Université Paris 13

IUT de Villetaneuse

Classement des pages web par Google

Ivan Lin – Alexandre Oudni – Kevin Wong

DUT Informatique

Projet MA

2017-2018

Table des matières

1 Introduction 1

2 Développement mathématiques

3 Algorithme mathématique

3.1 Pseudo-langage . . . . . . . . . .

4 Différents outils informatique utilisé

4.1 Choix des technologies

4.2Explication du développement et du fonctionnement de l’interface graphique

5 Résultat obtenus

6 Conclusion

6.1 Objectifs réalisés

6.2 Perspectives

6.3 Difficultés rencontrées

7 Bibliographie

Annexes

1 Introduction

1.1Présentation

Dans le cadre des Projets S4, nous avons choisi le sujet du classement des pages web par Google. En 1999 Sergey Brin et Larry Page ont mis au point un algorithme mathématique appelé PageRank qui a pour but de déterminer le degré d’importance d’une page Web. Il se démarque des autres moteurs de recherche car il propose un classement des pages en fonction de leur indice de popularité. Celui-ci ne dépend pas du nombre de consultations de la page mais du nombre de liens qui pointent sur cette page à partir d’autres pages Web. En Novembre 2016 le nombre de pages indexées par Google est de 130 mille milliards de pages web.

Le projet a pour but de représenter la structure du Web par une matrice C ∈ Mn,n(R+) , pour cela nous devons trouver une méthode, un algorithme qui va calculer la pertinence de chaque page. On peut représenter la pertinence par un nombre ou un score positif avec la convention que plus le score est grand plus la page est importante.

2 Développement mathématiques

L’idée est de travailler sur le fait qu’il y a des liens reliant les pages les unes aux autres. La première chose à faire :

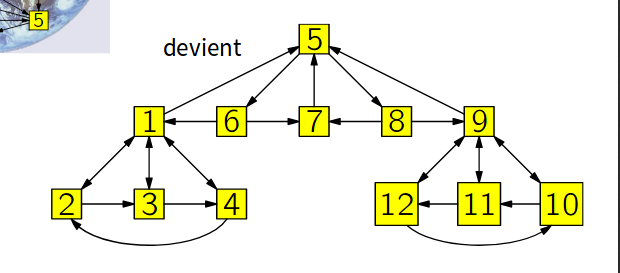
- Numéroter toutes les pages : p1, p2, ….. ,pj

Comme le but est de définir un critère de pertinence de la page, il faudra distinguer « p1 contient un lien qui pointe sur p2 » et « p2 contient un lien qui pointe sur p1 ».

- Si la page pj contient un lien vers la page pi on matérialise cela par une flèche pj → pi .

Considérons un réseau à 12 pages modélisé par le graphe suivant :





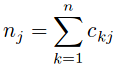
On peut créer la matrice C associée à ce graphe telle que Cij = 1 si la page j pointe sur la page i et 0 sinon. De plus, une page ne peut pas avoir de lien qui pointe sur elle-même, donc Cii = 0.

C =

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

i en ligne et j en colonne.

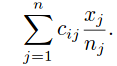
Chaque page i dispose d’un indice xi de façon à pouvoir classer l’ensemble des pages web par score décroissant. Nous devons donc calculer l’ensemble des indices xi en partant du principe qu’un lien de la page j pointant sur la page i contribue au score de cette dernière avec une pondération par xj (une page ayant un indice élevé a plus de poids qu’une page avec un mauvais



indice) et par le nombre total de liens nj présent sur la page

(une page avec beaucoup de lien a moins d’influence).

