaque terme.
- Donc le cout de stockage de G est du même ordre que le stockage de P
- Et le cout de la multiplication xG est du même ordre que le cout de la multiplication xP

Améliorer son ranking

- On a 2 notes : le contenu (créé par l'auteur), la pertinence (obtenu grâce aux pages qui pointent sur vous)
- Comment améliorer sa note de contenu
  - un peu de bon sens et quelques astuces
  - des conseils de Google
  - mais le calcul exact de la note de contenu n'est pas détaillé
  - Et il évolue toujours (maj BERT récente pour le Francais).

Comment augmenter sa pertinence

- Le pagerank est un entier de 0 à 10. 10 est la note la plue élevée.
- On passe à la partie entière du log de la pertinence et on ajoute 11 (ou 12) pour avoir une information simple
- On avait aussi un affichage graphique (une jauge) de sa pertinence
- Honnêtement : indexer les partenaires et se faire indexer.
- Moins Honnêtement : les links farms : (très présents au début 2000, moins maintenant),
- Principe : un site qui a une pertinence élevé va vendre des liens vers d'autres sites de manière à augmenter leur pertinence.
- Efficace tant que le degré de sortie est faible (la pertinence de la page de sortie est divisé par le degré)
- Marché du lien de sortie (Rank boosting), Procès entre Google et Searchking (compagnie de rank boosting service) en 2003.

Changer la pertinence

- Idée : Il est "naturel" d'ajouter les mots qui peuvent avoir été oublié par le créateur de la page.
- Associe des mots qui ne sont pas dans le texte de la page à une page
- Comment : par les anchor text (ancres) prévu dans HTML pour associer un texte à une URL
- Les robots de visite du web ajoute à une page pointée les textes présents dans les ancres qui pointent.

< a href="URL" > TEXTE < /a >

Google bombing

- Attaque massive
- plusieurs pages envoient le même texte (en général une insulte avec des mots "rares") vers l'URL de la cible
- URL Cible : la victime
- On fait en sorte que les pages attaquantes aient une pertinence pas trop faible (par exemple en se connectant entre elles)
- En général les pages cibles ont déjà une pertinence forte (attaque contre une personne publique ou une institution).

Gros Balourd, Iznogoud

- Ensuite on laisse faire Google pour calculer les pertinences et le dictionnaire.
- Et si Google laisse faire (de moins en moins vrai), taper l'insulte dans la barre de recherche vous amène à la page de la cible
- Exemple : en France vers 2005-2006, Gros Balourd amenait à la page du 1er Ministre de l'époque (cherchez qui) et Iznogoud à un ministre de l'intérieur.
- Il y a des exemples dans de nombreux pays
- Il y a aussi des exemples où des firmes ont tenté d'imposer des qualificatifs élogieux sur leurs produits
- Riposte de Google : Menace de déréférencement, détection de communautés (il faut être nombreux pour que l'attaque marche).
- Riposte de la cible : Attaque et Contre-attaque (Miserable Failure, Michael Moore ou G.W. Bush).

Personnalisation de la recherche

- Faire en sorte que la même requête, dans le même pays, ait deux réponses différentes selon l'usager.
- Les calculs de la matrice Google utilise e/N comme vecteur de téléportation
- Ce vecteur garantit l'irréductibilité et le fait que la matrice G soit primitive
- Donc si on choisit un vecteur de téléportation v qui dépende de l'usager, il doit vérifier ces 2 propriétés pour que l'algorithme converge.

$$G = \alpha M + (1 - \alpha) v^t e$$

Comment conserver ces propriétés

- Si G est irréductible il suffit qu'il existe une page j telle que $G[j,j] > 0$ pour que la matrice soit primitive.
- Comme $||v||_1 = 1$, il existe j, tel $v[j] > 0$.
- Donc $(v^t e)[j,j] > 0$
- Car $(v^t e)[j,j] = v[j]e[j]$ et $e[j] = 1$
- Donc si G est irréductible, elle est aussi primitive (rappel, ceci n'est pas vrai en général)

Irréductibilité

- Le graphe du web est réductible (expérimentalement)
- Puisqu'il existe j tel que $v[j] > 0$, alors pour tout i, $G[j, i] > 0$
- En effet $G[j, i] \geq (v^t e)[j, i] = v[j]e[i] = v[j] > 0$.
- Mais rien ne prouve qu'il existe un chemin de i à j

Theorem

Si pour toute composante fortement connexe du graphe du Web, il existe un sommet (disons k) élément de cette composante fortement connexe tel que $v[k] > 0$ alors G est irréductible

- Preuve en TD
- Pas très pratique à vérifier ou à construire...
- Donc on préfère imposer des vecteurs v tels que $v[i] > 0$ pour tout i.
- Si un utilisateur est peu intéressé par une page, il a quand même une probabilité non nulle de faire un saut direct sur cette page.

