I Vade-Mecum

Règles :

- 4 séances, temps total : 7h (de 9h à 12h30 et de 14h à 17h30) ;
- travail réalisé en binôme ;
- un travail copié ou effectué en collaboration entre N binômes divise les notes des binômes concernés par N ;
- travail en temps limité : les documents et sources devront être rendus avant 17h30.

Objectif et contenu :

Le projet consiste en la résolution mathématique d'un problème et sa mise en oeuvre informatique. Ce qui est attendu est donc un document qui contient :

- Une présentation formalisée et claire du problème mathématique et de sa solution mathématique,
- Une description algorithmique (en pseudo-code) des solutions implémentées,
- Un programme mettant en oeuvre les solutions implémentées en `Scilab`.
- Un exemple traité et commenté.

Le programme fourni sera réalisé de manière modulaire, commenté, les variables auront des noms explicites, ou au moins univoque. Il sera fourni sous forme compilée et sous forme de codes.
NB : Il est fortement recommandé de procéder, en premier lieu, à une lecture intégrale de l'énoncé.

II Epitome

Une fonctionnalité primordiale des moteurs de recherche de pages web consiste en un tri des résultats associés à une requête par ordre d'importance ou de pertinence. L'énoncé présente un modèle permettant de définir une quantification de cette notion a priori floue et des éléments de formalisation pour la résolution numérique du problème. Il décrit une première approche naturelle à justifier théoriquement et à implémenter mais qui se révèle non satisfaisante dans certains cas. Un raffinement de l'algorithme est donc introduit à fin d'implémentation après justification théorique.

III Introduction : en amont de la classification des résultats

La première fonction (que nous décrivons très rapidement ici à seule fin d'introduire le vocabulaire et les notions) d'un moteur de recherche consiste à répertorier les pages web contenant un ou plusieurs mots-clés : une liste de mots-clés est appelée une *requête*. Le principe de la *fouille de donnée* est simple : les pages web sont copiées en mémoire locale, leur contenu trié par ordre alphabétique afin de pouvoir effectuer rapidement une recherche lexicale. La *réponse* est une liste de pages contenant les mots-clés de la requête. Dans la suite, une réponse contenant $n \in \mathbb{N}^*$ pages sera identifiée avec l'ensemble $\mathbb{N}_n^* = \mathbb{N} \cap [1, n] = \{1, 2, ..., n\}$, i.e. les pages de la réponse sont numérotées dans un ordre arbitraire.

L'énorme quantité de pages constituant une réponse entraîne un problème : le tri automatisé de ces pages selon un ordre de pertinence correspondant assez précisément aux attentes des utilisateurs. L'approche choisie pour ce tri utilise la particularité des documents *hypertexte* contenant des liens vers d'autres pages que nous appèlerons *pointage* par la suite : si une page j contient un lien hypertexte vers une page i, on dit que "j pointe vers i" et on note "$j \to i$".

Étant donné la réponse $\mathbb{N}_n^*$ à une requête, l'ensemble des pointages d'une page vers une autre définit un graphe orienté $\mathcal{G}$ dont $\mathbb{N}_n^*$ est l'ensemble des sommets et l'ensemble des pointages $j \to i$ celui des arètes. Pour $j \in \mathbb{N}_n^*$, on note $l_j \in \mathbb{N}$ le nombre de pages vers lesquelles pointe j. La figure 1 donne un exemple de graphe de pointage entre 14 pages.

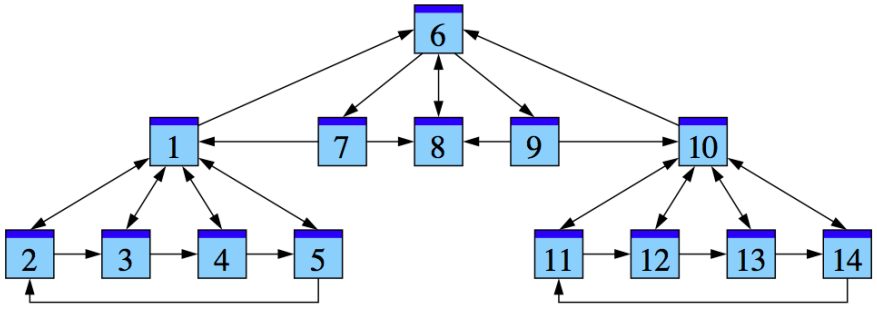

FIGURE 1 – Un exemple de graphe.

Du point de vue du formalisme mathématique, le graphe $\mathcal{G}$ est l'ensemble de couples $(j,i) \in (\mathbb{N}_n^*)^2$ tels que $j \to i$. En `Scilab`, il est préconisé d'utiliser une liste de listes (la liste est ordonnée, ce qui la différencie d'un ensemble). On peut également associée à $\mathcal{G}$ la matrice $M \in \mathcal{M}_n(\{0,1\})$ définie par $m_{ij} = 1$ si $j \to i$ et $m_{ij} = 0$ sinon. Réciproquement, et plus généralement, à toute matrice $M = (m_{ij}) \in \mathcal{M}_n(\mathbb{R}_+)$, on peut associer le graphe orienté :

$$\mathcal{G} = \left\{ (j,i) \in (\mathbb{N}_n^*)^2 \,|\, m_{ij} > 0 \right\}.$$

Enfin, pour $j \in \mathbb{N}_n^*$, on note l_j le nombre de pages vers lesquelles pointe la page j.

