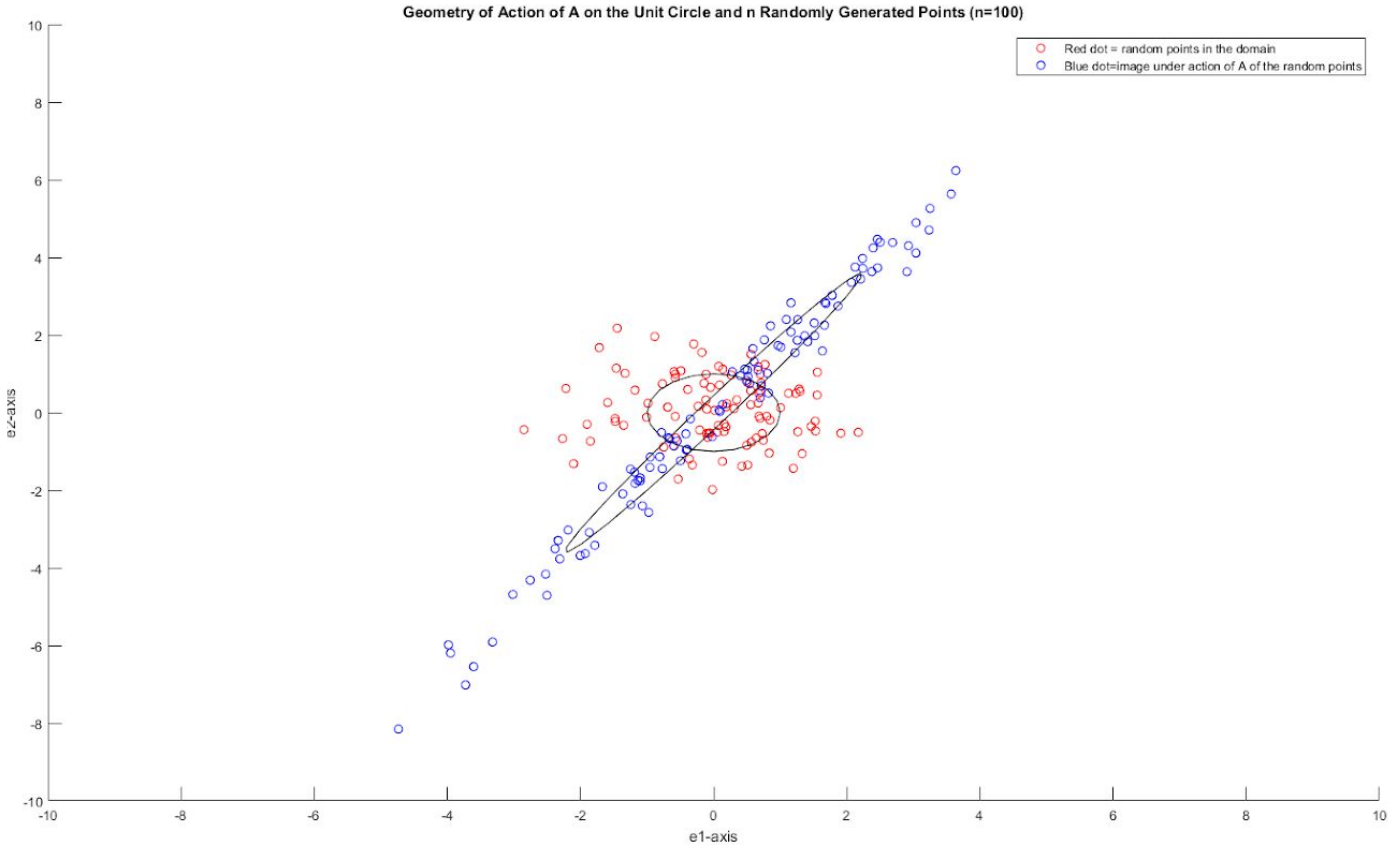