Faisabilité

- Pour autant, avoir un v associé à chaque utilisateur demande de calculer à l'avance la pertinence pour tous les utilisateurs qui ont un vecteur v distinct.
- Infaisable si il y a trop de vecteurs v distincts (personnalisés, mais pas trop...)
- Solution Heuristique : On calcule à l'avance un certain nombre de pertinences pour quelques vecteurs v typiques puis on les combine

Solution Kaltix

- StartUp Stanford racheté par Google en 2003
- On considère un nombre fini K de vecteurs v associé au sujet i, par exemple le "top 16 level subjects" pour la classification de l'Open Directory Project.
- On va calculer les vecteurs de pertinence π_i associés aux vecteur v_i
- Puis on associe à chaque usager des coéfficients β_i positifs sommant à 1.

Exemple

- Supposons que vous tapiez la requête "science project"
- Si Kaltix pense que vous êtes enseignant, il va vous donner un poids sur les catégories 7 (Teens and Kids) et 12 (Science)
- Si Kaltix pense que vous êtes ingénieur, il va vous donner un poids sur les catégories 10 (Reference) et 12 (Science)

- Donc un utilisateur va être associé à des coéfficients β_i.
- La pertinence finale est

$$\pi = \beta_1\pi_1 + \beta_2\pi_2 + ... + \beta_K\pi_K$$

- Comment deviner les β_i : données personelles.

Autres Algorithmes: HITS

- HyperText Induces Topic Search
- 1998, J. Kleinberg, IBM
- HITS calcule deux indices pour une page (l'indice d'autorité et l'indice de hub)
- HITS est "query dependent"
- Utilisé par le système CLEVER (IBM) et l'outil de recherche TEOMA

Hubs et Autorités

- Un Hub est une page avec beaucoup de liens d'entrées
- Une Autorité est une page avec beaucoup de liens de sorties
- On va définir des bons Hubs et de bonnes Autorités grâce à la définition circulaire suivante :

Definition

Les bonnes Autorités sont pointées par de bons Hobs et les bons Hubs pointent vers de bonnes Autorités.

Formellement

- Chaque page i a une note d'autorité x_i et une note de hub y_i
- Soit E l'ensemble des arcs du graphe du Web (aucune transformation)
- HITS calcule itérativement des valeurs $x_i^{(k)}$ et $y_i^{(k)}$ pour approcher x_i et y_i par :

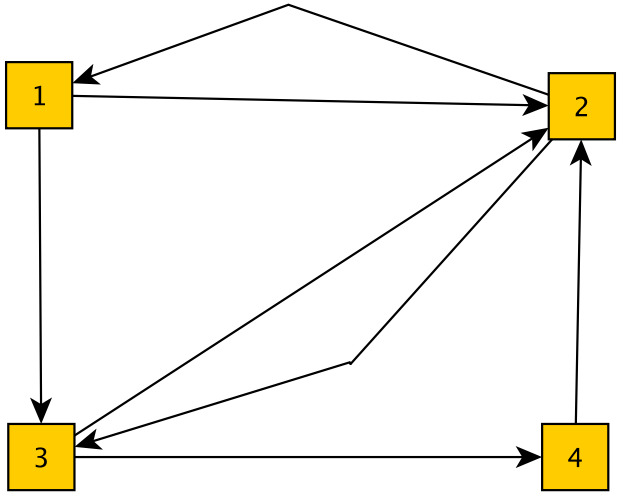
$$x_i^{(k)} = \sum_{j \quad pred \ \ de \ \ i} y_i^{(k-1)}$$

et

$$y_i^{(k)} = \sum_{j \quad succ \ \ de \ \ i} x_i^{(k)}$$

On définit les vecteurs $x^{(k)}$ et $y^{(k)}$.

