BDA – Practical Sessions

Session 4

PageRank

Jen Alchimowicz

Agenda for today

1. Quick recap
2. Exercises from the book (From Session 5 on moodle)
3. PageRank with NetworkX example

Agenda for today

1. Quick recap

2. Exercises from the book (From Session 5 on moodle)

3. PageRank with NetworkX example

PageRank

PageRank is a function that assigns a real number to each page in the Web. The intent is that the higher the PageRank of a page, the more 'important' it is.

Web as a directed graph:

- *Nodes = pages*
- *Edges = links*

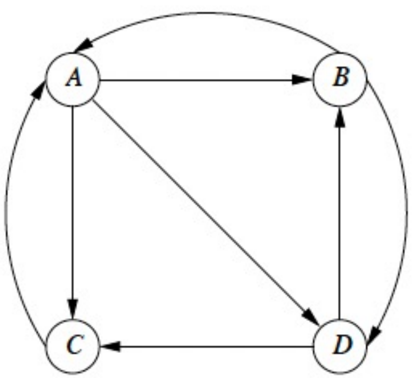


Figure 5.1: A hypothetical example of the Web

Transition matrix:

- *Columns = output probabilities*
- *Columns must add up to 1*
- *Rows = input probabilities*

$$
\begin{array}{cc}
 & \begin{array}{cccc} a & b & c & d \end{array} \\
M = & \begin{bmatrix} 0 & 1/2 & 1 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix}
\end{array}
\begin{array}{c} a \\ b \\ c \\ d \end{array}
$$

$= 1$

PageRank

$$v = \begin{bmatrix} 0.14 \\ 0.23 \\ 0.01 \\ \dots \\ 0.11 \end{bmatrix}$$

PageRank vector **v**:

- Holds a PageRank score for each element in the web
- Shape of the vector: $[n \times 1]$, where n = # pages in web
- Initialised with $1/n$ for each component (equal rank)

The algorithm:

- For x number of steps:
  - Update $v = Mv$
  - If the update was small, that is v was already very close to Mv, stop the iteration.

$$\begin{array}{ccccc} v & = & M & & v \\ \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix} & = & \begin{bmatrix} 0 & 1/2 & 1 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix} & & \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix} \end{array}$$

Shapes: $[n \times 1] \quad = \quad [n \times n] \quad [n \times 1]$

PageRank

The convergence of $v = Mv$ depends on two conditions:

1. The graph is *strongly connected*; that is, it is possible to get from any node to another node
2. There are no *dead ends*: nodes that have no edges out

Why do we need the iterative approach?

- The equation $v = Mv$ can be solved through Gaussian elimination
- Gaussian elimination is $O(n^3)$ in complexity
- Unfeasible for even medium sized graphs
- Gaussian elimination has an infinite number of solutions (we can multiply v by any constant and get another solution). In iterative approach we introduce the constraint that the sum of components of v must equal to 1, giving us a unique solution.

Linear algebra interpretation:

- In convergence, when $v = Mv$ holds true, v is an eigenvector of M
- Eigenvectors of matrices satisfy: $v = \lambda Mv$
- Because M is stochastic (it's columns add up to 1), v is the principal eigenvector and it's associated eigenvalue is 1.

PageRank – Dead ends and Spider traps

The convergence of $v = Mv$ depends on two conditions:

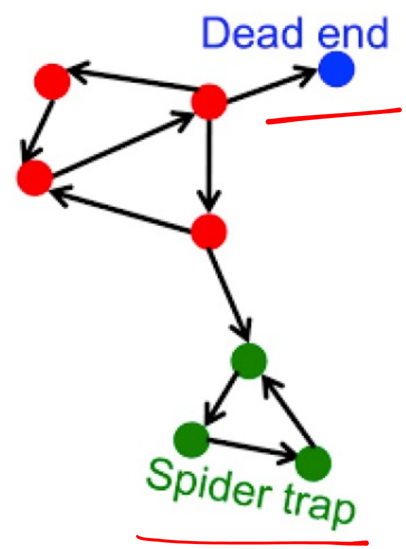
1. The graph is *strongly connected*
2. There are no *dead ends*: nodes that have no edges out

