A2DI - Random walk

Quentin Baert - Antonin Carette

Master 2 Informatique - MoCAD

Question 1

Le calcul du vecteur propre dominant via l'algorithme itératif **power method** aboutit au bout du nombre d'itérations suivant :

- Pour la matrice **A** : 2 itérations,
- Pour la matrice **B** : 113 itérations

Ainsi, nous constatons qu'il faut plus d'itérations afin de trouver le vecteur propre dominant de **B**.

Les vecteurs propres dominants trouvés à l'aide de la **power method** sont les suivants :

- $q_A^T = (0.70710678 \;\; 0.70710678)$
- $q_B^T = (0,44721168 \;\; 0,89442815)$

On note leurs valeurs propres associées (trouvées à l'aide de la fonction `numpy.linalg.eig()`). Elles sont de la forme $(\lambda_{n-1} \;\; \lambda_n)$ où λ_n est la valeur propre associée au vecteur propre dominant et λ_{n-1} est la valeur propre associée au vecteur propre précédent.

- $v_A^T = (-1 \;\; 10)$
- $v_B^T = (-9 \;\; 10)$

On sait que le rapport $|\frac{\lambda_{n-1}}{\lambda_n}|$ détermine la vitesse de convergence de la **power method**. Plus ce rapport est proche de 0, plus la convergence sera rapide. On observe les rapports suivants qui confirment nos observations :

- Pour **A** : $|\frac{-1}{10}| = 0.1$
- Pour **B** : $|\frac{-9}{10}| = 0.9$

Question 2

La matrice d'adjacence qui correspond au graphe des 3 documents est la suivante :

$$W = \begin{array}{c c} & \begin{array}{ccc} d_1 & d_2 & d_3 \end{array} \\ \begin{array}{c} d_1 \\ d_2 \\ d_3 \end{array} & \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \end{array}$$

L'algorithme itératif **power iteration** permet d'extraire le `rank` de chacun des documents. Cependant, la méthode présentée dans le cours ne donne pas un vecteur de probabilité (dont la somme des valeurs vaut 1). Afin de rectifier ce biais, nous avons ajouter une méthode `make_proba_vector()` au processus.

Pour $\alpha = 0.85$ on obtient le `rank` suivant :

- $\texttt{rank}(d_1) = 0.11538462$
- $\texttt{rank}(d_2) = 0.4423076$
- $\texttt{rank}(d_3) = 0.44230769$

En faisant varier les valeurs de α, on obtient les valeurs suivantes (arrondies au 10^e près) :

α	$\texttt{rank}(d_1)$	$\texttt{rank}(d_2)$	$\texttt{rank}(d_3)$
0.0	0.33	0.33	0.33
0.1	0.32	0.34	0.34
0.2	0.31	0.35	0.35
0.3	0.29	0.35	0.35
0.4	0.27	0.36	0.36
0.5	0.25	0.38	0.38
0.6	0.22	0.39	0.39
0.7	0.19	0.43	0.43
0.8	0.14	0.43	0.43
0.9	0.08	0.46	0.46
1.0	0.0	0.5	0.5

On constate que plus α augmente, plus le `rank` de d_1 baisse alors que ceux de d_2 et d_3 augmentent. On en conclue que plus α augmente, plus on considère les `rank` des documents loin dans le temps.

Ainsi, pour $\alpha = 0$, on considère les `rank` au temps $t = 0$: il est équiprobable de se trouver sur n'importe lequel des trois documents.
Pour $\alpha = 1$, $t = +\infty$: il est impossible de retourner sur le document d_1 car aucun des autres documents ne pointe sur lui et qu'il est improbable que nous soyons toujours dessus. Les documents d_2 et d_3 pointent l'un sur l'autre et possèdent le même nombre de liens entrant (à savoir 2), il est donc équiprobable de se trouver sur l'un ou l'autre.