ST758, Homework 4

Due Oct 14, 2014

In this assignment, you are going to try different numerical methods learnt in class on the vanilla Google PageRank problem.

1. Let $\boldsymbol{A} \in \{0,1\}^{n \times n}$ be the connectivity matrix of n web pages with entries

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

$r_i = \sum_j a_{ij}$ is the out-degree of page i. That is r_i is the number of links on page i. Imagine a random surfer exploring the space of n pages according to the following rules.

- From a page i with $r_i > 0$
  - with probability p, (s)he randomly chooses a link on page i (uniformly) and follows that link to the next page
  - with probability $1 - p$, (s)he randomly chooses one page from the set of all n pages (uniformly) and proceeds to that page
- From a page i with $r_i = 0$ (a dangling page), (s)he randomly chooses one page from the set of all n pages (uniformly) and proceeds to that page

The process defines a Markov chain on the space of n pages. Write down the transition matrix $\boldsymbol{P}$ of the Markov chain (in matrix/vector notation).

2. According to standard Markov chain theory, the (random) position of the surfer converges to the stationary distribution $\boldsymbol{x} = (x_1, \ldots, x_n)^T$ of the Markov chain. x_i has the natural interpretation of the proportion of times the surfer visits page i in the long run. Therefore $\boldsymbol{x}$ serves as page ranks; a higher x_i means page i is more visited. It is well-known that $\boldsymbol{x}$ is the left eigenvector corresponding to the top eigenvalue 1 of the transition matrix $\boldsymbol{P}$. That is $\boldsymbol{P}^T\boldsymbol{x} = \boldsymbol{x}$. Therefore $\boldsymbol{x}$ can be solved as an eigen-problem. Show that it can also be cast as solving a linear system. Remember $\boldsymbol{x}$ is a distribution so we normalize it to have $\sum_{i=1}^n x_i = 1$.

3. Download the `stat-ncsu.zip` package from course webpage. Unzip the package, which contains two files `U.txt` and `A.txt`. `U.txt` lists the 500 URL names. `A.txt` is the 500×500 connectivity matrix. Read data into R. Compute summary statistics:

- number of pages
- number of edges
- number of dangling nodes
- max in-degree
- max out-degree
- visualize sparsity pattern of $\boldsymbol{A}$

4. Set the *teleportation* parameter at $p = 0.85$. Try the following methods to solve for $\boldsymbol{x}$ using the `stat-ncsu` data.

   (a) A dense linear system solver such as LU decomposition

   (b) A simple iterative linear system solver such as Jacobi or Gauss-Seidel

   (c) (Optional) A sophiscated iterative linear system solver such as biconjugate gradients stabilized method or generalized minimum residual method (GMRES)

   (d) A dense eigen-solver

   (e) A simple iterative eigen-solver such as the power method

   (f) (Optional) A sophisticated iterative eigen-solver such as the Arnoldi and Lanczos algorithms

5. List the top 20 ranked URLs you found.

6. As of Thursday Sep 25 2014, there are at least 1.21 billion indexed webpages on internet according to `http://www.worldwidewebsize.com/`. Comment on whether each of these methods may or may not work for the PageRank problem at this scale.