On a xi =



|  |
| --- |
| x1= x2/2 + x3/2 + x4/2 + x6/2 |
| x2= x1/4 + x4/2 |
| x3= x1/4 + x2/2 |
| x4= x1/4 + x3/2 |
| x5= x1/4 + x7/1 +x9/4 |
| x6= x5/2 |
| x7= x6/2 + x8/2 |
| x8= x5/2 |
| x9= x8/2 + x10/2 +x11/2 + x12/2 |
| x10= x9/4 + x12/2 |
| x11= x9/4 + x10/2 |
| x12= x9/4 + x11/2 |

Prenons le vecteur X =

|  |
| --- |
| x1 |
| x2 |
| x3 |
| x4 |
| x5 |
| x6 |
| x7 |
| x8 |
| x9 |
| x10 |
| x11 |
| x12 |

Prenons une matrice A =

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1/2 | 1/2 | 1/2 | 0 | 1/2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1/4 | 0 | 0 | 1/2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1/4 | 1/2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1/4 | 0 | 1/2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1/4 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1/4 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1/2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1/2 | 0 | 1/2 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1/2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1/2 | 0 | 1/2 | 1/2 | 1/2 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1/4 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1/4 | 1/2 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1/4 | 0 | 1/2 | 0 |

Définition : v est un vecteur propre de A si il existe λ ∈ ℝ tel que Av = λv.

λ est appelée valeur propre associée à v.

Le problème du classement des pages du Web se trouve ramené à la recherche d’un vecteur associé à la valeur λ=1 de la matrice A.

AX = X

A – X = 0

(A – I)X = 0

Cependant, il est possible qu’un réseau contiennent des pages qui n’ont pas de liens (ce qui n’est pas le cas dans l’exemple que nous traitons) ; on se retrouverai donc avec une matrice A contenant des lignes pleines de 0. Dans ce cas , nous rajoutons des liens artificiels pondérés par 1/n qui tend vers 0 et on considère la matrice P telle que pij = (1/nj)Cij + (1/n)dj où d est tel que dj=1 si nj = 0 (si il n’y a pas de liens) et 0 sinon.

La matrice P est la transposée d’une matrice stochastique, pour eT = (1,…..,1) ∈ ℝn, nous avons eT\*P = eT, ce qui signifie que P admet bien la valeur propre 1.

Pour être sûr que la valeur propre est simple, on pose A = αP + (1−α)(1/n)eeT où 0<α<1

Il ne reste plus qu’á résoudre AX = X avec l’algorithme de Gauss.

3 Algorithme mathématique

Fonction Gauss(Tab)

(Pivot de Gauss)

n = taille de Tab

colonne = 0

Tantque ligne < n et colonne < n alors: (on parcourt le tableau en diagonale)

Si Tab[ligne][colonne]=0 alors

Organisé(Tab) (Organise le tableau de façon à obtenir une valeur non nulle comme pivot sur la colonne actuel)

Si Tab[ligne][colonne]=0 alors

colonne = colonne + 1 (Si il n'y a que des 0 sur la colonne alors nous allons sur la prochaine colonne)

Si colonne = n alors (si on dépasse la dernière colonne du tableau)

break

Pour ligne2 allant de ligne+1 à n: (soustraire les lignes en dessous du pivot de façon à obtenir 0 à l'aide pivot sur la colonne actuelle)

Soustraire(Tab,ligne,ligne2,colonne) (colonne est le pivot pour la fonction Soustraire qui soustrait la ligne2 par la ligne )

Si colonne = n alors (si on dépasse la dernière colonne du tableau)

break

colonne = colonne + 1

ligne = ligne + 1

<------------------------------------------------------------------------------->

(Remontée de Gauss)

Tantque ligne >= 0 et colonne >= 0 alors:(on parcourt le tableau en diagonale de bas en haut)

Si Tab[ligne][colonne]!=0 : (si sur la diagonale la valeur est différent de 0 )

Diviser(Tab,ligne,coef) (fonction qui divise toute la ligne par son pivot)

ligne2 = ligne

Tantque ligne2 > 0 faire : (on met à 0 toutes les cases au dessus du pivot à l'aide de la fonction soustraire )

Soustraire(Tab,ligne,ligne2,colonne)

ligne2=ligne2-1

ligne=ligne-1

colonne=colonne-1

return Tab

4 Différents outils informatiques utilisés

4.1 Choix des technologies

Nous avons utilisé le langage python afin de programmer l’algorithme de Gauss. Il s'agissait d'une nouvelle occasion pour nous de nous entraîner à la maîtrise de ce langage, que nous n'avons pas vu en profondeur lors des cours magistraux. En revanche, nous avions pris l’habitude de programmer des algorithmes mathématiques en python pour les TP de modélisation mathématique.