Exemple



Donc

$$L = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

et

$$x^{(k)} = y^{(k-1)}L \quad et \quad y^{(k)} = x^{(k)}L^t$$

Algorithme HITS

- En ajoutant une initialisation
  1. $y^{(0)} = e$
  2. Jusqu'à convergence faire
     1. $x^{(k)} = y^{(k-1)} L$
     2. $y^{(k)} = x^{(k)} L^t$
     3. k++
  3. normaliser $x^{(k)}$ et $y^{(k)}$

Remarque

- En combinant les deux équations internes, on peut écrire $x^{(k)}$ en fonction de $x^{(k-1)}$
- ou $y^{(k)}$ en fonction de $y^{(k-1)}$.

$$x^{(k)} = y^{(k-1)}L = x^{(k-1)}L^tL$$

et

$$y^{(k)} = x^{(k)}L^t = x^{(k-1)}L\ L^t$$

- L^tL est la matrice des Autorités
- $L\ L^t$ est la matrice des Hubs
- HITS fait donc une méthode des puissances sur les matrices L^tL et $L\ L^t$ plutôt que sur la matrice G

Property

Les deux matrices $L\ L^t$ et L^tL sont symétriques.

- Preuve:

$$(L\ L^t)^t = ((L)^t)^t L^t = L\ L^t$$

- Donc la matrice est symétrique puisqu'elle est égale à sa transposée.

Il est inutile de calculer x et y par la méthode des puissances. Si on a calculé x, on peut obtenir y par $y = xL^t$.

Query Dependent

1. On construit le graphe associé à une requête
2. On ajoute le voisinage à distance 1 (prédécesseurs et successeurs)
3. On obtient les matrices $(L\ L^t)$ et/ou $(L^t\ L)$
4. On obtient les notes x et/ou y

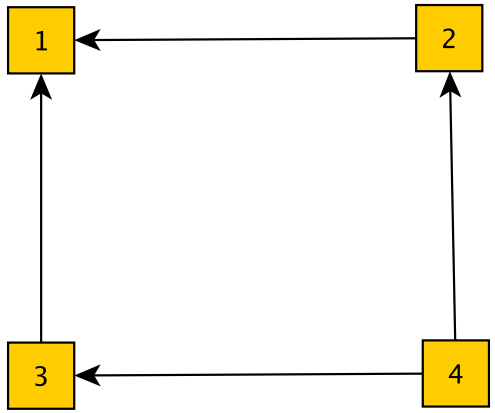
Le graphe est plus petit que celui utilisé dans PageRank. On pourrait donc utiliser des méthodes qui soient moins efficaces en mémoire ou en temps.

Etape 1 en détail

1. on prend la liste des mots avec les opérateurs logiques (par défaut OR)
2. on obtient une liste de pages
3. on complete avec les synonymes et leurs pages associées

Convergence de la méthode des puissance dans HITS

- Posons $B = L^t L$.
- B n'est pas stochastique.
- 1 n'est pas toujours valeur propre.
- Preuve par l'exemple



Exemple

$$L = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

$$L^t L = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Valeurs propres : 0,0,2,2.

Exemple - suite

- Le graphe n'est pas fortement connexe
- Conséquence : si on change $x^{(0)}$, on ne converge pas toujours vers la même solution.
- Si $x^{(0)} = e/4$, $x^{(k)}$ a pour limite $(1/3, 1/3, 1/3, 0)$ (après renormalisation)
- Si $x^{(0)} = (1/4, 1/8, 1/8, 1/2)$, $x^{(k)}$ a pour limite $(1/2, 1/4, 1/4, 0)$ (après renormalisation)

Normalisation

- Il faut normaliser à chaque étape le produit $x^{(k)}B$.
- On divise $x^{(k)}$ par $||x^{(k)}||_1$.
- En effet,

$$||\frac{x^{(k)}}{||x^{(k)}||_1}||_1 = \frac{1}{||x^{(k)}||_1} \quad ||x^{(k)}||_1 = 1$$

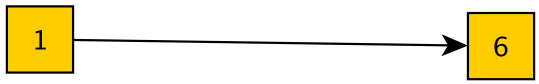
- la vitesse de convergence est celle de $\frac{\lambda_2(B)}{\lambda_1(B)}$.
- Mais pour converger vers une limite unique et indépendante du vecteur initial, il faut que le graphe soit fortement connexe.

