PageRank

MATH 2740 – Mathematics of Data Science – Lecture 26

Julien Arino

julien.arino@umanitoba.ca

Department of Mathematics @ University of Manitoba

Fall 202X

The University of Manitoba campuses are located on original lands of Anishinaabeg, Ininew, Anisininew, Dakota and Dene peoples, and on the National Homeland of the Red River Métis. We respect the Treaties that were made on these territories, we acknowledge the harms and mistakes of the past, and we dedicate ourselves to move forward in partnership with Indigenous communities in a spirit of Reconciliation and collaboration.

Outline

Eigenvector centrality

PageRank

What makes an important webpage?

In days of yore, the web was a small thing

Alta Vista was the search engine of choice

Google started in 1998, based on an algorithm (PageRank) described in a paper of Page, Brin, Motwani and Winograd (link)

Overview

Give each page a rating (of its importance), a recursively defined measure whereby a page becomes important if important pages link to it

Recursive definition: the importance of a page refers back to the importance of other pages that link to it

Random surfer model: a random surfer on the web follows links from page to page. Page rank $\simeq \mathbb{P}$ random surfer lands on a particular page. Popular page $\implies$ higher probability to go there. ($\mathbb{P}$ stands for “probability”)

Example of a Markov chain

Eigenvector centrality

PageRank

FIGS-slides-admin/Gemini_Generated_Image_cdlhu6cdlhu6cdlh.jpeg

Constructing a stochastic matrix from an adjacency matrix

Let A be the adjacency matrix of a simple graph $G = (V, E)$ and D its degree matrix, i.e., the diagonal matrix $D = (d_{ij})$ with diagonal entries

$$d_{ii} = \sum_{j=1}^{n} a_{ji} = \sum_{j=1}^{n} a_{ij}$$

(Recall that A symmetric since G nondirected.) Then the matrix AD^{-1} is column stochastic

Indeed, write entries in D^{-1} as d_{ij}^{-1}. Of course, $d_{ii}^{-1} = 1/d_{ii}$, $1 \leq i \leq n$ and $d_{ij}^{-1} = 0$ if $i \neq j$
Then

$$AD^{-1} = \sum_{k=1}^{n} a_{ik} d_{kj}^{-1} = a_{ij} d_{jj}^{-1}, \quad i, j = 1, \ldots, n$$

So the sum of column j of AD^{-1} is

$$\begin{aligned}
\sum_{i=1}^{n} a_{ij} d_{jj}^{-1} &= d_{jj}^{-1} \sum_{i=1}^{n} a_{ij} \\
&= d_{jj}^{-1} \sum_{i=1}^{n} a_{ji} \\
&= d_{jj}^{-1} d_{jj} \\
&= 1
\end{aligned}$$

and the matrix is column-stochastic

Eigenvector centrality (undirected graph)

Let $\boldsymbol{x}$ be an eigenvector corresponding to the largest eigenvalue λ of the non-negative adjacency matrix A of the undirected graph $G = (V, E)$

(We often call λ the **Perron root** of A and $\boldsymbol{x}$ a **Perron eigenvector**)

The **eigenvector centrality** (or **prestige score**) of vertex i is the ith component of the eigenvector $\boldsymbol{x}$ of the (column) stochastic matrix $N := AD^{-1}$ corresponding to the eigenvalue 1:

$$N\boldsymbol{x} = \boldsymbol{x}$$

Consider a particular vertex i with its neighbouring vertices $\mathcal{N}(i)$:

$$x_i = \sum_{j \in \mathcal{N}(i)} x_j = \sum_j A_{ij} x_j$$

The eigenvector centrality defined this way depends both on the number of neighbours $|\mathcal{N}(i)|$ and the quality of its connections x_j, $j \in \mathcal{N}(i)$

Let $A = (a_{ij})$ be the adjacency matrix of a graph. The eigenvector centrality x_i of vertex i is given by

$$x_i = \frac{1}{\lambda} \sum_k a_{ki} \, x_k$$

where $\lambda \neq 0$ is a constant. In matrix form

$$\boldsymbol{x}^T A = \lambda \boldsymbol{x}^T$$

Hence the centrality vector $\boldsymbol{x}$ is the left eigenvector of the adjacency matrix A associated with the eigenvalue λ

Power method to solve eigenvector centrality

$m(v)$: signed component of maximal magnitude of vector v; if more than one maximal component, let $m(v)$ be the first one. E.g., $m(-3, 3, 2) = -3$

Let $x^{(0)}$ be an arbitrary vector. For $k \geq 1$

- repeatedly compute $x^{(k)} = x^{(k-1)}A$
- normalize $x^{(k)} = x^{(k)}/m(x^{(k)})$

until desired precision is achieved

Then $x^{(k)}$ converges to the dominant eigenvector of A and $m(x^{(k)})$ converges to the dominant eigenvalue of A

If matrix A is sparse, each vector-matrix product can be performed in linear time in the size of the graph.

