1 PageRank

A principal component (in several meanings) behind Google is its PageRank algorithm, which quantitatively rates the importance of each page on the web, allowing Google to rank the pages and thereby present to the user the more important pages first. Search engines such as Google have to carry out three basic steps:

- Crawl the web and locate all accessible webpages.
- Index the data of the webpages from step 1, so that they can be searched efficiently for relevant key words or phrases.
- Rate the importance of each page in the database, so that when a user does a search and the subset of pages in the database with the desired information has been found, the more important pages can be presented first.

Here, we will focus on the third step. We follow mainly the derivation in [1]. We aim to develop a score of importance for each webpage. A score will be a non-negative number. A key idea in assigning a score to any given webpage is that the page's score is derived from the links made to that page from other webpages. A person is important not if it knows a lot of people, but if a lot of people know that person.

Suppose the web of interest contains n pages, each page indexed by an integer k, $1 \leq k \leq n$. A typical example is illustrated in Figure 1, in which an arrow from page k to page j indicates a link from page k to page j. Such a web is an example of a directed graph. The links to a given page are called the backlinks for that page. We will use x_k to denote the importance score of page k in the web. x_k is nonnegative and $x_j > x_k$ indicates that page j is more important than page k.

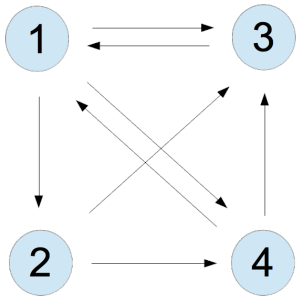

Figure 1: A toy example of the Internet

A very simple approach is to take x_k as the number of backlinks for page k. In the example in Figure 1, we have $x_1 = 2, x_2 = 1, x_3 = 3$, and $x_4 = 2$, so that page 3 is the most important, pages 1 and 4 tie for second, and page 2 is least important. A link to page k becomes a vote for page k's importance. This approach ignores an important feature one would expect a ranking algorithm to have, namely, that a link to page k from an important page should

boost page k's importance score more than a link from an unimportant page. In the web of Figure 1, pages 1 and 4 both have two backlinks: each links to the other, but the second backlink from page 1 is from the seemingly important page 3, while the second backlink for page 4 is from the relatively unimportant page 2. As such, perhaps we should rate the importance of page 1 higher than that of page 4.

As a first attempt at incorporating this idea, let us compute the score of page j as the sum of the scores of all pages linking to page j. For example, consider the web in our toy example. The score of page 1 would be determined by the relation $x_1 = x_3 + x_4$. However, since x_3 and x_4 will depend on x_1, this seems like a circular definition, since it is self-referential (it is exactly this self-referential property that will establish a connection to eigenvector problems!).

We also seek a scheme in which a webpage does not gain extra influence simply by linking to lots of other pages. We can do this by reducing the impact of each link, as more and more outgoing links are added to a webpage. If page j contains n_j links, one of which links to page k, then we will boost page k's score by x_j/n_j, rather than by x_j. In this scheme, each webpage gets a total of one vote, weighted by that web page's score, that is evenly divided up among all of its outgoing links. To quantify this for a web of n pages, let $L_k \subset \{1, 2, \ldots, n\}$ denote the set of pages with a link to page k, that is, L_k is the set of page ks backlinks. For each k we require

$$x_k = \sum_{j \in L_k} \frac{x_j}{n_j},$$

where n_j is the number of outgoing links from page j.

If we apply these scheme to the toy example in Figure 1, then for page 1 we have $x_1 = x_3/1 + x_4/2$, since pages 3 and 4 are backlinks for page 1 and page 3 contains only one link, while page 4 contains two links (splitting its vote in half). Similarly, $x_2 = x_1/3, x_3 = x_1/3 + x_2/2 + x_4/2$, and $x_4 = x_1/3 + x_2/2$. These conditions can be expressed as linear system of equations $Ax = x$, where $x = [x_1, x_2, x_3, x_4]^T$ and

$$A = \begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix}$$

Thus, we end up with an eigenvalue/eigenvector problem: Find the eigenvector x of the matrix A, associated with the eigenvalue 1. We note that A is a column-stochastic matrix, since it is a square matrix for which all of its entries are nonnegative and the entries in each column sum to 1. Stochastic matrices arise in the study of Markov chains and in a variety of modeling problems in economics and operations research. See e.g. [2] for more details on stochastic matrices. The fact that 1 is an eigenvalue of A is not just coincidence in this example, but holds true in general for stochastic matrices.

Theorem 1. *A column-stochastic matrix A has an eigenvalue equal to 1 and 1 is also its largest eigenvalue.*

Proof. Let A be an $n \times n$ column-stochastic matrix. We first note that A and A^T have the same eigenvalues (their eigenvector will usually be different though). Let $e = [1, 1, \ldots, 1]^T$ be the vector of length n which has all ones as entries. Since A is column-stochastic, we have $A^T e = e$ (since all columns of A sum up to 1). Hence e is an eigenvector of A^T (but not of A) with eigenvalue 1. Thus 1 is also an eigenvalue of A.

To show that 1 is the largest eigenvalue of A we apply the Gershgorin circle theorem [2] to A^T. Consider row k of A^T. Let us call the diagonal element $a_{k,k}$ and the radius will be $\sum_{i \neq k} |a_{k,i}| = \sum_{i \neq k} a_{k,i}$ since $a_{k,i} \geq 0$. This is a circle with its center at $a_{k,k} \in [0, 1]$ and with radius $\sum_{i \neq k} a_{k,i} = 1 - a_{k,k}$. Hence, this circle has 1 on its perimeter. This holds for all Gershgorin circles for this matrix. Thus, since all eigenvalues lie in the union of the Gershgorin circles, all eigenvalues λ_i satsify $|\lambda_i| \leq 1$. □

In our example, we obtain as eigenvector x of A associated with eigenvalue 1 the vector $x = [x_1, x_2, x_3, x_4]^T$ with entries $x_1 = \frac{12}{31}, x_2 = \frac{4}{31}, x_3 = \frac{9}{31}$, and $x_4 = \frac{6}{31}$. Hence, perhaps somewhat surprisingly, page 3 is no longer the most important one, but page 1. This can be explained by the fact, that the in principle quite important page 3 (which has three webpages linking to it) has only one outgoing link, which gets all its "voting power", and that link points to page 1.

In reality, A can easily be of size a billion times a billion. Fortunately, we do not need compute all eigenvectors of A, only the eigenvector associated with the eigenvalue 1, which, as we know, is also the largest eigenvalue of A. This in turn means we can resort to standard *power iteration* to compute x fairly efficiently (and we can also make use of the fact that A will be a sparse matrix, i.e., many of its entries will be zero). The actual PageRank algorithms adds some minor modifications, but the essential idea is as described above.

References

[1] Kurt Bryan and Tanya Leise. The $25,000,000,000 eigenvector: The linear algebra behind Google. *Siam Review*, 48(3):569–581, 2006.

[2] R.A. Horn and C.R. Johnson. *Matrix analysis*. Cambridge University Press, Cambridge, 1990. Corrected reprint of the 1985 original.