LAB3: Markov Chains, web searches, and PageRank

Additional materials and references are associated with Cleve Moler's `http://www.mathworks.com/moler/exm/chapters/pagerank.pdf`.

You also should have Handout on Markov Chains available from Canvas class website or form class website.

Students in MTH 420 follow and answer Q1-Q10 for the network A. Next, change p to be 0.9, and later to p = 0.5. (Produce PageRank and answer Q9 and Q10 for each case). Repeat PageRank for network C when p=0.5 **Extra:** work with network B.

Below network A = network from class (Textbook Fig. 2.1, p88). Also, network B = network from Fig. 7.5 in Moler's paper, which is similar to A (but not identical). We will also call network C = the network from Fig. 7.1 in Moler's paper.

INSTEAD: *Omegas are encouraged to solve Q1-Q10 for network A. Next, identify some real (but small) web network, for example, Math Department website, and encode it in a graph and find the page rank of its variants. You can make a lot of assumptions here, and disregard some of the links and so on. State clearly and describe what you're doing.*

Hint: many of the answers to Q1-Q10 are clear from the Handout on Markov Chains (read it carefully). Moler's paper gives a nice illustration of the whole PageRank idea.

Recall from class that the network A can be encoded in the matrix $G \in R^{n\times n}, n = 4$. Recall that $G_{ij} = 1$ if there is an edge **to node** i **from node** j**.**

$$G = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix} \tag{1}$$

Q1: Is G a stochastic matrix according to the handout ?

Now imagine playing a game in which one has to move from a node to a node of the network at each step, but only along one of the edges that are allowed.

Start by sitting at node 1, and having to move. You have three edges to choose from (so your destination is node 2, 3, or 4). We can assume each can be chosen with probability 1/3. Similarly, if you sit at node 2, you can move towards node 3, or 4 only, so the probability is 1/2.

These probabilities are actually equal $1/c_j$ where $c_j = \sum_i G_{ij}$ are the column sums.

Now let us record our "state" when moving, in a vector $x \in \mathbb{R}^n$. The "state" is *a stochastic vector* (see handout) which records the probability that we are at a certain node.

Start with $x_0 = [1, 0, 0, 0]^T$ that is, at time t_0 we are (certainly) sitting in node 1. Next we move. Where to ? Well, according to the rules established above, we can

only go to 2, 3, or 4, so the probability that we are in either one of them is 1/3. So $x_1 = [0, 1/3, 1/3, 1/3]^T$. Notice we could have gotten there by setting $x_1 = Ax_0$, where A is like G except it has 1/3 in the first column wherever G has a 1.

Q2 What if we started at node 2, that is with $x_0 = [0, 1, 0, 0]^T$? Compute x_1. Now you would like to have 1/2 in the second column of G so that $x_1 = Gx_0$.

Now we combine the ideas from above and want to have

$$A_0 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1/3 & 0 & 0 & 0 \\ 1/3 & 1/2 & 0 & 0 \\ 1/3 & 1/2 & 1 & 0 \end{bmatrix} \tag{2}$$

so that $x_{k+1} = A_0 x_k$.

Q3 No matter what you start with, if $x_0 = e_1$, what happens after $k = 17$ steps ? What if $x_0 = e_2$?

Q4 Is A_0 a stochastic (transition) matrix ? If not, why ? (this explains Q3).

So we need to fix A_0 and in fact, we need to fix the entire process of transitions and "moving" so it will resemble actual web-surfing. It seems reasonable to assume that once you land in a node from where there are no out-links (like node 4), you just leave that page to go to any other page (of the four available). In fact, when you sit at any node at any time, maybe you choose *some other* link than those listed on that page. Say, sitting at 2, you can click on links to 3 or 4, but perhaps *also* you can go to 1 or you can choose to remain at 2. [Remember that you can always type the URL of any pages yourself.]

So let us implement these ideas and set up a new "correct" matrix A. Let the probability that when being in a node (on a page), you actually follow one of the links from that page, be p (typical value is $p = 0.85$ or something like that). So the probability that you choose some other page than these is 1-p, and now we see that the probability that you choose one of the possible n pages is $\sigma = \frac{1-p}{n}$. So we set up A as follows

$$A_{ij} = pG_{ij}/c_j + \sigma \tag{3}$$

That makes sense of course if $c_j \neq 0$. What to do if $c_j = 0$?

Q5 Recall to yourself in human language what it means if $c_j = 0$.

If $c_j = 0$, we set up $A_{ij} = 1/n$ for all i and this particular j, because it seems reasonable that a user would go to any random url possible with equal probability.

Q6 Write out by hand what A is like.

You can automate the process in MATLAB by typing, for a given G, the following

```
c = sum(G,1)
k = find(c~=0)
D = sparse(k,k,1./c(k),n,n)
e = ones(n,1)
z = ((1-p)*(c~=0) + (c==0))/n
A = p*G*D+e*z
```

Or you can do it all by hand ...

Q7 Is A a stochastic matrix ? Is it a regular stochastic matrix ?

Q8 Perform experiments starting with different x_0 of your choice. They can describe the initial state when you start from a specific node. They can also be any stochastic vectors.
For each such experiment, perform the operation `x=A*x` several times which calculates new elements of the sequence $x_{k+1} = Ax_k$.
What do you think is the limit of this process? Does the limit (call it x_*) depend on what you started with ?

Q9 The limit x_* discussed in Q8 is the steady-state vector described in the handout. Interpret the equation it has to satisfy using eigenvalues and eigenvectors.

Q10 Verify in MATLAB that your interpretation from Q9 is the same as the limit you found in Q8. (Remember this vector has to be a *stochastic vector*.

SUMMARY: the vector x_* you found in Q8 and Q10 is the Google Page Rank of the network we discussed. Read Moler's paper for more information.

Extra: experiment with the code `pagerank` code which is part of the EXM collection provided by Cleve Moler at
`http://www.mathworks.com/moler/exm/chapters.html`
Report on what you have learned and on anything else related to this project.