Solution: Taxation (or 'teleportation')

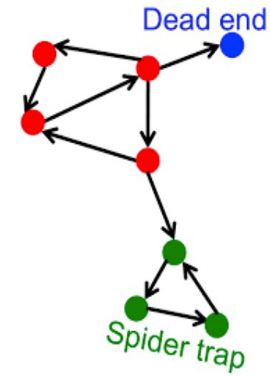
$$v = \beta M v + (1 - \beta)\frac{e}{n}$$

Case when with probability β the random surfer decides to follow an out-link as as usual.

Case when with probability $(1 - \beta)$ the random surfer 'teleports' to a random page *(e is a vector of all 1s)*



PageRank – Dead ends and Spider traps



Solution: Taxation (or ‘teleportation’)

$$v = \underline{\beta M v} + \underline{(1-\beta)\frac{e}{n}}$$

Let’s take $\beta = 0.8$:

$$v \quad = \quad \beta \quad M \quad v \quad + (1-\beta)\ \frac{1}{n}\ e$$

$$\begin{bmatrix}0.25\\0.25\\0.25\\0.25\end{bmatrix} = \underline{0.8}\begin{bmatrix}0 & 1/2 & 1 & 0\\1/3 & 0 & 0 & 1/2\\1/3 & 0 & 0 & 1/2\\1/3 & 1/2 & 0 & 0\end{bmatrix}\begin{bmatrix}0.25\\0.25\\0.25\\0.25\end{bmatrix} + \underline{0.2 * \frac{1}{4}}\begin{bmatrix}1\\1\\1\\1\end{bmatrix}$$

Shapes: $[n\times 1] = [n\times n]\ [n\times 1] + [n\times 1]$

Agenda for today

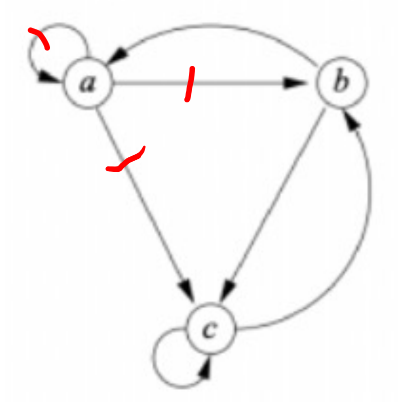
1. Quick recap

2. Exercises from the book (From Session 5 on moodle)

3. PageRank with NetworkX example

HW5

1. **(Exercise 5.1.1 MMDS book)** Compute the PageRank of each page in Fig. 5.7, assuming no taxation.



$$v = Mv$$

$$\begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} = \begin{bmatrix} 1/3 & 1/2 & 0 \\ 1/3 & 0 & 1/2 \\ 1/3 & 1/2 & 1/2 \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

$$M = \begin{bmatrix} 1/3 & 1/2 & 0 \\ 1/3 & 0 & 1/2 \\ 1/3 & 1/2 & 1/2 \end{bmatrix}$$

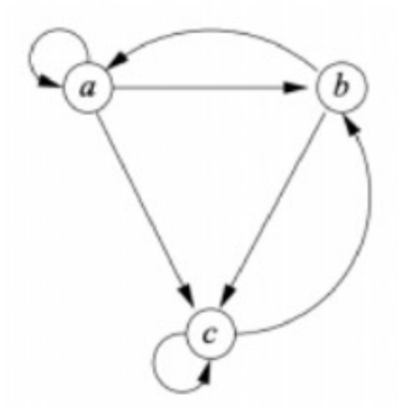
1

$$\begin{cases} r_1 = r_1/3 + r_2/2 \\ r_2 = r_1/3 + r_3/2 \\ r_3 = r_1/3 + r_2/2 + r_3/2 \\ r_1 + r_2 + r_3 = 1 \end{cases}$$

