Lab 2, February 12th, 2016
Numerical Linear Algebra, Spring 2016

1. Find the SVD of the following matrices using the Matlab function *svd*:

$$\begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix}, \qquad \begin{pmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{pmatrix}, \qquad \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 0 \end{pmatrix}, \qquad \begin{pmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{pmatrix}.$$

2. Choose one of the matrices in (1) and find its SVD by hand.

3. "Matrix times circle equals ellipse". Geometrical illustration in Matlab.

```
% Generate a circle of points with 1 degree separation':
circ = zeros(2, 360); %a 2 x 360 zero array
deg = [ 1 : 1 : 360 ];
circ(1, :) = cos(deg*pi / 180); %pi is a built-in constant
circ(2, :) = sin(deg*pi / 180);

% Define A to be your nonsingular 2 x 2 matrix
% (do not choose the identity matrix, it is not interesting.)
A = [ 0.5 -2; -1 0.3]; % for instance ...

% Now compute the SVD of A:
[U, S, V] = svd(A);

% plot A times circle as follows:
hold on % concatenate all subsequent plots on the same graph
axis equal
plot(circ(1,:), circ(2,:), 'o');
```

```
Acirc = A*circ;
plot(Acirc(1,:), Acirc(2,:), 'o');
% plot original basis vectors
origBasis = [ V(:,1) [ 0 0 ]' V(:,2) ];
plot(origBasis(1,:), origBasis(2,:), '--');

% The major and minor axes of the ellipse are given by
% the left singular vectors of A (the columns of U) or,
% equivalently, by the eigenvectors of AA'.
% Add them to your plot as follows:
U_line = [ U(:,1)*S(1,1) [ 0 0 ]' U(:,2)*S(2,2) ];
plot(U_line(1,:), U_line(2,:));

hold off
```

4. Low rank approximations (image compression using SVD): approximate a given matrix A (and the corresponding image) by a matrix A_k of rank $k < \text{rank}(A)$. A_k is obtained by the SVD of A where the smallest singular values starting from the $(k+1)$-th one are replaced with zeros. Consider the following image compression example (you can choose your favorite image and vary k).

```
X = imread('ngc6543a.jpg'); % returns a 650-by-600-by-3 array, A.
image(X); % display the image;
A = double(rgb2gray(X)); % transform to real values
A = A(:,:,1); % for gray-scale images,
%we consider only the first matrix in the array
figure; imagesc(A); colormap gray; % original b&w image
set(gca,'XTick',[],'YTick',[]);
[m,n] = size(A);
```

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

% Low rank approximations
k = 100; % choose k
Ak = U(:,1:k)*S(1:k,1:k)*V(:,1:k)';
% display compressed (low rank) image
figure; imagesc(Ak); colormap gray;
set(gca,'XTick',[],'YTick',[]);

figure; imagesc(A–Ak); colormap gray; % residual image
set(gca,'XTick',[],'YTick',[]);
```