Homework 3: Eigenvalues and SVD

Due: Feb 24, 2014

1. Prove similar matrices have the same eigenvalues. Matrices A and B are similar if there exists a matrix P such that $B = P^{-1}AP$. Will similar matrices have the same eigenvectors?

2. Let A be a symmetric matrix. Prove that if x and y are eigenvectors corresponding to distinct eigenvalues then x and y are orthogonal.

3. Given $A \in \mathbf{R}^{m\times n}$, prove $||A||_2 = \sigma_1$ the largest singular value of A. Recall $||A||_2 = \max_{\|x\|=1} \|Ax\|_2$.

4. Consider the matrices

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 0 & -4 & 2 \\ 0 & 0 & 6 \end{pmatrix} \text{ and } B = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 3 & 2 \\ 0 & 0 & -6 \end{pmatrix}$$

   Which matrix do you expect convergence with the Power Method to be quickest? Is there something that you can do to the other matrix to speed the convergence?

5. MATLAB: Compare the timing (using `tic` and `toc`) of the Power Method (starting with a random unit vector) and the QR Factorization for calculating the largest eigenvalue of

$$A = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 2 \end{pmatrix}$$

   for $n = 100$. For both algorithms, continue the iteration between the guess for the largest eigenvalue and the true largest eigenvalue (use `eig`) is less than `1e-6`. When would you want to use the QR and when would you want to use the Power Method?

6. MATLAB: For the programming exercise of this problem set, you will be looking at compression of pictures. First, let's experiment with a picture stored in MATLAB. To begin, let's open the image called `rice.png`.

```
close all
A=imread('rice.png');
A = A(:,:,1);
A=im2double(A);
imshow(A)
```

   (a) Now experiment with MATLAB's built in SVD operation to calculate a low rank approximation. Print a 1-rank, 20-rank, 50-rank, and 100-rank approximation and comment on the savings for using such a low rank approximation.

(b) Load your favorite picture into MATLAB and experiment with different low rank approximations. Calculate which low rank approximation gives the best approximation while saving the most space. In other words, which rank gives you the most bang for the buck! Be sure to print your picture and your low rank approximation and give an explanation on why you chose the low rank approximation.