Math 443/543 Graph Theory Notes 8: Graphs as matrices and PageRank

David Glickenstein

October 30, 2008

1 Representing graphs as matrices

It will sometimes be useful to represent graphs as matrices. This section is taken from C-10.1.

Let G be a graph of order p. We denote the vertices by $v_1, \ldots, v_p$. We can then find an adjacency matrix $A = A(G) = [a_{ij}]$ defined to be the $p \times p$ matrix such that $a_{ij} = 1$ if $v_i v_j \in E(G)$. This matrix will be symmetric for an undirected graph. We can easily consider the generalization to directed graphs and multigraphs.

Note that two isomorphic graphs may have different adjacency matrices. However, they are related by permutation matrices.

Definition 1 *A permutation matrix is a matrix gotten from the identity by permuting the columns (i.e., switching some of the columns).*

Proposition 2 *The graphs G and G' are isomorphic if and only if their adjacency matrices are related by*

$$A = P^T A' P$$

for some permutation matrix P.

Proof (sketch). Given isomorphic graphs, the isomorphism gives a permutation of the vertices, which leads to a permutation matrix. Similarly, the permutation matrix gives an isomorphism. ■

Now we see that the adjacency matrix can be used to count uv-walks.

Theorem 3 *Let A be the adjacency matrix of a graph G, where $V(G) = \{v_1, v_2, \ldots, v_p\}$. Then the (i,j) entry of A^n, where $n \geq 1$, is the number of different $v_i v_j$-walks of length n in G.*

Proof. We induct on n. Certainly this is true for $n = 1$. Now suppose $A^n = \left(a_{ij}^{(n)}\right)$ gives the number of $v_i v_j$-walks of length n. We can consider the entries of $A^{n+1} = A^n A$. We have

$$a_{ij}^{(n+1)} = \sum_{k=1}^{p} a_{ik}^{(n)} a_{kj}.$$

This is the sum of all walks of length n between v_i and v_k followed by a walk from v_k to v_j of length 1. All walks of length $n+1$ are generated in this way, and so the theorem is proven. ■

2 PageRank problem and idea of solution

We will generally follow the paper by Bryan and Leise, denoted BL.

Search engines generally do three things:

1. Locate all webpages on the web.
2. Index the data so that it can be searched efficiently for relevent words.
3. Rate the importance of each page so that the most important pages can be shown to the user first.

We will discuss this third step.

We will assign a nonnegative score to each webpage such that more important pages have higher scores. The first idea is:

- Derive the score for a page by the number of links to that page from other pages (called the "backlinks" for the page).

In this sense, other pages vote for the page. The linking of pages produces a digraph. Denote the vertices by v_k and the score of vertex v_k by x_k.

Approach 1: Let x_k equal the number of backlinks for page v_k. See example in BL Figure 1. We see that $x_1 = 2$, $x_2 = 1$, $x_3 = 3$, and $x_4 = 2$. Here are two problems with this ranking:

Problem 1: Links from more important pages should increase the score more. For instance, the scores of v_1 and v_4 are the same, but v_1 has a link from x_3, which is a more important page, so maybe it should be ranked higher. We will deal with this by, instead of letting x_i equal the total number of links to it, we will have it be equal to the sum of the scores of the pages linking to it, so more important pages count more. Thus we get the relations

$$\begin{aligned} x_1 &= x_3 + x_4 \\ x_2 &= x_1 \\ x_3 &= x_1 + x_2 + x_4 \\ x_4 &= x_1 + x_2. \end{aligned}$$

This doesn't quite work as stated, since to solve this linear system, we see that we get $x_1 = x_2 = \frac{1}{2}x_4 = \frac{1}{4}x_3$, which means that if we look at the first equality, we must have that they are all equal to zero. However, a slightmodification in regard to the next problem will fix this.