Irréductibilité

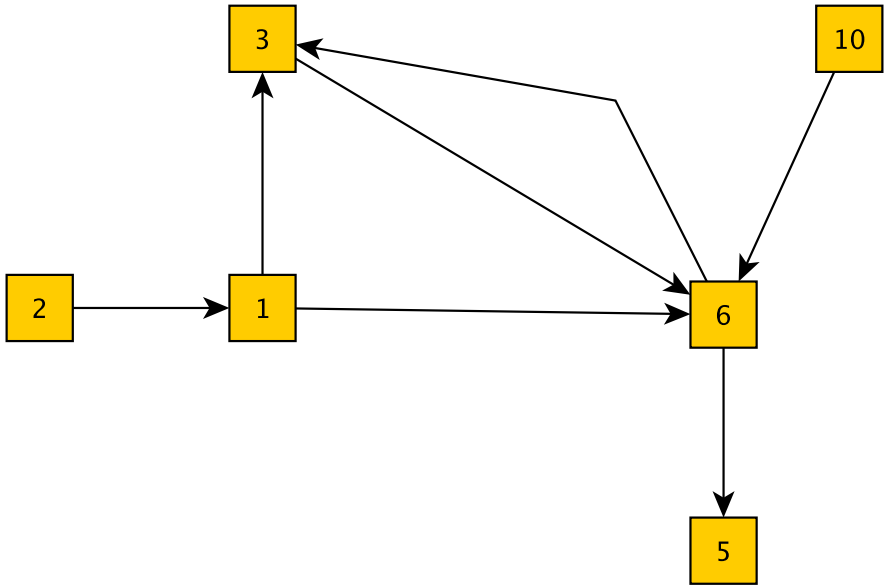
- Même astuce que PageRank :
- Randomized HITS : on mélange HITS et une marche aléatoire uniforme sur les pages de la requête
- On considère la matrice $\psi L^t L + (1-\psi)/N e^t\ e$
- avec ψ entre 0 et 1.

Exemple

Graphe initial



Graphe étendu au voisinage



Exemple - suite

$$L = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad L^t L = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

grace à Scilab.
On va utiliser Randomized HITS avec $\psi = 0.95$.

Exemple - suite

- vecteur propre dominant des autorités
  $x = (0.032, 0.023, 0.3634, 0.1351, 0.4936, 0.023)$
- Il y a des égalités (réglés en FIFO, ou RANDOM)
- le ranking est des autorités est (attention ce sont des numéros de page)
- 6,3,5,1,2,10
- On peut aussi trouver le ranking des hubs:
- 1,3,6,10,2,5

Interprétation de $L^t L$ et de $L\ L^t$

$$L^t L = D_{in} + C_{cit}$$

et

$$L\ L^t = D_{out} + C_{ref}$$

- D_{in} matrice diagonale contenant le degré entrant de i dans $D_{in}[i,i]$
- D_{out} matrice diagonale, degré sortant de i dans $D_{out}[i,i]$
- $C_{cit}[i,j]$ nombre de sommets prédécesseurs de i et de j
- $C_{ref}[i,j]$ nombre de sommets successeurs de i et de j

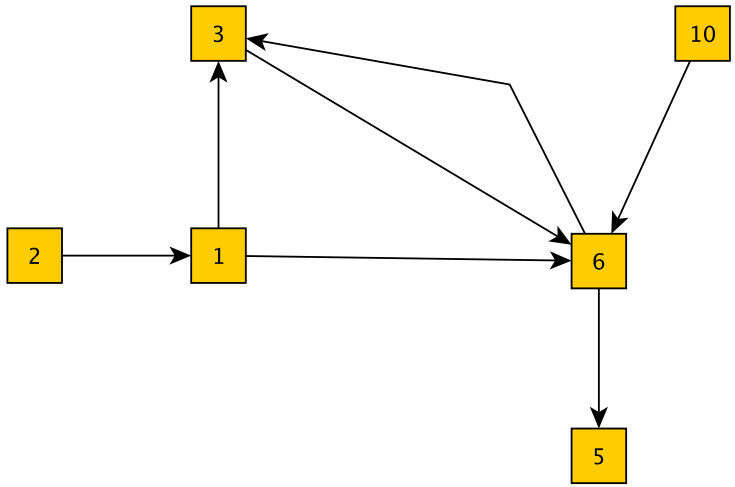
- Connue en bibliométrie.
- $C_{ref}[i,j]$ est le nombre de références en commun dans les articles i et j
- $C_{cit}[i,j]$ est le nombre de travaux qui citent à la fois i et j

SALSA

- Stochastic Approach to Link Structure Analysis
- Query Dependent (comme HITS)
- Repose sur une marche aléatoire (comme PageRank)