De plus sa syntaxe est simple d’utilisation.

Notre éditeur de texte Sublime Text car il facilite la lisibilité du code et c’est une licence gratuite.

Nous avions effectué le projet sous Ubuntu et Mac OS.

5 Résultats obtenus

5.1 Présentation des résultats

Reprenons notre réseau de départ avec sa matrice associée :

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1/2 | 1/2 | ½ | 0 | 1/2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1/4 | 0 | 0 | ½ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1/4 | 1/2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1/4 | 0 | 1/2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1/4 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1/4 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1/2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1/2 | 0 | 1/2 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1/2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1/2 | 0 | 1/2 | 1/2 | 1/2 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1/4 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1/4 | 1/2 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1/4 | 0 | 1/2 | 0 |

Créons la matrice A = αP + (1−α)(1/n)eeT avec α = 0,85

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0,0125 | 0,4375 | 0,4375 | 0,4375 | 0,0125 | 0,4375 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,225 | 0,0125 | 0,4375 | 0,4375 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,225 | 0,4375 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,225 | 0,0125 | 0,4375 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,225 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,8625 | 0,0125 | 0,225 | 0,0125 | 0,0125 | 0,0125 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,4375 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,4375 | 0,0125 | 0,4375 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,4375 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,4375 | 0,0125 | 0,4375 | 0,4375 | 0,4375 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,225 | 0,0125 | 0,0125 | 0,0125 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,225 | 0,4375 | 0,0125 | 0,0125 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,225 | 0,0125 | 0,4375 | 0,0125 |

AX=X

(A-I)X = 0

On peut créer la matrice A-I

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| -0,9875 | 0,4375 | 0,4375 | 0,4375 | 0,0125 | 0,4375 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,225 | -0,9875 | 0,4375 | 0,4375 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,225 | 0,4375 | -0,9875 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,225 | 0,0125 | 0,4375 | -0,9875 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,225 | 0,0125 | 0,0125 | 0,0125 | -0,9875 | 0,0125 | 0,8625 | 0,0125 | 0,225 | 0,0125 | 0,0125 | 0,0125 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,4375 | -0,9875 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,4375 | -0,9875 | 0,4375 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,4375 | 0,0125 | 0,0125 | -0,9875 | 0,0125 | 0,0125 | 0,0125 | 0,0125 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,4375 | -0,9875 | 0,4375 | 0,4375 | 0,4375 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,225 | -0,9875 | 0,0125 | 0,0125 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,225 | 0,4375 | -0,9875 | 0,0125 |
| 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,0125 | 0,225 | 0,0125 | 0,4375 | -0,9875 |

Lorsqu’on lance l’algorithme de Gauss avec cette matrice on obtient :

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -1,8583 |  |  | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -1 |  |  | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -1 |  |  | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -1 |  |  | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | -1,8084 |  |  | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | -0,9486 |  |  | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | -0,9864 |  |  | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | -0,9486 |  |  | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | -1,8583 |  |  | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | -1 |  |  | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | -1 |  |  | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |  |  | 0 |

Résultats :

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| x1=  1,8583x12 | x2=  x12 | x3=  x12 | x4=  x12 | x5=  1,808x12 | x6=  0,9486x12 | x7=  0,9864x12 | x8=  0,9486x12 | x9=  1,8583x12 | x10=  x12 | x11=  x12 | x12 |

Commentaire : Malgré que notre algorithme ne nous donne pas la pertinence exacte de chaque page, il nous est tout de même possible d’établir un classement de ces pages grâce aux coefficients qui multiplient x12.

- On sait que x1 et x9 est environ 1,85 fois supérieur à x12

- 1,808x12 < 1,8583x12 ---> x12< x5 < x1,x9

- x6,x8 < x7 < x12

- x2 = x3 = x4 = 10 = x11 = x12

On en déduit le classement suivant :

1. Pages 1 et 9

2. Page 5

3. Pages 2,3,4,10,11,12

4. Page 7

5. Pages 6 et 8

6 Conclusion

6.1

L’objectif initial qui était de classer des pages selon leur pertinence est réussi même si il aurait été préférable de réussir á obtenir la pertinence exacte de toutes les pages.

