**Computational Methods** Homework #3

CMSC / AMSC 460, Fall 2018 Due 11:59, October 24, 2018

**Problem 1.** (15)

(i) Suppose that $\lambda_1, \lambda_2, \cdots, \lambda_n$ are the eigenvalues of $A$ with associated eigenvectors $x_1, x_2, \cdots, x_n$ and that $\lambda_1$ has multiplicity 1. If $y$ is any vector with the property that $y^T x_1 = 1$, prove that, for a fixed constant $\sigma$, the matrix

$$B = A - \sigma x_1 y^T$$

has eigenvalues $\lambda_1 - \sigma, \lambda_2, \cdots, \lambda_n$.

(ii) Consider now the following two matrices:

$$(a)\ A = \begin{bmatrix} 1 & 2 & -1 \\ 1 & 0 & 1 \\ 4 & -4 & 5 \end{bmatrix},$$

$$(b)\ A = [a_{ij}] = \begin{cases} i+j, & i = j, \\ ij, & i \neq j, \end{cases}$$

where $i, j = 1, 2, \ldots, 10$.

Write a MATLAB program based on the power method to compute the dominant eigenpair $(\lambda_1, x_1)$ of these matrices. Then, set $\sigma = \lambda_1$ in (i) and compute all the eigenpairs of the corresponding matrix $B$ using QR method, as given by the routine `qreig` on page 241 of Ascher and Greif's book. Moreover, use the MATLAB command `eigs` to obtain the eigenpairs and compare them with ones computed with your MATLAB routines. You should also tabulate the outputs of your routine after each iteration.

**Problem 2.** (15)

(a) Given a real matrix $A$ and vector $x$, show that the Rayleigh quotient

$$\lambda(x) = \frac{x^T A x}{x^T x}$$

is the least-squares solution of the problem

$$\min_{\lambda} \|Ax - \lambda x\|,$$

where the norm is the Euclidian norm $\|v\| = \sqrt{v^T v}$. What can you say if $x$ is an eigenvector of $A$?

(b) Write a MATLAB program to compute the dominant eigenpair of the matrices in Problem 1 (ii) above by applying the inverse iteration method to $A - \sigma I$. Here, you should set $\sigma_{k-1} = x_k^T A x_k$ at the $k$th iteration of your MATLAB routine, for an initial vector $x_0$.

**Problem 3.** (15) This problem will explore properties of the singular value decomposition (SVD) of a matrix $A \in \mathbb{R}^{m\times n}$, given by $A = U\Sigma V^T$, where $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ are orthogonal matrices (i.e., $U^T U = I_m$, $V^T V = I_n$) and $\Sigma$ is a diagonal matrix consisting of zeros and singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0, \quad r = \min\{m, n\}$.

(a) Show that for any $n$-vector $u$, $\|Uu\| = \|u\|$ where the norm is the Euclidean norm $\|u\| = \sqrt{u^T u}$.

(b) Show that the singular values are the square roots of eigenvalues of $A^T A$.

(c) The norm of $A$ is defined to be $\max_{v\neq 0} \frac{\|Av\|}{\|v\|}$. Show that $\|A\| = \sigma_1$.
Hint: Use the results of parts (a) and (b) above together with the result of Problem 2.

(d) Let $A_r = \sum_{i=1}^{r} \sigma_i u_i v_i^T$, where $u_i$ and $v_i$ are the $i$th column vectors of $U$ and $V$, respectively. Show that $||A - A_r|| = \sigma_{r+1}$.

**Problem 4.** (30) The SVD can be used in image compression to compress an image without losing much of the image quality. In this problem, you are required to choose only one of the three images provided and then perform an SVD on the image. In this exercise, you will use two MATLAB functions, `imread` and `imagsc`; the first of these reads data from an image and represents it in numerical form, and the second converts numerical data to an image.

(i) Open a new script in MATLAB and and type the following:

```
% Load your chosen image
A = imread('photo.gif'); % Here, photo=cat or city or house
A = double(A);
n = size(A,1);
% Display Image
figure,imagesc(A); colormap gray; axis image;
title('original')
```

(ii) Run this section of the code with your chosen image and save the figure in a pdf file as an image that maintains the resolution ($512 \times 512$). In your report, make sure to introduce what the image is and state that you will be looking at an introductory method of image compression.

(iii) In your MATLAB script, perform SVD decomposition on your image by typing the following:

```
[U,S,V] = svd(A);
```

Save the diagonal values of $S$ as the variable $sv$, stored as a vector.

(iv) Create a new section of your script and perform a series of low-rank approximations $A_r$ using reduced SVD for various values of $r$. To do this, type the following:

```
r = 25;
Ar = U(:,1:r)*S(1:r,1:r)*V(:,1:r)';
```

Then run the code to see what pixel matrix Ar that you got.

```
% Add the title name.
figure,imagesc(Ar); colormap gray; axis image; title('')
```

(v) Run this code for $r = 1,\ 10,\ 100,\ 200$. Using the MATLAB function `subplot`, create a figure for each $r$ beside the original in a figure. Save these figures. (You should have four figures each with the original image and the associated $r$-image.) On the title of the $r$-images, add the corresponding $r$ value. Add these four images to your pdf file.

(vi) In your own words, explain and comment on the image qualities in (v). It may be useful to look at the singular values over these $r$ ranges. What matters most in preserving the image quality?

(vii) Create a new section in your script and plot the diagonal values of $S$ from (iv), which should be stored as $sv$. This plot should be scaled in such a way that you can make some judgements about the data shown. The title of this plot should be "Singular values of the image". The x-label should be "Index" and the y-label should be "Singular value". Save this plot to your pdf file.

(viii) From the plot in (vii) above, use your judgment to determine an appropriate value for the index $r$. Run your code from (iv) with this value of $r$. Save the image to your pdf document and give your reason for choosing such an $r$.