Question 1

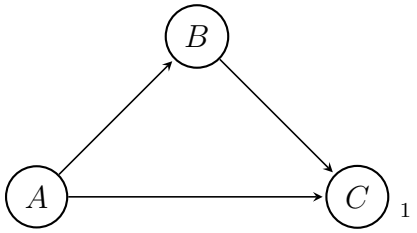
LJ Brown

May 3, 2018

1 Question

1. (25/25 points) Consider the following web pages and the set of web pages that they link to: Page A points to pages B and C. Page B points to page C. All other pages have no outgoing links. Apply the PageRank algorithm to this subgraph of pages. Assume $\alpha = 0.15$. Simulate the algorithm for three iterations.

Disconnected Webgraph



[1]

2 PageRank Algorithm

The "pageRank algorithm" attempts to gauge the importance of each website in a webgraph by ranking each node according to the number and "quality" of its inlinks.

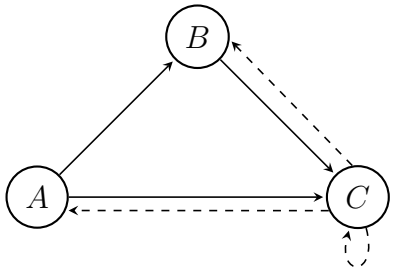
Imagine a web surfer who begins at some webpage or node in the "Disconnected Webgraph" above, and performs a random walk by clicking on different links. At every node the surfer picks a random outlink or edge to

[1] Webpage C, in the "Disconnected Webgraph" is called a "dangling node".

follow, each outlink from the same webpage having an equal chance being selected. In the "Disconnected Webgraph" above, the surfer will eventually become trapped at node C.

The "pageRank algorithm" removes the "dangling node" trap by adding connections to nodes with no outlinks to every single other node in the graph. By connecting the "dangling nodes" to every other node, not only would the surfer be able to click forever, but the surfer would continue to revisit every single node in the graph during this infinite walk.

Strongly Connected Webgraph

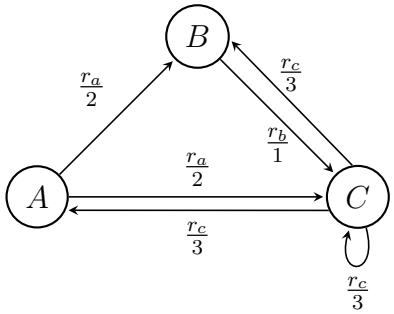


However, although the surfer would eventually revisit every single website, the probability that any particular website is visited next may be different. This probability is the websites "rank".

A nodes rank, r_i, is defined by summing the inlink values of all nodes pointing to it,

$$r_i = \sum_{k \to i} \frac{r_k}{o_k}$$

Where o_k is the number of outlinks for node k.



Ignoring α or the damping factor in the question above, the definition above for a websites rank, r_i, gives the following equations,

$$r_a = \frac{r_c}{3} \tag{1}$$

$$r_b = \frac{r_a}{2} + \frac{r_c}{3} \tag{2}$$

$$r_c = \frac{r_a}{2} + \frac{r_b}{1} + \frac{r_c}{3} \tag{3}$$

These equations have no unique solution (3 equations with 3 unknowns). But since we are kind of looking for the probability of a websites selection you can add the additional constraint that all the ranks sum to one in order to solve the system,

$$r_a + r_b + r_c = 1 \tag{4}$$

Writing this system in matrix form, performing Gauss-Jordan elimination and solving for website ranks r_a, r_b, and r_c,

$$\begin{matrix} r_a + 0 - \frac{r_c}{3} = 0 \\ \frac{r_a}{2} - r_b + \frac{r_c}{3} = 0 \\ \frac{r_a}{2} + r_b - \frac{2r_c}{3} = 0 \\ r_a + r_b + r_c = 1 \end{matrix} \rightarrow \begin{bmatrix} 1 & 0 & \frac{-1}{3} \\ \frac{1}{2} & -1 & \frac{1}{3} \\ \frac{1}{2} & 1 & \frac{-2}{3} \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} r_a \\ r_b \\ r_c \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \rightarrow \left[\begin{array}{ccc|c} 1 & 0 & \frac{-1}{3} & 0 \\ \frac{1}{2} & -1 & \frac{1}{3} & 0 \\ \frac{1}{2} & 1 & \frac{-2}{3} & 0 \\ 1 & 1 & 1 & 1 \end{array}\right]$$