Question 1

1. Choisir une méthode d'implémentation d'un graphe orienté en `Scilab` et l'expliquer. A titre d'exemple, implémenter dans `Scilab` le graphe donné par la figure 1.
2. Implémenter les fonctions suivantes :
   - prenant en argument un graphe $\mathcal{G}$ (du type choisi) et renvoyant la matrice associée $M \in \mathcal{M}_n(\{0,1\})$;
   - prenant en argument $M \in \mathcal{M}_n(\mathbb{R}_+)$ et renvoyant le graphe associé $\mathcal{G}$;
   - prenant en argument deux graphes $\mathcal{G}$ et $\mathcal{G}'$ sur $\mathbb{N}_n^*$ et renvoyant le graphe $\mathcal{G} \cup \mathcal{G}'$;
   - prenant en argument un graphe $\mathcal{G}$ sur $\mathbb{N}_n^*$ et $j \in \mathbb{N}_n^*$ et renvoyant l_j ;
   - prenant en argument un graphe $\mathcal{G}$ sur $\mathbb{N}_n^*$ renvoyant $(l_j)_{j=1}^n$.
3. Tester les fonctions sur le graphe donné par la figure 1.

IV Une quantification de la notion de pertinence

On donne une définition naturelle de la pertinence de chaque page du graphe $\mathcal{G}$ obtenu à partir d'une requête. Pour chaque page i, on appelle $\mu_i \in \mathbb{R}_+$ la pertinence de la page i où $\mu = (\mu_i)_{i=1}^n$ est solution de :

$$\mu_i = \sum_{j \to i} \frac{1}{l_j} \mu_j, \quad i \in \mathbb{N}_n^*. \tag{1}$$

Question 2

1. Donner une interprétation de cette définition de la pertinence d'une page dans un graphe. En particulier, on pourra expliquer quelles propriétés doit vérifier une page pour être jugée importante. Donner la définition de la matrice A telle que le système (1) s'écrive $A\mu = \mu$, ou encore $(A - I_n)\mu = 0$.
2. Montrer que $A = (a_{ij})$ vérifie :

$$a_{ij} \geqslant 1 \quad \text{pout tout } i, j, \tag{2}$$

$$\sum_i a_{ij} = 1 \quad \text{pour tout } j. \tag{3}$$

   Que signifie la relation (3) pour la matrice A ?
3. Implémenter une fonction prenant en argument un graphe et renvoyant la matrice A associée. Tester cette fonction sur le graphe donné par la figure 1. Vérifier les propriétés de la question précédente.

Une matrice vérifiant (2) et (3) est appelée une *matrice stochastique*. De même $u = (u_i)_{i=1}^n \in \mathbb{R}^n$ est un *vecteur stochastique* si $\forall i, u_i \geqslant 0$ et $\sum_i u_i = 1$.

V Marche aléatoire sur un graphe : une première méthode itérative

Cette approche de la pertinence des pages dans un graphe mène à la question : existe-t-il une solution de (1) et est-elle unique, en un sens à spécifier ? D'autre part, il est à noter que le résultat d'une requête peut contenir un très grand nombre n de pages et le système (1) est de dimension n^2. Il est donc nécessaire d'utiliser un algorithme performant pour calculer le vecteur de pertinence de ces pages en un temps raisonnable, voire court.

Question 3 Soit A et B deux matrices stochastiques de dimension $n \times n$ et $u \in \mathbb{R}^n$ un vecteur stochastique.

1. Montrer que 1 est valeur propre de A. (On pourra montrer que $1 \in Sp({}^tA)$). Vérifier le résultat sur la matrice associée au graphe de la figure 1 en `Scilab`. Que peut-on en déduire concernant le système (1) ?
2. En supposant que la valeur propre 1 de A est simple (de multiplicité 1), montrer qu'il existe un unique vecteur stochastique solution de (1).
3. Montrer que le rayon spectral de M vaut 1.

Afin de comprendre les conditions sur le graphe de pointage pour qu'il existe un unique vecteur stochastique solution de (1), nous introduisons la notion de *matrice irréductible* et le théorème de Perron-Frobénius.

Définition 1 Une matrice positive $A \in \mathcal{M}_n(\mathbb{R}_+)$ est dite irréductible si $(A + I_n)^{n-1}$ est strictement positive, i.e. à coefficients tous strictement positifs.

Théorème 1 (Perron-Frobenius) Soit $A \in \mathcal{M}_n(\mathbb{R}_+)$ une matrice stochastique irréductible. Alors, le rayon spectral $\rho = 1$ de A est une valeur propre simple de A et le sous-espace propre associé est une droite vectorielle engendrée par un vecteur strictement positif. De plus, la suite $(A^k)_{k\geqslant 0}$ converge.

