[MATH2605] Homework 8

Due: Mar. 9

Problem 1, Singular Value Decomposition

Consider a matrix $\mathbf{A} \in \mathbb{R}^{n\times m}$. Recall that the singular value decomposition of $\mathbf{A}$ is

$$\mathbf{A} = \mathbf{UDV}^T \quad , \qquad \text{where} \quad \mathbf{D} \in \mathbb{R}^{n\times m}, \; \mathbf{U} \in \mathbb{R}^{n\times n}, \text{ and } \; \mathbf{V} \in \mathbb{R}^{m\times m}. \tag{1}$$

Remember that $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices and the matrix $\mathbf{D}$ is

$$\begin{aligned}
\mathbf{D} &= \begin{bmatrix} \sigma_1 & 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \sigma_n & 0 & \dots & 0 \end{bmatrix} \quad \text{if } m > n \\
\mathbf{D} &= \begin{bmatrix} \sigma_1 & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \sigma_m \\ 0 & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix} \quad \text{if } m < n
\end{aligned} \tag{2}$$

The values $\sigma_1, \sigma_2, \dots$ are called singular values. Remember that singular values are non-negative numbers.
Let $\mathbf{U}$ and $\mathbf{V}$ be

$$\mathbf{U} = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3 & \mathbf{u}_4 & \dots & \mathbf{u}_n \end{bmatrix} \; , \quad \text{and} \quad \mathbf{V} = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 & \mathbf{v}_4 & \dots & \mathbf{v}_m \end{bmatrix}. \tag{3}$$

Note that $\mathbf{u}_i \in \mathbb{R}^{n\times n}$ and $\mathbf{v}_i \in \mathbb{R}^{m\times m}$.
Now, assume that $n < m$. Using (1),(2), and (3), show that

$$\mathbf{A} = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T + \dots \sigma_n \mathbf{u}_n \mathbf{v}_n^T \tag{4}$$

Assume that singular values are ordered $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \dots \geq \sigma_n$. Define $\mathbf{A}_1$, $\mathbf{A}_2$, ... as

$$\begin{aligned}
\mathbf{A}_1 &= \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T \\
\mathbf{A}_2 &= \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T \\
\mathbf{A}_k &= \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T + \dots + \sigma_k \mathbf{u}_k \mathbf{v}_k^T
\end{aligned} \tag{5}$$

Obviously, if $k = n$, $\mathbf{A}_k = \mathbf{A}_n = \mathbf{A}$. On the other hand, if $k < n$, $\mathbf{A}_k$ is not $\mathbf{A}$. However, since singular values are ordered, $\sigma_{k+1}, \sigma_{k+2}, \dots \sigma_n$ will be small, and therefore, $\mathbf{A}_k$ may be close to $\mathbf{A}$, i.e., $\mathbf{A}_k \approx \mathbf{A}$. We can test this idea using an image.

Consider a grey-scale image that has $n \times m$ pixels. We store brightness of pixel in the matrix $\mathbf{A} \in \mathbb{R}^{n\times m}$. The element $\mathbf{A}(i, j)$ represent the brightness of pixel at i$^{\text{th}}$column and j$^{\text{th}}$row. For simplicity, assume that $\mathbf{A}(i, j) = 0$ when the pixel is completely dark, and $\mathbf{A}(i, j) = 1$ when the pixel is fully bright.
Download an image from the web site:

```
http://www-static.cc.gatech.edu/~bmkim/MATH2605/hr.jpg
```

Copy the image file to the current working directory of Matlab. Load the image and store it in **A** matrix by

```
A = double(imread('hr.jpg','jpg'))/255;
```

You can draw the image by

```
imshow(A);
```

Compute the singular value decomposition by

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

Compute and show $\mathbf{A}_{200}$ by

```
A200 = U(:,1:200)*D(1:200,1:200)*V(:,1:200)';
imshow(A200);
```

Compute and show $\mathbf{A}_{150}$, $\mathbf{A}_{100}$, $\mathbf{A}_{50}$, $\mathbf{A}_{30}$, $\mathbf{A}_{30}$, and $\mathbf{A}_{10}$. Submit the printed copies of these images and discuss their qualities.
Try with another image.

Problem 2, Nonlinear ODE, Local Stability

Consider the following ODE.

$$\begin{aligned} \dot{x}_1 &= 2x_1 - x_1 x_2 \\ \dot{x}_2 &= 2x_1^2 - x_2 \end{aligned} \tag{6}$$

Compute all equilibrium points. Compute the Jacobian at each point. Compute eigenvalues of each Jacobian, and discuss their stability. The vector field and trajectory can be drawn by using Matlab. First create a m file, whose file name must be *p2_fun.m*

```
function [xdot] = p2_fun(t,x,p1,p2)
xdot(1,1) =  2*x(1) - x(1)*x(2);
xdot(2,1) =  2*x(1)*x(1) - x(2);
```

Then, execute the following.

```
clf;
figure(1), hold on
[x1,x2] = meshgrid(-2:0.1:2,  -0.5:0.1:3);
dx1 =  2*x1 - x1.*x2;
dx2 =  2*x1.*x1 - x2;
quiver(x1,x2,dx1,dx2);
ax = axis;
axis(ax);

N = 50;
options = odeset('RelTol',1e-4);
for i=1:N
  xic=ginput(1)
  [t,x]=ode45('p2_fun',[0 20],xic,options);
  plot(x(:,1),x(:,2),'r')
end
hold off;
```

You can click on the graph to start the trajectory. Plot the vector field and trajectories, and submit a printed copy or a hand-drawn sketch.