CS485 Spring 2007
Homework 11

Due Date: April 13 2007

NOTE: To speed up homework grading, please submit each homework problem on a separate sheet of paper, with you name and NetID on the top. Thank you!

1. Consider a d-regular graph (every vertex has degree exactly d). Prove that $\lambda_n = -d$ if and only if the graph is bipartite (λ_n is the smallest eigenvalue of the adjacency matrix).

2. This problem should convince you empirically that standard deviation of sum of n independent identically distributed random variables scales as $\sqrt{n}$. Pick a random variable X (any distribution is fine, as long as it has a finite variance). And generate random variables $S_n = \sum_1^n X$ (every X is an independent draw). Plot standard deviation of S_n (you might have to do many draws from S_n to get a reasonable estimate of the standard deviation, depending on the distribution of X) for various values of n, and see how it relates to standard deviation of X. Comment on what you see.

3. Plot a distribution of eigenvalues of adjacency matrixes of random graphs. That is, generate a random graph $G(n, \frac{1}{2})$ and get its adjacency matrix for some n (or, equivalently, generate a random symmetric $0,1$ matrix), find its eigenvalues (using e.g. Matlab) and plot a histogram of the eigenvalues for many such random matrixes. Don't forget to comment on what you see.

4. One very interesting problem concerning graphs is how to plot them. A lot can be learned about a graph if one can get a good "feeling" of how it looks like. This is actually a research area of its own, but there are some simple ways to do it that work amazingly well, based on spectral analysis of a special matrix associated with the graph. It's almost like "magic' :-)

   Let $L_G = (l_{ij})$ be an $n \times n$ matrix associated with graph $G = (V, E)$ defined as:

   $$l_{ij} = \begin{cases} -1 & \text{if } (i,j) \in E \\ \deg(i) & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

   This matrix is called the *Laplacian matrix* of G (note that it's very similar to the adjacency matrix of G). To plot the graph G in 2D, one only needs two numbers per vertex (as the x and y coordinates), so two vectors of size n. And now the magic: try two eigenvectors associated with the two smallest nonzero eigenvalues of L_G. Here is a little Matlab code to get you started:

```
% plots graphs using eigenvectors of Laplacian
Ks = 1:30;        % try Ks=1:60
A = full(bucky);
A = A(Ks,Ks); % get adjacency matrix of a part of a 'ball' graph
L = -A+diag(sum(A));  % get Laplacian
[Evecs,Evals] = eig(L);   % find its eigenvalues/vectors
gplot(A,Evecs(:,2:3),'-*')  % plot it!
```

Your task is to do the same for some graphs you have seen during the class. Don't forget to include a random graph, graph grown using preferential attachment, and a (small) real-world graph (e.g from the first homework). Make plots of at least one graph from each category. You can also extend it to 3D by considering the eigenvectors associated with the smallest three nonzero eigenvalues.