On peut interpréter la composante u_j d'un vecteur stochastique $u \in \mathbb{R}^n$ comme une probabilité de se trouver sur une page j du graphe et a_{ij} comme une probabilité d'aller de la page j à la page i en suivant un des l_j liens. Ainsi, l'évènement "être sur la page j" est alors représenté par le j-ème vecteur de la base canonique de $\mathbb{R}^n$ noté e_j. En partant de $x^{(0)} = e_j$, la composante $x_i^{(1)}$ de $x^{(1)} = Ax^{(0)}$ est donc la probabilité de se trouver sur la page i du graphe en partant de la page j en suivant la loi de probabilités définie par la matrice A. On introduit naturellement la méthode itérative :

$$\begin{cases} x^{(0)} \text{ vecteur stochastique,} \\ x^{k+1} = Ax^k, \end{cases} \tag{4}$$

Cette méthode modélise donc un *surfeur aléatoire* qui suit les liens à partir de la page où il se trouve suivant la loi de probabilités donnée par A et fournit, après k itérations, le vecteur des probabilités de se trouver sur chaque page du graphe.

Question 4 On représente la page j du graphe $\mathcal{G}$ par le j-ème vecteur de la base canonique de $\mathbb{R}^n$

1. Comment interpréter les images de e_j par $A + I_n$ et $(A + I_n)^k$ par rapport au graphe $\mathcal{G}$? Illustrer les résultats dans `Scilab` avec le graphe donné par la figure 1 et la matrice A associée au système (1). Commenter.
2. Que dire d'un graphe $\mathcal{G}$ associé à une matrice positive irréductible ? Que peut-on dire du graphe donné par la figure 1 ? Vérifier les résultats avec `Scilab`.
3. Montrer que, si A est irréductible, la méthode (4) converge vers le vecteur stochastique solution de (1).

Une des caractéristiques de la matrice A est le relativement faible nombre de ses coefficients qui sont non nuls. Ainsi pour calculer l'itération $x^{(k+1)} = Ax^{(k)}$, on préfère, pour des raisons de performance de l'algorithme, utiliser directement le graphe $\mathcal{G}$: on parcourt toutes les pages émettrices (contenant un lien vers au moins une autre page) et, pour chaque page j émettrice, on parcourt les l_j pages vers lesquelles pointent j.

Question 5

1. Ecrire l'algorithme correspondant à la méthode décrite ci-dessus permettant de calculer $x^{(k+1)} = Ax^{(k)}$ connaissant $x^{(k)}$.
2. Calculer le nombre d'opérations nécessaires à chaque itération.
3. Implémenter la fonction `Scilab` correspondante et la méthode itérative (4) pour un nombre donné d'itérations.
4. Tester la méthode sur le graphe donné par la figure 1. Commenter. Modifier le code pour afficher, en plus du vecteur résultat, les temps de calcul de chacune des itérations.
5. Enrichir le graphe de la figure 1 pour obtenir un graphe dont la matrice A associée n'est pas irréductible. On s'aidera des résultats de la question 4.
6. Tester la méthode itérative (4) sur ce nouveau graphe. Commenter.

VI Raffinement du modèle, convergence et implémentation

Pour résoudre le problème précédemment mis en lumière, on considère un modèle plus raffiné, dépendant d'un paramètre $c \in [0,1]$. A chaque itération :

— avec une probabilité $1-c$, on suit un des pointages de la page de laquelle on part en appliquant la loi de probabilités donnée par la matrice A,

— avec une probabilité c, on choisit une des n pages du graphe de manière équiprobable.

Ceci évite en particulier de se faire piéger par une page sans lien vers une autre page. Plus généralement, elle garantit de pouvoir explorer toutes les pages du graphe, indépendamment des questions de connexité.

Ce nouveau modèle utilise donc la fonction de transition :

$$\begin{array}{rcl} T : \mathbb{R}^n & \to & \mathbb{R}^n \\ x & \to & c\varepsilon + (1-c)Ax \end{array} \tag{5}$$

où A est la matrice stochastique associée à (1) et ε est le vecteur stochastique $\left(\frac{1}{n}, ..., \frac{1}{n}\right) \in \mathbb{R}^n$ correspondant à l'équiprobabilité sur toutes les pages.

Question 6

1. Interpréter la signification de $1/c$ dans le parcours du graphe par la méthode itérative.
2. Montrer la convergence de la méthode itérative (pour toute condition initiale) associée à la fonction de transition T. Proposer un test d'arrêt de la méthode.
3. En utilisant une méthode similaire à celle décrite au-dessus de la question 5, donner un algorithme pour le calcul de $x^{(k+1)} = T(x^{(k)})$ connaissant $x^{(k)}$.
4. Calculer le nombre d'opérations nécessaires à chaque itération.
5. Implémenter la fonction `Scilab` correspondante et la méthode itérative complète pour un nombre donné d'itérations.
6. Tester la méthode sur le graphe donné par la figure 1 avec différentes valeurs de c. Commenter.
7. Tester la méthode itérative sur différents graphes d'intérêt. Commenter.