Problem 2: One site should not be able to *significantly* affect the rankings by creating lots of links. Of course, creating links should affect the rankings, but by creating thousands of links from one site, one should not be able to boost the importance too much. So instead of giving one vote for each link out, we will give equal votes to each outlink from a particular page, but the total votes is equal to one. This changes the above system to

$$\begin{aligned}
x_1 &= x_3 + \frac{1}{2}x_4 \\
x_2 &= \frac{1}{3}x_1 \\
x_3 &= \frac{1}{3}x_1 + \frac{1}{2}x_2 + \frac{1}{2}x_4 \\
x_4 &= \frac{1}{3}x_1 + \frac{1}{2}x_2.
\end{aligned}$$

This can be solved as follows.

$$\begin{aligned}
x_2 &= \frac{1}{3}x_1 \\
x_4 &= \frac{1}{3}x_1 + \frac{1}{2}x_2 = \frac{1}{2}x_1 \\
x_3 &= \frac{1}{3}x_1 + \frac{1}{2}x_2 + \frac{1}{2}x_4 = \frac{1}{3}x_1 + \frac{1}{6}x_1 + \frac{1}{4}x_1 = \frac{3}{4}x_1.
\end{aligned}$$

Thus we can have a score of $x_1 = 1$, $x_2 = \frac{1}{3}$, $x_3 = \frac{3}{4}$, $x_4 = \frac{1}{2}$. Notice that x_1 has the highest ranking! This is because x_3 threw its whole vote to x_1 and so that even though x_3 got votes from three different sites, they still do not total as much as what x_1 gets. Note, usually we will rescale so that the sum is equal to 1, and so we get

$$x_1 = \frac{12}{31},\ \ x_2 = \frac{4}{31},\ \ x_3 = \frac{9}{31},\ \ x_4 = \frac{6}{31}.$$

3 General formulation

We can state this in a more general way. We want to assign scores so that

$$x_i = \sum_{j \in L_i} \frac{x_j}{n_j}$$

where L_i are the indices such that v_j links to v_i if $j \in L_i$, and n_j is equal to outdegree of v_j. Note that L_i contains $i\deg(v_i)$ elements. The set L_i is called the set of *backlinks* of vertex v_i. This can be rewritten as a vector equation

$$x = Ax,$$

where A is the matrix $A = (a_{ij})$ given by

$$a_{ij} = \begin{cases} \frac{1}{n_j} & \text{if } j \in L_i \\ 0 & \text{otherwise} \end{cases} .$$

This matrix is called the *link matrix.* We note that in the example, the matrix A was

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

The problem of solving for the scores x then amounts to finding an eigenvector with eigenvalue 1 for the matrix A.

We can consider the link matrix as giving the probabilities of traversing a link from the page represented by the column to the page representing the row. Thus it makes sense that the sum of the values of the columns are equal to one.

Definition 4 *A matrix is called a* column stochastic *matrix if all of its entries are positive and the sum of the elements in each column are equal to* 1.

Now the question is whether we can find an eigenvector for a column stochastic matrix, and the answer is yes.

Proposition 5 *If A is a column stochastic matrix, then* 1 *is an eigenvalue.*

Proof. Let e be the column vector of all ones. Since A is column stochastic, we clearly have that

$$e^T A = e^T .$$

Thus

$$A^T e = e$$

and e is an eigenvector with eigenvalue 1 for A^T. However, A and A^T have the same eigenvalues (not eigenvectors, though), so A must have an eigenvalue 1, too. ■

Remark 6 *Do you remember why A and A^T have the same eigenvalues? The eigenvalues of A are the solutions λ of* $\det(A - \lambda I) = \det\left(A^T - \lambda I\right)$.

4 Challenges to the algorithm

There are two issues we will have to deal with.

