MATH 513 Spring 2009, Homework 10, Addendum

It's usually hard to compute eigenvalues of a matrix – primarily because it involves finding roots of polynomials, which is hard. However, there is one easy way to compute the largest and smallest real eigenvalues called the "Power method".

Here's what it does for a square matrix A and a given tolerance $\epsilon > 0$:

1. Choose a random unit vector x_0. Make sure its size is compatible with A. Set $k = 1$.
2. Repeat the following until $\|x_k - x_{k-1}\| \leq \epsilon$:
   (a) Let $y_k = Ax_{k-1}$
   (b) Let $x_k = y_k/\|y_k\|$
   (c) Increment k
3. The eigenvalue is $\|y_k\|$ and the eigenvector is x_k.

Problem 1 Take the above verbal description and turn it into a Matlab function. Take a few random 5x5 matrices and try it out with a tolerance of 10^{-5}. Compare the results of your function with the results of Matlab's "eig" command. (Hint: they had better be close!)

Problem 2 How can you modify the above algorithm to find the smallest eigenvalue? Write another Matlab function to do this and test it. (Hint: you'll want to avoid taking inverses as usual, though you'd really want to, theoretically.)