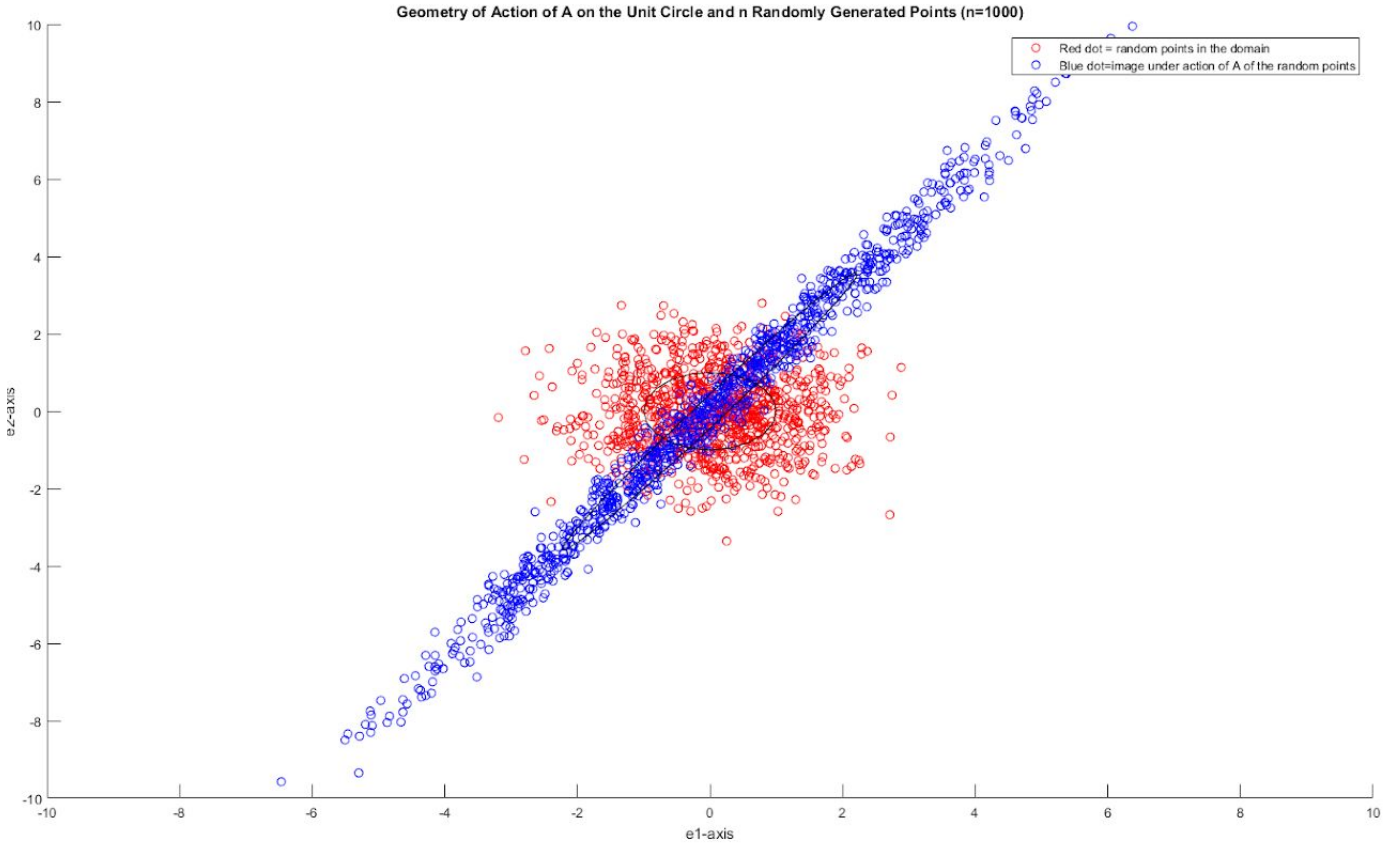
Q1)b)