4.1 Nonuniqueness

We would like our ranking to be unique, which means that we should have only one eigenvector representing the eigenvalue 1. It turns out that this is true if the web is a strongly connected digraph. We will show this later. However, if the web is disconnected, then we can have a higher dimensional eigenspace for eigenvalue 1. Consider the web in BL Figure 2.2. The link matrix is

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

It is easy to see that the vectors $\left[\frac{1}{2}, \frac{1}{2}, 0, 0, 0\right]^T$ and $\left[0, 0, \frac{1}{2}, \frac{1}{2}, 0\right]^T$ have eigenvalue 1. However, we also have that any linear combination of these have eigenvalue 1, and so we have vectors like $\left[\frac{3}{8}, \frac{3}{8}, \frac{1}{8}, \frac{1}{8}, 0\right]^T$ as well as $\left[\frac{1}{8}, \frac{1}{8}, \frac{3}{8}, \frac{3}{8}, 0\right]^T$, which give different rankings!

Proposition 7 *Let W be a web with r components $W_1, W_2, \ldots, W_r$. Then the eigenspace of the eigenvalue 1 is at least r-dimensional.*

Proof (Sketch). A careful consideration shows that if we label the web by assigning the vertices in W_1 first, then the vertices in W_2, etc., then the link matrix will have a block diagonal form like

$$A = \begin{bmatrix} A_1 & 0 & 0 & 0 \\ 0 & A_2 & 0 & 0 \\ & & \ddots & \\ 0 & 0 & & 0 \\ 0 & 0 & 0 & A_r \end{bmatrix},$$

where A_k is the link matrix for the web W_k. If each is column stochastic, each has an eigenvector v_k with eigenvalue 1, and that can be expanded into a eigenvector w_k for A by letting

$$w_1 = \begin{pmatrix} v_1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad w_2 = \begin{pmatrix} 0 \\ v_2 \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

etc. Each of these is linearly independent and part of the eigenspace V_1 of eigenvalue 1. ■

We will figure out a way to deal with this soon.

4.2 Dangling nodes

Definition 8 *A* dangling node *is a vertex in the web with outdegree zero (i.e., with no links).*

The problem with a dangling node is that it produces a column of zeroes. This means that the resulting link matrix is not column-stochastic, since some columns may sum to zero. This means that we may not use our theorem that 1 is an eigenvalue. In fact, it may not be true. We will sketch how to deal with this later.

5 Solving the problems

5.1 Dealing with multiple eigenspaces

Recall that we seemed to be okay if we had a strongly connected graph (web). We will now take our webs that are not strongly connected and make them strongly connected by adding a little bit of an edge between any two vertices. From a probabilistic perspective, we are adding on a possibility of randomly jumping to any page on the entire web. We will make this probability small compared with the probability to navigate from a page.

Let S be the matrix $n \times n$ matrix with all entries $1/n$. Notice that this matrix is column stochastic. In terms of probabilities, this matrix represents equal probabilities of jumping to any page on the web (including the one you are already on). Also notice that if A is a column stochastic matrix, then

$$M = (1 - m) A + mS$$

is column stochastic for all values of m between zero and one. Supposedly the original value for m used by Google was 0.15. We will show that the matrix M has a one-dimensional eigenspace $V_1(M)$ for the eigenvalue 1 as long as $m > 0$.

Note that the matrix M has all positive entries. This motivates:

Definition 9 *A matrix $M = (M_{ij})$ is positive if $M_{ij} > 0$ for every i and j.*

For future use, we define the following.

Definition 10 *Given a matrix M, we write $V_\lambda(M)$ for the eigenspace of eigenvalue λ.*

We will show that a positive column-stochastic matrix has a one dimensional eigenspace V_1 for eigenvalue 1.

Proposition 11 *If M is a positive, column-stochastic matrix, then $V_1(M)$ has dimension* 1.

Proof. Suppose v and w are in $V_1(M)$. Then we know that $sv + tw \in V_1(M)$ for any real numbers s and t. We will now show that (1) any eigenvector in $V_1(M)$ has all positive or all negative components and that (2) if x and y are any two linearly independent vectors, then there is some s and some t such that $sx + ty$ has both positive and negative components. This would imply that $sv + tw$ has all positive or all negative components, and thus v and w must be linearly dependent. ■

Before we prove those propositions, let's define the one-norm of a vector.