$$\left[\begin{array}{ccc|c} 1 & 0 & \frac{-1}{3} & 0 \\ \frac{1}{2} & -1 & \frac{1}{3} & 0 \\ \frac{1}{2} & 1 & \frac{-2}{3} & 0 \\ 1 & 1 & 1 & 1 \end{array}\right] \rightarrow \left[\begin{array}{ccc|c} 1 & 0 & 0 & \frac{2}{11} \\ 0 & 1 & 0 & \frac{3}{11} \\ 0 & 0 & 1 & \frac{6}{11} \\ 0 & 0 & 0 & 0 \end{array}\right]$$

$$r_a = \frac{2}{11},\ r_b = \frac{3}{11},\ r_c = \frac{6}{11}$$

If there was no damping factor ($alpha = 0$) then these solutions for r_a, r_b, and r_c would be the page ranks corresponding to the "Strongly Connected Webgraph".

3 Link Matrix

Another way to represent this problem involves a "Link Matrix" or adjacency matrix representation of the "Disconnected Webgraph". [2] The Link Matrix, $L_{disconnected}$, is an $n \times n$ Matrix (n is the number of nodes) where a 1 is placed at element l_{ij} if there is an edge from node i to node j,

$$L_{disconnected} = \begin{matrix} \\ a \\ b \\ c \end{matrix} \begin{matrix} \begin{matrix} a & b & c \end{matrix} \\ \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \end{matrix}$$

The elements of $L_{disconnected}$ containing 1's represent a path of length 1 from from node i to node j. This matrix has the property that by raising it to the kth power you find all paths of length k within its graph. For example to find paths of length 2,

$$L^2_{disconnected} = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} = \begin{matrix} \\ a \\ b \\ c \end{matrix} \begin{matrix} \begin{matrix} a & b & c \end{matrix} \\ \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \end{matrix}$$

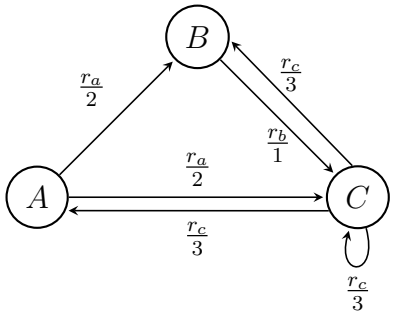
Meaning the only path of length 2 in the "Disconnected Webgraph" is from node a to node c. If you go further and raise $L_{disconnected}$ to the 3rd power you will find a matrix of all zeros, implying that there are no paths of length 3 in the disconnected graph given. But if you make another link matrix, L, from the "Strongly Connected Webgraph" and raise that to the kth power you will always get a positive matrix out for any value of $k >= 1$.

$$L^2 = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} = \begin{matrix} \\ a \\ b \\ c \end{matrix} \begin{matrix} \begin{matrix} a & b & c \end{matrix} \\ \begin{pmatrix} 1 & 1 & 2 \\ 1 & 1 & 1 \\ 1 & 2 & 3 \end{pmatrix} \end{matrix}$$

This is the reason that the surfer can continue clicking forever in the "Strongly Connected Webgraph".

[2] The transpose of this matrix works too; The transition matrix becomes left stochastic and the starting probability vector becomes a column vector.

4 Transition Matrix



The "Transition Matrix", T_0, or weighted matrix is like the link matrix except takes into account the outlinks, o_i, for a given edge. o_i is the number of non zero entries in a row, i, of the link matrix. And the non zero values of the link matrix are replaced by $\frac{1}{o_i}$ for every element in T_0.

$$T_0 = \begin{array}{c} \begin{matrix} a & b & c \end{matrix} \\ \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} \end{array} \begin{matrix} a \\ b \\ c \end{matrix}$$

Every row of the "Transition Matrix" sums to 1, making it a "Right Stochastic Matrix". This matrix can be used to model the surfers movement. Given a starting probability vector [3], $\vec{v_a}$, with the surfer starting on webpage a, you can find the probability that the surfer will be on webpage a on his first visit (which should be 1),

$$\vec{v_a}T_0^1 = \begin{array}{c} \begin{matrix} a & b & c \end{matrix} \\ \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \end{array} \begin{array}{c} \begin{matrix} a & b & c \end{matrix} \\ \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} \end{array} \begin{matrix} a \\ b \\ c \end{matrix} = \begin{array}{c} \begin{matrix} a & b & c \end{matrix} \\ \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \end{array}$$