HW5

2. **(Exercise 5.1.2 MMDS book)** Compute the PageRank of each page in Fig. 5.7, assuming taxation with $\beta = 0.8$

$$v = \beta M v + (1-\beta) * \frac{1}{n} * e$$



$$\begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} = 0.8 * \begin{bmatrix} 1/3 & 1/2 & 0 \\ 1/3 & 0 & 1/2 \\ 1/3 & 1/2 & 1/2 \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} + 0.2 * \frac{1}{3} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} = \begin{bmatrix} 4/15 & 2/5 & 0 \\ 4/15 & 0 & 2/5 \\ 4/15 & 2/5 & 2/5 \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} + \begin{bmatrix} 1/15 \\ 1/15 \\ 1/15 \end{bmatrix}$$

$$M = \begin{bmatrix} 1/3 & 1/2 & 0 \\ 1/3 & 0 & 1/2 \\ 1/3 & 1/2 & 1/2 \end{bmatrix}$$

$$\left\{ \begin{aligned} r_1 &= r_1 \frac{4}{15} + r_2 \frac{2}{5} + \frac{1}{15} \\ r_2 &= r_1 \frac{4}{15} + r_3 \frac{2}{5} + \frac{1}{15} \\ r_3 &= r_1 \frac{4}{15} + r_2 \frac{2}{5} + r_3 \frac{2}{5} + \frac{1}{15} \\ r_1 &+ r_2 + r_3 = 1 \end{aligned} \right.$$

HW5

3. **(Exercise 5.2.1 MMDS book)** Suppose we wish to store an $n \times n$ Boolean matrix (0 and 1 elements only). We could represent it by the bits themselves, or we could represent the matrix by listing the positions of the 1's as pairs of integers, each integer requiring $\log_2(n)$ bits. The former is suitable for dense matrices; the latter is suitable for sparse matrices. How sparse must the matrix be (i.e., what fraction of the elements should be 1's) for the sparse representation to save space?

$[n \times n]$ - size of the matrix
N - number of 1's

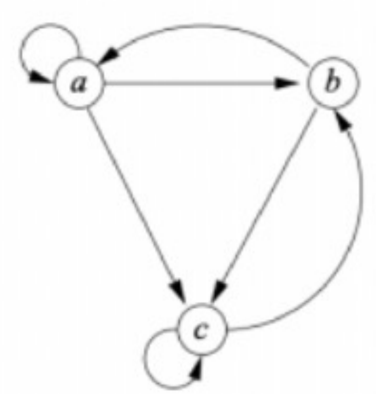
Size of the dense matrix = n^2

Size of the sparse matrix = $2N \log_2(n)$

We save space if: $2N \log_2(n) < n^2$

HW5

4. **(Exercise 5.2.2 MMDS book)** Using the method of Ex 3, represent the transition matrices of the graph from Figure 5.7.



Indexes of positions in matrix M:

$$\begin{bmatrix} (1,1) & (1,2) & (1,3) \\ (2,1) & (2,2) & (2,3) \\ (3,1) & (3,2) & (3,3) \end{bmatrix}$$

We look at non-zero entries in matrix M:
{(1,1), (1,2), (2,1), (2,3), (3,1), (3,2), (3,3)}

$$M = \begin{bmatrix} 1/3 & 1/2 & 0 \\ 1/3 & 0 & 1/2 \\ 1/3 & 1/2 & 1/2 \end{bmatrix}$$

$$\begin{matrix} \parallel & \parallel & \parallel \\ 1 & 1 & 1 \end{matrix}$$

Agenda for today

1. Quick recap

2. Exercises from the book (From Session 5 on moodle)

3. PageRank with NetworkX example