Portofolio Assignment 1

Candidate: 25

September 2020

Problem 1

(1a)

Supervised learning consists of machine learning algorithms which both has inputs and outputs. The goal of the supervised learning algorithm is using the observed values of x to make an prediction of y, where we, the creater of the algorithm is the supervisor. So generally the algorithm consists of whats called a mapping of x to y. Or we can generalize this as $y = f(x)$, where f is the mapping of x to y. Examples of supervised learning algorithms include classification, regression etc.

Whereas in unsupervised learning the goal of the algorithm is to find connections in the data, such that we can learn more from it. Here we don't have a supervisor, we simply try to better see patterns in the input data. We also aren't interested in any output since the input is used to train the model to rule out differences of the variables.

The PageRank algorithm is a unsupervised learning algorithm.

(1b)

We are given an equation representing the PageRank method,

$$r(P_i) = \sum_{P_j \in B_{P_i}} \frac{r(P_j)}{|P_j|}. \tag{1}$$

This sum ranks, with the $r(P_i)$ method, the given page, where $p_1, p_2, ..., p_n$ represent all the pages we want to compare against, where n is the number of pages. P_j is a page contained within the set of all the other pages that links to P_i, denoted B_{P_i}. Then the same is true for $r(P_j)$ as is for $r(P_i)$. $|P_j|$ represents the number of links from P_j to other pages.

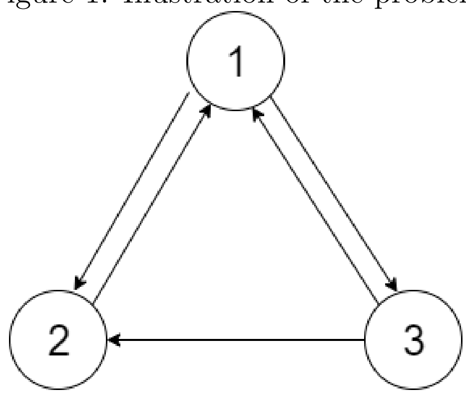
This equation will provide a ranking vector, π, to all the pages $p_1, ..., p_n$. That means we need n of these rankings to compute all the ranks for the different pages.
In a sense we can view this as a way for each page to cast a *vote* for the other pages, but it's altered in a way that we can see in equation 1. What happens here is that to give a rank to

page i, we sum all the ranks of the other pages that link to page i, and divide each of those pages rank by their outlinks. If we now say again that each page can cast a vote for another page, then the rank gained by page i from a particular page j is the vote, divided by the other votes page j has casted to other pages.

(1c)

For this problem I'll show an example i think makes everything a bit clearer.
We are given 3 pages: P_1, P_2, P_3

Figure 1: Illustration of the problem



This system gives the transitionmatrix, $H^T = \begin{bmatrix} 0 & 1 & 1/2 \\ 1/2 & 0 & 1/2 \\ 1/2 & 0 & 0 \end{bmatrix}$.
Where the first column represents P_1's votes for P_2 and P_3. We note that P_1's votes have been divided by the total number of votes P_1 has given. This also goes for the second and third column. This means if we write this as equation 1 we get the system:

$$\begin{aligned} r(P_1) &= \frac{r(P_1)}{|P_1|} + \frac{r(P_2)}{|P_2|} + \frac{r(P_3)}{|P_3|} \\ r(P_2) &= \frac{r(P_1)}{|P_1|} + \frac{r(P_2)}{|P_2|} + \frac{r(P_3)}{|P_3|} \\ r(P_3) &= \frac{r(P_1)}{|P_1|} + \frac{r(P_2)}{|P_2|} + \frac{r(P_3)}{|P_3|} \end{aligned}$$

Which look suspiciously similar to a matrix system of equations. Let's define a vector π, such that,

$$\pi = \begin{bmatrix} r(P_1) \\ r(P_2) \\ r(P_3) \end{bmatrix}, \text{ then when we calculate } H^T\pi$$

We get, $H^T\pi = \begin{bmatrix} r(P_2) + \frac{r(P_3)}{2} \\ \frac{r(P_1)}{2} + \frac{r(P_3)}{2} \\ \frac{r(P_1)}{2} \end{bmatrix} = \begin{bmatrix} r(P_1) \\ r(P_2) \\ r(P_3) \end{bmatrix} = \pi \quad (\Rightarrow \lambda = 1)$

Which is our eigenvector. This also holds for the general case with n pages.
To solve this as an iterative method we use the power method. The power method consists of a random vector, that sums to one, which means it's normalized. Then if you multiply the matrix by itself many times, the transition matrix will converge. The remaining step now is to multiply the random vector with the *power matrix*, to get an eigenvector, hence the name power method. Because of the convergent behavior of the transition matrix we know that we will always get the same eigenvector as an result of the multiplication.

(1d)

If a matrix is stochastic that means that every element of a matrix A, with elements, $[a_{ij}]$ has the property $0 \leq a_{ij} \leq 1$, where each row sums to 1. This can also be seen in the matrix i wrote in the example of the previous task, H^T.