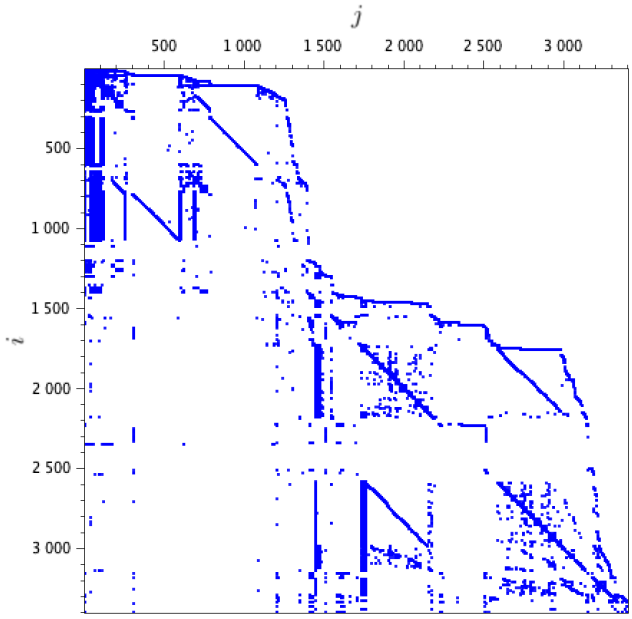
MT94/P23/TD - Valeurs propres - Algorithme PageRank

La recherche d'informations pertinentes sur le Web est un des problèmes les plus cruciaux pour l'utilisation de de ce dernier. Des enjeux économiques colossaux sont en jeu, et diverses multinationales se livrent à de grandes manoeuvres. Le leader actuel de ce marché, Google, utilise pour déterminer la pertinence des références fournies, un certain nombre d'algorithmes dont certains sont des secrets industriels jalousement gardés, mais d'autres sont publics. On va s'intéresser ici à l'algorithme *PageRank*, lequel fait intervenir une énorme matrice (que l'on ne stocke pas vraiment), calculée à partir de la matrice d'adjacence $\mathbf{C}$ du graphe représentant les pages ainsi que les liens entre elles. Par exemple, la figure suivante représente les termes non-nuls de la matrice $\mathbf{C}$ pour un corpus des 3404 premières pages interconnectées obtenues à partir de l'URL `http://www.utc.fr`, où les liens d'une page vers elle-même n'ont pas été pris en compte :



Cette matrice est dite « creuse » car il n'y a que 82627 éléments non nuls (égaux à 1) sur un total de 11587216 éléments (soit un taux de remplissage de 0,71%). Vous pouvez récupérer cette matrice creuse ainsi que les URL des pages correspondantes en chargeant le fichier de données sur Moodle, puis en utilisant la commande `spget` pour récupérer les indices des éléments non-nuls.

```
--> load www_utc_fr.sod
--> ij=spget(C);
--> plot(ij(:,2),ij(:,i),'.')
```

Attention dans le graphique obtenu ci-dessus l'abscisse est l'indice de colonne de la matrice et l'ordonnée l'indice de ligne ! Voici les premiers URL contenus dans la matrice U (matrice de chaines de caractères)

```
--> U(1:10)

  "https://www.utc.fr"
  "https://www.utc.fr/vous-etes/une-future-etudiante.html"
  "https://www.utc.fr/vous-etes/une-etudiante-internationale.html"
  "https://www.utc.fr/vous-etes/une-entreprise.html"
  "https://www.utc.fr/vous-etes/une-diplomee.html"
  "https://www.utc.fr/newsletter.html"
```

```
"https://www.utc.fr/en.html"
"https://www.utc.fr/rechercher-sur-le-site.html"
"https://www.utc.fr/utc.html"
"https://www.utc.fr/formations.html"
...
```

Optimisation du produit dans la méthode de la puissance

Comme on l'a vu en cours, la matrice de Google **P** est calculée à partir de **C** de la manière suivante : on a

$$\begin{aligned} p_{ij} &= q\frac{c_{ij}}{\sum_{j=1}^{n} c_{ij}} + \frac{1-q}{n}, \quad \text{si } \sum_{j=1}^{n} c_{ij} \neq 0, \\ &= \frac{1}{n}, \quad \text{sinon,} \end{aligned}$$

où $q = 0.85$. On peut écrire cette matrice sous la forme

$$\mathbf{P} = q\mathbf{DC} + \frac{1}{n}(\mathbf{e} - \mathbf{qf})\mathbf{e}^{\top},$$

où

— **D** est une matrice diagonale définie par $d_{ii} = (\sum_{j=1}^{n} c_{ij})^{-1}$ si $\sum_{j=1}^{n} c_{ij} \neq 0$ et $d_{ii} = 0$ sinon,

— le vecteur **e** est défini par $\mathbf{e} = (1,\ldots,1)^{\top}$ et le vecteur **f** par $f_i = 0$ si $\sum_{j=1}^{n} c_{ij} = 0$ et $f_i = 1$ sinon.

Pour écrire en Scilab l'algorithme de la puissance on ne va pas calculer et stocker en mémoire cette matrice car bien que **C** soit creuse, ce n'est plus le cas de **P**. Ce qui importe ici est de pouvoir effectuer le produit vecteur ligne $\times$ matrice dans les itérations

$$\boldsymbol{\pi}_{k+1} = \boldsymbol{\pi}_k \mathbf{P}.$$

Il suffit de noter que pour un vecteur ligne $\boldsymbol{\pi}$ donné on a

$$\boldsymbol{\pi}\mathbf{P} = q\boldsymbol{\pi}\mathbf{DC} + \frac{1}{n}\boldsymbol{\pi}(\mathbf{e} - \mathbf{qf})\mathbf{e}^{\top}.$$

D'une part le produit $\boldsymbol{\pi}\mathbf{DC}$ revient à multiplier les composantes de $\boldsymbol{\pi}$ par les éléments diagonaux de **D** puis de multiplier le vecteur ligne obtenu par **C**. D'autre part le deuxième terme est égal au produit scalaire $\frac{1}{n}\boldsymbol{\pi}(\mathbf{e} - \mathbf{qf})$ multiplié ensuite par le vecteur ligne $\mathbf{e}^{\top} = (1,\ldots,1)$. Il suffira donc d'ajouter ce nombre aux composantes du vecteur colonne obtenu précédemment. **Cette lecture associative de l'expression évite de former une matrice pleine !**

Les éléments diagonaux de **D** (vecteur `d` ci-dessous) ainsi que les vecteurs **e** et **f** sont calculés de la manière suivante avec scilab :

```
d = sum(C,2)
d(d>0) = 1./d(d>0)
e = ones(n,1)
f = ones(n,1)
f(d==0) = 0;
```

Travail à réaliser

Déterminer, à l'aide de l'algorithme de la puissance, la valeur propre dominante de P (on la connait déjà, c'est 1) ainsi que le vecteur propre $\hat{\boldsymbol{\pi}}$ associé. En déduire le classement des pages (on pourra utiliser la macro `gsort` de scilab pour classer les composantes de $\hat{\boldsymbol{\pi}}$).

Vous trouverez sur Moodle une fonction `search(U,pi,term)` utilisant la distribution stationnaire trouvée et renvoyant les pages contenant (dans leur URL) un mot-clé donné, classées par ordre de pageRank décroissant.