6.2

Nous avions eu des problèmes pour bien modéliser le sujet du projet, en particulier lors de la création des matrices. C’est lors de multiples rendez-vous avec notre tutrice de projet que nous avons réussi à régler ce problème.

Nous avons utilisé python3 pour programmer, nous avons remarqué que cette version de python cause des erreurs de calcul comme 1 – 0.85 = 0.1500000000002 cela nous a créé des problèmes d’imprécision à la longue dans notre algorithme

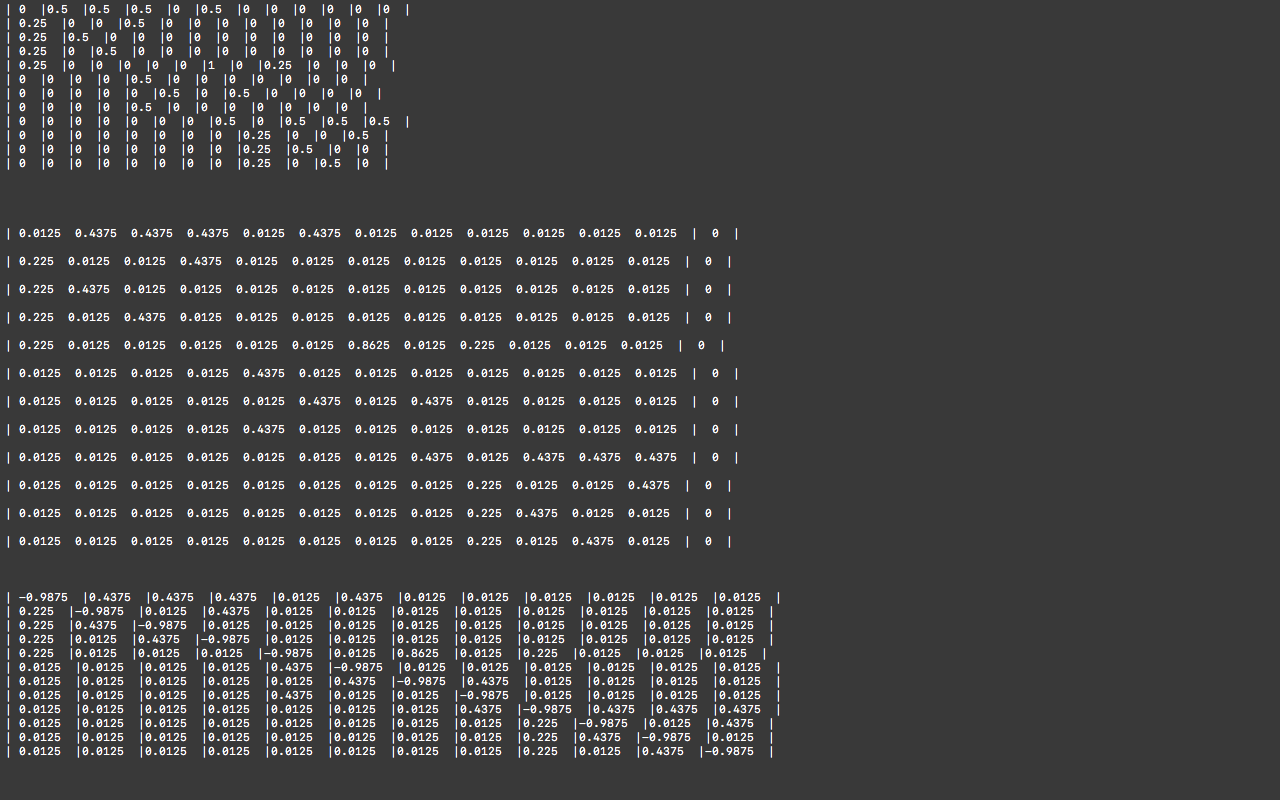
Nous avions entamé tardivement le cours sur les vecteurs propres. Heureusement cette difficulté a été résolu grâce à la tutrice.