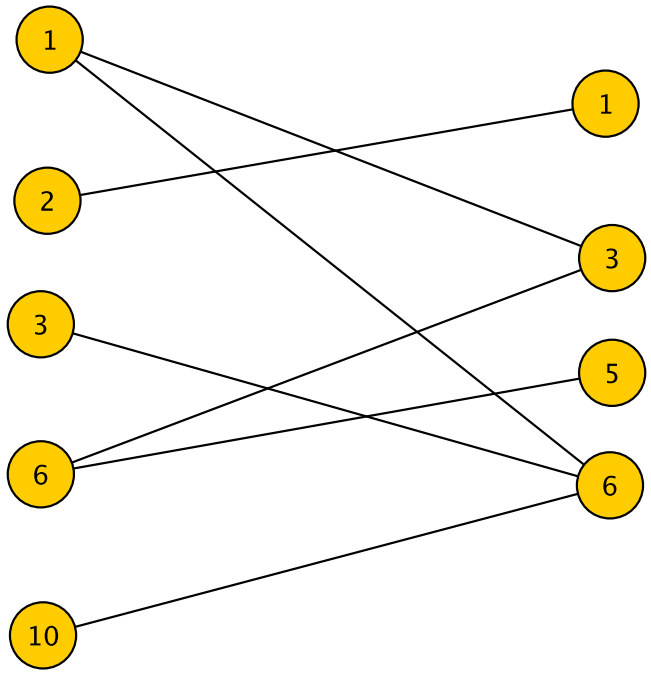
Méthode

1. On commence comme HITS par construire le graphe de la requête étendu aux voisins (sous-graphe du Web)



2. On construit un graphe non orienté biparti (V_1, V_2, E)
   - V_1 est l'ensemble des hubs (sommets du graphe dont le degré sortant > 0).
   - V_2 est l'ensemble des autorités (sommets du graphe dont le degré entrantant > 0). .
   - $E =$ Si il y un arc de V_1 vers V_2 dans le graphe de la requête

- Dans l'exemple, $V_1 = (1, 2, 3, 6, 10)$ et $V_2 = (1, 3, 5, 6)$



- On construit L comme dans *HITS* (matrice d'adjacence du graphe de la rêquete)
- On ajoute deux nouvelles matrices
- L_r : matrice L normalisée à 1 par ligne (quand la ligne est non nulle)
- L_c : matrice L normalisée à 1 par colonne
- On construit H la matrice des hubs de SALSA en faisant le produit $L_r L_c^t$ dont on retire les colonnes/lignes nulles.
- On construit A la matrice des autorités de SALSA en faisant le produit $L_c^t L_r$ dont on retire les colonnes/lignes nulles.

Exemple

- Avec le graphe précédent, on obtient :

$$H = \begin{bmatrix} 5/12 & 0 & 2/12 & 3/12 & 2/12 \\ 0 & 1 & 0 & 0 & 0 \\ 1/3 & 0 & 1/3 & 0 & 1/3 \\ 1/4 & 0 & 0 & 3/4 & 0 \\ 1/3 & 1/3 & 0 & 0 & 1/3 \end{bmatrix}$$

et

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ & 1/2 & 1/4 & 1/4 \\ 0 & 1/2 & 1/2 & 0 \\ 0 & 1/6 & 0 & 5/6 \end{bmatrix}$$

Property

H et A sont des matrices stochastiques.

- Preuve
- L_r est une matrice de somme en ligne égale à 1 ou 0 dont tous les éléments sont positifs
- L_c^t est une matrice de somme en ligne égale à 1 ou 0 dont tous les éléments sont positifs
- Donc le produit $L_r L_c^t$ vérifie la même propriété.
- Si on enlève les lignes nulles, on obtient une matrice stochastique (c'est H).
- Idem pour A.

Irréductible et Primitive

Property

Si le graphe biparti de la requête est connexe, alors H et A sont des chaines de Markov irréductibles.

Sans Preuve

Property

$H[i,i] > 0$ pour tout i (il y a des boucles de longueur 1 donc la chaine est apériodique

Preuve : à la normalisation près,
$H[i,i] = \sum_k L[i,k]L^t[k,i] = \sum_k L[i,k]L[i,k] > 0$ (sinon la ligne est enlevée).
Idem pour A.

Méthode

- On cherche les notes d'autorité et de hub
- Mais comme les deux matrices sont stochastiques on peut utiliser la méthode des puissances
- Et puisqu'elles sont plus petites on peut employer des méthodes moins sensibles à l'occupation mémoire
- Tout doit être fait en temps réel, il faut privilégier la vitesse