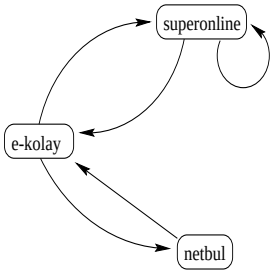
CMPE 360: Numerical Methods
Homework 2 (due May 30th)

Problem 1 (40 pts)

The aim of this problem is to illustrate how linear algebra can be made use of in internet search engines. When several pages match a search query, the pages should be displayed in order of their *importance*. So how is this page importance determined? Consider the patent pending `PageRank` scheme used by `Google` to determine the importance of each web page: A page is important if important pages link to it. This is a recursive definition of importance which needs to be solved. If we imagine that each page has one unit importance initially, then we can iterate a process of each page sharing whatever importance it has among its successors and receiving new importance from its predecessors. This can be represented as a matrix vector product as illustrated in the following example: Consider the following link structure:



Let s, n and e denote the importance of Superonline, Netbul and E-kolay respectively. Then, in the following iteration, a column means that the corresponding site is giving $1/m$th of its importance (with m being the outdegree of the site) to the sites it points to. The iteration for this example is given as:

$$\begin{bmatrix} s^{(t+1)} \\ n^{(t+1)} \\ e^{(t+1)} \end{bmatrix} = \begin{bmatrix} 1/2 & 0 & 1/2 \\ 0 & 0 & 1/2 \\ 1/2 & 1 & 0 \end{bmatrix} \begin{bmatrix} s^{(t)} \\ n^{(t)} \\ e^{(t)} \end{bmatrix}$$

where $s^{(0)} = n^{(0)} = e^{(0)} = 1$.

Eventually, the above iteration will reach a limit which happens to be *its component* in the principal eigenvector of this *example* matrix. In the limit, the solution will be $s = 6/5$, $e = 6/5$ and $n = 3/5$, i.e. Superonline and E-kolay will have the same importance. Netbul, on the other hand, will have half the importance.

a) Answer the following questions:

(*i*) Implement the above procedure by using sparse matrix representation in MATLAB. Compare the result returned with that of MATLAB function which returns the principal eigenvector.

(*ii*) What will happen if the link from Netbul to E-kolay is removed by Netbul ?

(*iii*) If after the action in (b), Netbul wants to become the most important site, what should it do ?

(*iv*) Make up an example involving 16 sites such that exactly 4 sites will have zero importance in the limit, BUT at iteration $t = 1$, none of the sites should have zero importance.

(*v*) Give an example link graph whose link matrix will lead to an iteration which will not converge, but rather cycle through some values.

b) In general, if we have a graph with n nodes, then we have an $n \times n$ link matrix P and a rank vector r. The above iteration is then given as:

$$r^{(t+1)} = Pr^{(t)} \tag{1}$$

where $r^{(0)} = [1 \dots 1]^T$. However, because of the problems exemplified by (*ii*), (*iii*), (*iv*) and (*v*), the iteration (1) for finding the page ranks is NOT used in practice. Instead, a dampening factor α ($0 < \alpha < 1$) is introduced and the following iteration is employed in practice:

$$r^{(t+1)} = \alpha Pr^{(t)} + (1 - \alpha)c \tag{2}$$

where $c = [1 \dots 1]^T$.

Suppose you are told that the iteration in (1) will converge if P is *primitive*. A square non-negative matrix T is called primitive if there exists a positive integer k such that $(T^k)_{i,j} > 0$ for all i and j. Answer the following questions:

(*i*) Carry out the iteration (2) for the example in the figure.

(*ii*) Prove that iteration (2) will converge. (Hint: use the result that (1) will converge if P is primitive).

Problem 2 (40 pts)

In this project, you will implement a natural boundary cubic spline drawing program using `MATLAB`. Your program should have the following features:

- The user should be able to enter a sequence of points graphically (i.e. you display a square drawing area with a grid and the user enters the coordinates graphically by clicking on the drawing area). You can use `MATLAB` `ginput` function to input coordinates of the clicked point.
- You should fit natural cubic splines for the parametric curves $x(t)$ and $y(t)$.
- You should solve for coefficients of the cubic splines and then draw the whole curve afterwards.

Problem 3 (20 pts)

The aim of this project is to illustrate how `MATLAB` can be used to quickly develop an image compression program based on singular value decomposition. Read the online tutorial on singular value decompositon at the address :

`http://www.davidson.edu/academic/math/will/svd/index.html`

Take an image of *yourself* and perform singular value decomposition using `MATLAB`. Display the images you get for rank 10, 20, 30, 40, 100, 200 and 280 approximations of the original image. The following `MATLAB` functions may be useful to you:

- `svd`
- `gray`
- `image`
- `colormap`
- `reshape`