Definition 12 *The one-norm of a vector $v = (v_i) \in \mathbb{R}^n$ is equal to*

$$\|v\|_1 = \sum_{i=1}^{n} |v_i|,$$

where $|v_i|$ is the absolute value of the ith component of v.

Proposition 13 *Any eigenvector in $V_1(M)$ has all positive or all negative components.*

Proof. Suppose $Mv = v$. Since M is column-stochastic, we know that

$$\sum_{i=1}^{n} M_{ij} = 1$$

for each j, and since M is positive, we know that

$$|M_{ij}| = M_{ij}$$

for each i and j. Therefore, we see that

$$\begin{aligned}
\|v\|_1 &= \|Mv\|_1 \\
&= \sum_{i=1}^{n} \left| \sum_{j=1}^{n} M_{ij} v_j \right| \\
&\leq \sum_{i=1}^{n} \sum_{j=1}^{n} |M_{ij}| \, |v_j| \\
&= \sum_{j=1}^{n} \sum_{i=1}^{n} M_{ij} \, |v_j| \\
&= \sum_{j=1}^{n} |v_j| = \|v\|_1 .
\end{aligned}$$

That means that the inequality must be an equality, meaning that

$$\left| \sum_{j=1}^{n} M_{ij} v_j \right| = \sum_{j=1}^{n} |M_{ij}| \, |v_j| .$$

This is only true if $M_{ij}v_j \geq 0$ for each i and j (or $M_{ij}v_j \leq 0$ for each i and j). However, since $M_{ij} > 0$, this implies that $v_j \geq 0$ for each j (or $v_j \leq 0$ for each j). Furthermore, since

$$v_i = \sum_{j=1}^{n} M_{ij}v_j$$

with $v_j \geq 0$ ($v_j \leq 0$)and $M_{ij} > 0$, we must have that either all v are zero or all are positive (negative). Since v is an eigenvector, it is not the zero vector. ■

Remark 14 *A similar argument shows that for any positive, column-stochastic matrix, all eigenvalues λ satisfy $|\lambda| \leq 1$.*

Proposition 15 *For any linearly independent vectors x and $y \in \mathbb{R}^n$, there are real values of s and t such that $sx+ty$ has both negative and positive components.*

Proof. Certainly this is true if either x or y have both postive and negative components, so we may assume both have only positive components (the other cases of both negative or one positive and one negative are handled by adjusting the signs of s and t appropriately). We may now consider the vector

$$x = \left(\sum_{i=1}^{n} w_i\right) v - \left(\sum_{i=1}^{n} v_i\right) w.$$

Both the sums in the above expression are nonzero by assumption (in fact, positive). Also x is nonzero since v and w are linearly independent. Notice that

$$\sum_{i=1}^{n} x_i = 0.$$

Since x is not the zero vector, this implies that x must have both positive and negative components. ■

Thus, the matrix M can be used to produce unique rankings if there no dangling nodes.

5.2 Dealing with Dangling nodes

For dangling nodes, we have the following theorem of Perron:

Theorem 16 *If A is a matrix with all positive entries, then A contains a real, positive eigenvalue ρ such that*

1. *For any other eigenvalue λ, we have $|\lambda| < \rho$ (recall that λ could be complex).*

2. *The eigenspace of ρ is one-dimensional and there is a unique eigenvector $x = [x_1, x_2, \ldots, x_p]^T$ with eigenvalue ρ such that $x_i > 0$ for all i and*

$$\sum_{i=1}^{p} x_i = 1.$$

This eigenvector is called the *Perron vector*. Thus, if we had a matrix with all postive entries, as we got in the last section, we can use the Perron vector as the ranking.