Power method converges when the dominant (largest) and the sub-dominant (second largest) eigenvalues of A λ_1 and λ_2 are separated, i.e., are different in absolute value, i.e., when $|\lambda_1| > |\lambda_2|$

Rate of convergence is rate at which $(\lambda_2/\lambda_1)^k$ goes to 0. Hence, if the sub-dominant eigenvalue is small compared to the dominant one, the method converges quickly

Why use the leading eigenvector?

We want a nonnegative measure, so we want a vector in $\mathbb{R}_+$

We know from the Perron-Frobenius Theorem that the eigenvector corresponding to the dominant eigenvalue of a nonnegative matrix is nonnegative

Furthermore, if the graph is strongly connected, the matrix is irreducible and the eigenvector corresponding to the dominant eigenvalue is *positive*

Eigenvector centrality

PageRank

FIGS-slides-admin/Gemini_Generated_Image_n8c43hn8c43hn8c4.jpeg

PageRank

Variant of the Eigenvector centrality measure for directed network

Basic PageRank

- Whenever a vertex i has no outgoing link, we add a self-loop to i such that $k_i^{in} = k_i^{out} = 1$. Therefore $A_{ii} = 1$ for such vertices in the adjacency matrix
- Let D^+ be the diagonal matrix of outdegrees where each element $D_{ii}^+ = k_i^{out}$
- Define a column stochastic matrix $N = A(D^+)^{-1}$
- The PageRank centrality of node i is equal to the eigenvector xi of matrix N (The leading eigenvalue is 1): x = Nx

Problem

Given n interlinked webpages, rank them in order of “importance”

Assign the pages importance scores $x_1, x_2, \ldots, x_n \geq 0$

Key insight: use the existing link structure of the web to determine importance. A link to a page “is” a vote for its importance

How does this help with web searches?

First attempt: let x_k equal the number of links to page k

Criticism: a link from an “important” page (like Google) should carry more weight than a link from some random blog!

Second attempt: let x_k equal the sum of the importance scores of all pages linking to page k

Criticism 1: a webpage gets more “votes” (exerts more influence) if it has many outgoing links

Criticism 2: this system only has the trivial solution!

Third attempt (Brin and Page, late 90s): let x_k equal the sum of x_j/n_j, where the sum is taken over all the pages j that link to page k, and n_j is the number of outgoing links on page j

A page's number of votes is then its importance score, and gets split evenly among the pages it links to

Summary: given a web with n pages, construct an $n \times n$ matrix A as

$$a_{ij} = \begin{cases} 1/n_j & \text{if page } j \text{ links to page } i \\ 0 & \text{otherwise} \end{cases}$$

where n_j is number of outgoing links on page j

Sum of jth column is $n_j/n_j = 1$, so A is a stochastic matrix

The ranking vector $\tilde{x}$ solves $A\tilde{x} = \tilde{x}$

Possible issues: existence of solution with nonnegative entries? Non-unique solutions?

PF Theorem guarantees a unique steady-state vector if entries of A are strictly positive or A irreducible. But irreducible $\not\Rightarrow$ λ_1 and λ_2 separated, so make it primitive

Brin-Page: replace A with

$$B = 0.85A + \frac{0.15}{n}\mathbb{J}$$

where $\mathbb{J}$ is the matrix of all ones

$B > 0$ is primitive $\implies$ PF Theorem says B has a unique steady-state vector, x. So x can be used for rankings!

The Random Surfer

Why Markov chains?

Brin and Page

PageRank can be thought of as a model of user behavior. We assume there is a "random surfer" who is given a web page at random and keeps clicking on links, never hitting "back" but eventually gets bored and starts on another random page

Surfer clicks on a link on the current page with probability 0.85; opens up a random page with probability 0.15

A page's rank is the probability the random user will end up on that page, OR, equivalently the fraction of time the random user spends on that page in the long run

In practice

Estimates of the number of web pages vary.. 4.77×10^9 to more than 50×10^9

Computing stationary distribution is **hard** computationally

Instead, use power method, i.e., an iterative method, starting with initial distribution $(1/n, \ldots, 1/n)$