Or the different probabilities after 2 clicks,

$$\vec{v_a}T_0^2 = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{6} & \frac{1}{6} & \frac{2}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{9} & \frac{5}{18} & \frac{11}{18} \end{pmatrix} = \begin{pmatrix} \frac{1}{6} & \frac{1}{6} & \frac{2}{3} \end{pmatrix}$$

[3] $\vec{v_a}$ is a row vector because the transition matrix is right stochastic.

Or finally the website ranks, or,

$$\lim_{k\to\infty} \vec{v_a} T_0^k = \begin{pmatrix} \frac{2}{11} & \frac{3}{11} & \frac{6}{11} \end{pmatrix}$$

Notice the answer comes out to the same one obtained from the system of equations above,

$$r_a = \frac{2}{11},\ r_b = \frac{3}{11},\ r_c = \frac{6}{11}$$

$$\vec{r} = \begin{pmatrix} \frac{2}{11} & \frac{3}{11} & \frac{6}{11} \end{pmatrix}$$

$\vec{r}$ will be the same for any starting vector (in this case $\vec{v_a}$). $\vec{r}$ is also the left eigenvector for the eigenvalue, $\lambda = 1$, for the Transition Matrix, T_0, created from the "Strongly Connected Webgraph". It is the left eigenvector because the Transition Matrix is right stochastic. If the matrix were left stochastic then $\vec{r}$ would be the right eigenvector for $\lambda = 1$.

$$L = \begin{array}{c} \begin{matrix} a & b & c \end{matrix} \\ \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \end{array} \begin{matrix} a \\ b \\ c \end{matrix}, T_0 = \begin{array}{c} \begin{matrix} a & b & c \end{matrix} \\ \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} \end{array} \begin{matrix} a \\ b \\ c \end{matrix}, \vec{r} = \begin{pmatrix} \frac{2}{11} & \frac{3}{11} & \frac{6}{11} \end{pmatrix}$$

5 Damping Factor

However if a damping factor is added, $\alpha = 0.15$, is included then the flow equations and graph are changed and the the Transition Matrix, $T_{0.15}$, becomes,

$$T_{0.15} = (0.85) \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} + (0.15) \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} = \begin{pmatrix} 0.05 & 0.475 & 0.475 \\ 0.05 & 0.05 & 0.09 \\ 0.33\ldots & 0.33\ldots & 0.33\ldots \end{pmatrix}$$

Or for different values of α,

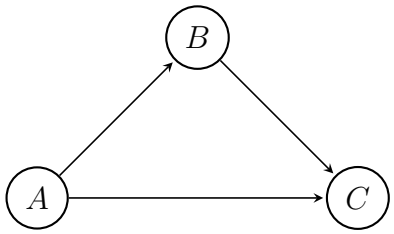
$$T_\alpha = (1-\alpha) \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} + (\alpha) \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix}$$

Adding α makes it possible for the surfer to jump to a random website in the graph.

6 Question 1 Answer

1. (25/25 points) Consider the following web pages and the set of web pages that they link to: Page A points to pages B and C. Page B points to page C. All other pages have no outgoing links. Apply the PageRank algorithm to this subgraph of pages. Assume $\alpha = 0.15$. Simulate the algorithm for three iterations.

If you choose the starting vector, $\vec{v}$, to be the left eigenvector of $T_{0.15}$ corresponding to the eigenvalue $\lambda = 1$, then the probabilities will not change for each iteration.



$$T_{0.15} = \begin{pmatrix} 0.05 & 0.475 & 0.475 \\ 0.05 & 0.05 & 0.09 \\ 0.33\ldots & 0.33\ldots & 0.33\ldots \end{pmatrix}$$

$$\vec{v} \approx \begin{pmatrix} 0.19757929 & 0.28155074 & 0.52086996 \end{pmatrix}$$

Then for $k = 3$,

$$\vec{r} = \vec{v}T_{0.15}^3 = \vec{v} \approx \begin{matrix} a & b & c \\ (0.19757929 & 0.28155074 & 0.52086996) \end{matrix}$$