6 Computing the ranking

The basic idea is that we can try to compute an eigenvector iteratively like

$$x_{k+1} = Mx_k = M^k x_0.$$

Certainly, if $Mx_0 = x_0$, then this procedure fixes x_0. In general, if we replace this method with

$$x_{k+1} = \frac{Mx_k}{\|Mx_k\|}$$

for any vector norm, we will generally find an eigenvector for the largest eigenvalue.

Proposition 17 *Let M be a positive column-stochastic $n \times n$ matrix and let V denote the subspace of $\mathbb{R}^n$ consisting of vectors v such that*

$$\sum_{i=1}^{n} v_i = 0.$$

Then for any $v \in V$ we have $Mv \in V$ and

$$\|Mv\|_1 \leq c \|v\|_1 ,$$

where $c < 1$.

Corollary 18 *In the situation in the proposition,*

$$\left\|M^k v\right\|_1 \leq c^k \|v\|_1 .$$

Proof. This is a simple induction on k, using the fact that $Mv \in V$ and

$$\left\|M^k v\right\|_1 \leq c \left\|M^{k-1} v\right\|_1 .$$

■

This is essentially showing that the iteration is a contraction mapping, and that will allow us to show that the method works.

Proposition 19 *Every positive column-stochastic matrix M has a unique vector q with positive components such that $Mq = q$ and $\|q\|_1 = 1$. The vector can be computed as*

$$q = \lim_{k \to \infty} M^k x_0$$

for any initial guess x_0 with positive componets such that $\|x_0\|_1 = 1$.

Proof. We already know that M has 1 as an eigenvalue and that the subspace $V_1(M)$ is one-dimensional. All eigenvectors have all positive or all negative components, so we can choose a unique representative q with positive components and norm 1 by rescaling. Now let x_0 be any vector in $\mathbb{R}^n$ with positive components and $\|x_0\| = 1$. We can write

$$x_0 = q + v$$

for some vector v. We note that if we sum the components of x_0 or the components of q, we get one since both have positive components and 1-norm equal to one. Thus $v \in V$ as in the previous proposition. Now we see that

$$\begin{aligned} M^k x_0 &= M^k q + M^k v \\ &= q + M^k v. \end{aligned}$$

Thus

$$\left\|M^k x_0 - q\right\|_1 = \left\|M^k v\right\|_1 \le c^k \|v\|_1 .$$

Since $c < 1$, we get that $\left\|M^k x_0 - q\right\|_1 \to 0$ as $k \to \infty$. ■

We now go back and prove Proposition 17.

Proof of Proposition 17. It is pretty clear that $Mv \in V$ since

$$\begin{aligned} \sum (Mv)_j &= \sum_j \sum_i M_{ji} v_i \\ &= \sum_i \sum_j M_{ji} v_i \\ &= \sum_i v_i = 0 \end{aligned}$$

since M is column-stochastic. Now we consider

$$\begin{aligned} \|Mv\|_1 &= \sum_j \left| \sum_i M_{ji} v_i \right| \\ &= \sum_j e_j \sum_i M_{ji} v_i \\ &= \sum_i a_i v_i \end{aligned}$$

where

$$e_j = sgn\left(\sum_i M_{ji} v_i \right)$$

and

$$a_i = \sum_j e_j M_{ji}.$$

Note that if $|a_i| \leq c$ for all i, then

$$\|Mv\|_1 \leq c \|v\|_1 .$$

We can see that

$$\begin{aligned}
|a_i| &= \left| \sum_j e_j M_{ji} \right| \\
&= \left| \sum_j M_{ji} + \sum_j (e_j - 1) M_{ji} \right| \\
&= \left| 1 + \sum_j (e_j - 1) M_{ji} \right| .
\end{aligned}$$

Each term in the sum is nonpositive, and since M_{ji} are positive and e_j are not all the same sign, the largest this can be is if most e_j are 1 except for a single e_j which is negative and corresponds to the smallest M_{ji}. Thus we see that

$$|a_i| \leq 1 - 2 \min_j M_{ji} \leq 1 - 2 \min_{i,j} M_{ji} < 1.$$

■