QUESTION – 1

We know singular values of A are the square root of the eigenvalues of AA^T and $A^T A$.

$$A = U\Sigma V^T$$

$$A^T = (U\Sigma V^T)^T = V\Sigma^T U^T$$

We know $\Sigma^T = \Sigma$ since it is a diagonal matrix. We also know $U^T = U^{-1}$ and $V^T = V^{-1}$ since they are orthogonal matrices.

$$AA^T = X\Lambda X^{-1}$$

$$(U\Sigma V^T)(V\Sigma^T U^T) = U\Sigma(V^T V)\Sigma U^T = U\Sigma I\Sigma U^T = U\Sigma^2 U^T = X\Lambda X^{-1}$$

Thus equation

$$\Sigma^2 = \Lambda$$

can be obtained which concludes that SVD decomposition gives square root of eigenvalues of AA^T which are the singular values of A.

$$A^T A = Y\Lambda Y^{-1}$$

$$A^T A = (V\Sigma U^T)(U\Sigma V^T) = V\Sigma(U^T U)\Sigma V^T = V\Sigma I\Sigma V^T = V\Sigma^2 V^T = Y\Lambda Y^{-1}$$

Thus equation

$$\Sigma^2 = \Lambda$$

can be obtained which again concludes that SVD decomposition gives square root of eigenvalues of $A^T A$ which are the singular values of A.

QUESTION – 2

a-)

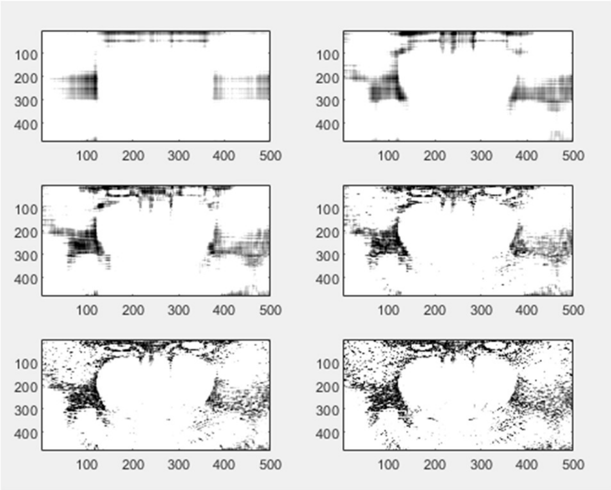

Figure 1: Mandrill image compressed with truncated SVD, with r values ranging from 2^1 to 2^6

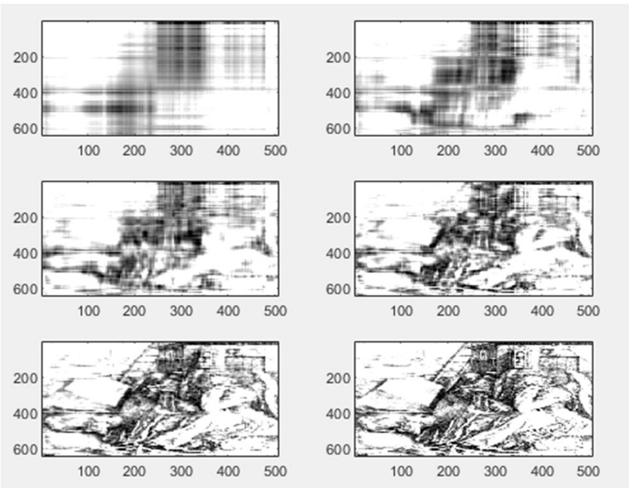

Figure 1: Drawing of Albrecht Dürer compressed with truncated SVD, with r values ranging from 2^1 to 2^6

```
close all
figure()
colormap(gray);
load mandrill;
[U,S,V] = svd(X);
for i = 1:6
    r = 2.^i;
    subplot(3,2,i),
image(U(:,1:r)*S(1:r,1:r)*V(:,1:r)');
end

figure()
colormap(gray);
load durer;
[U,S,V] = svd(X);
colormap(gray)

for i = 1:6
    r = 2.^i;
    subplot(3,2,i),
image(U(:,1:r)*S(1:r,1:r)*V(:,1:r)');
end
```

b-) For both images until r = 32 image looks way too blurry to understand anything from it. For higher r values drawing of Dürer looks better and more detailed than the Mandrill image since original image contains more details. For lower r values Mandrill looks better since it does not have many details, thus required singular values to make a reasonable image is much less than the singular values required by Dürer's drawing.

Original Mandrill image is a 480x500 image which takes 1 byte per pixel thus takes $480 \times 500 \; = \; 225{,}000$ bytes to store. After compression the storage that is required can be calculated by

$$f(r) = r \times (500 + 480 + 1)$$

Original Durer image is a 648x509 image thus takes $648 \times 509 \; = 329{,}832$ bytes to store. Required storage can be calculated by

$$g(r) = r \times (648 + 509 + 1)$$

QUESTION – 3

```
function vk = power_method(inputMatrix, initialGuess, iterationCount)
    vk = initialGuess;
    for i = 1:iterationCount
        tmp = inputMatrix * vk;
        vk = tmp / norm(tmp);
        disp(-vk);
    end
end
A = [-2 1 4; 1 1 1; 4 1 -2];
v0 = [1; 2; -1];
vt = [1; 2; 1];
power_method(A,v0,20);
power_method(A,vt,5);
```

Iteration	1	2	3	4	5
$V0 = (1,2,\text{-}1)^T$	0.4364 -0.2182 -0.8729	-0.8083 -0.1155 0.5774	0.6448 -0.0586 -0.7621	-0.7356 -0.0294 0.6768	0.6922 -0.0147 -0.7216
$V0 = (1,2,1)^T$	-0.5774 -0.5774 -0.5774	-0.5774 -0.5774 -0.5774	-0.5774 -0.5774 -0.5774	-0.5774 -0.5774 -0.5774	-0.5774 -0.5774 -0.5774

For $\mathrm{V0}(1,2,1)^T$ power method found the eigenvector on the first iteration. $V0 = (1,2,-1)^{\mathrm{T}}$ seems like converging but it requires more steps.

[V, D] = eig(A)

V = 0.7071 0.4082 -0.5774

0 -0.8165 -0.5774

-0.7071 0.4082 -0.5774

D = -6.0000 0 0

0 0.0000 0

0 0 3.0000