Le sujet au départ était compliqué à appréhender, mais au fur et à mesure de l’avancée du projet, le sujet nous semblait plus clair petit à petit

7 Bibliographie

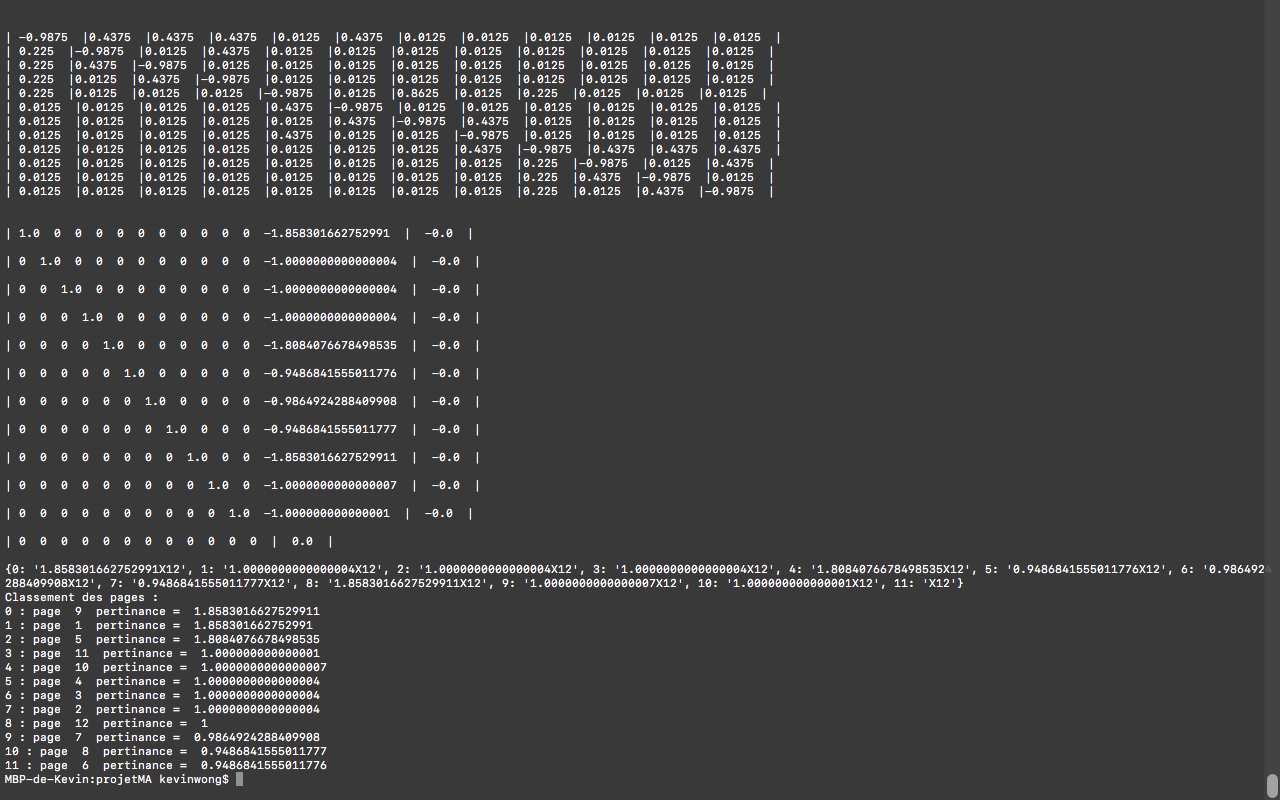
https://fr.wikipedia.org/wiki/%C3%89limination\_de\_Gauss-Jordan

Annexes



Cette image montre l’exemple traité dans ce rapport.

Le premier tableau montre la matrice de départ, le second montre la matrice A = αP + (1−α)(1/n)eeT avec α = 0,85, le troisième lui soustrait la matrice identité.



Le dernier tableau montre la matrice après le lancement de l’algorithme de Gauss ainsi que les résultats obtenus pour la pertinence de chaque page.

La pertinence de x12 a été mise à 1 manuellement pour l’affichage du classement.

Lancement du programme avec un réseau de 30 pages