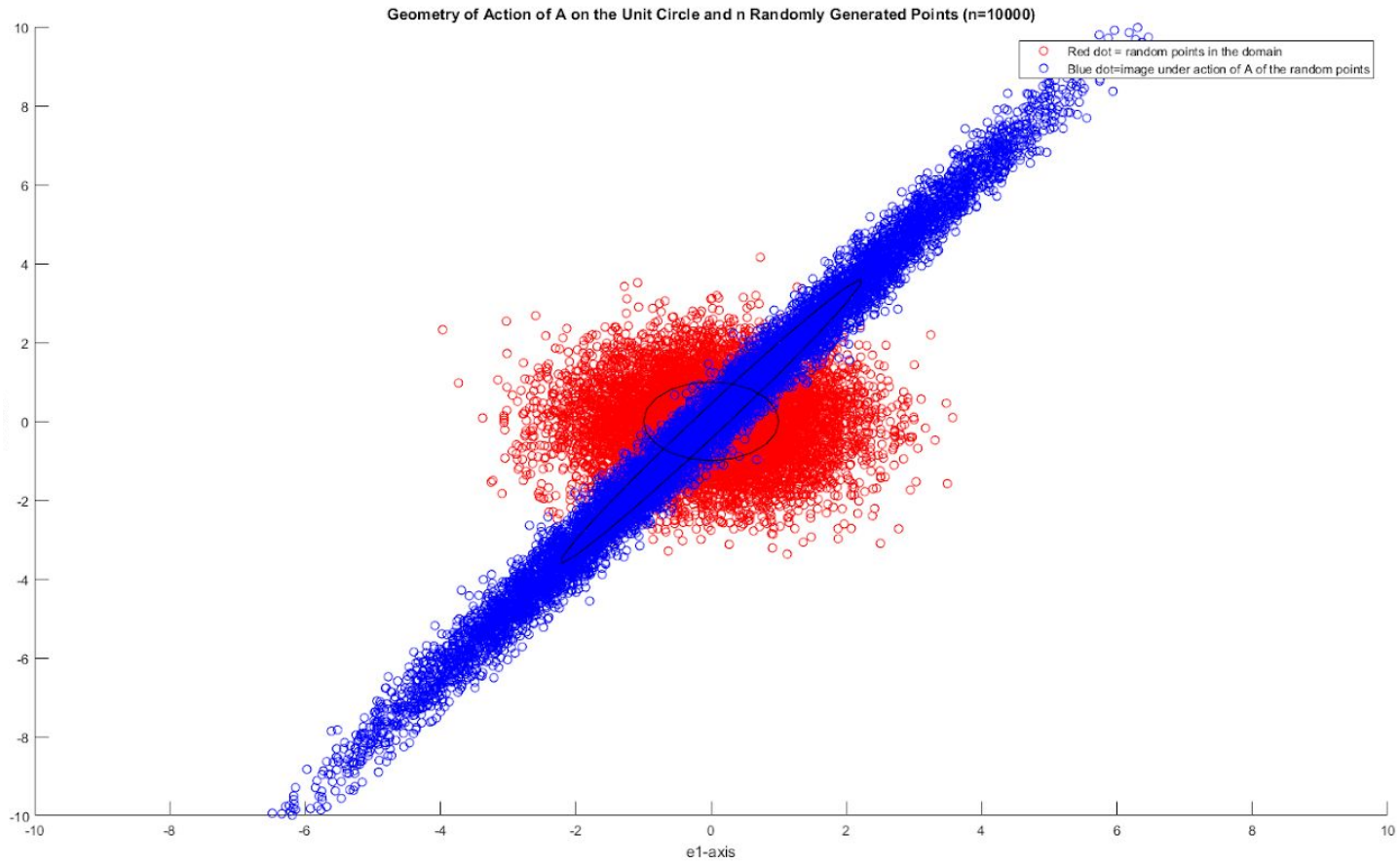
n=100



n=1,000



n=10,000



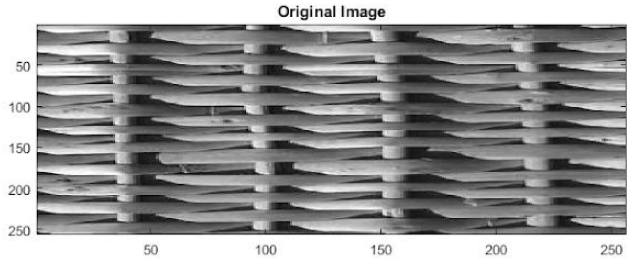
Q1 Code

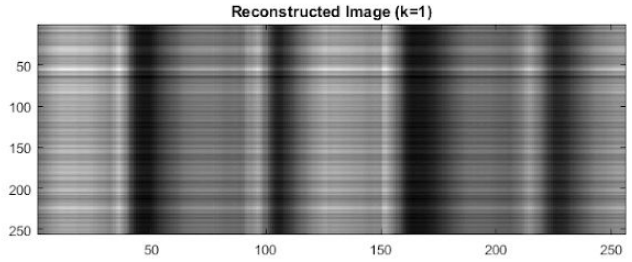
Editor - D:\Users\jpearc6\Documents\MATLAB\q1.m

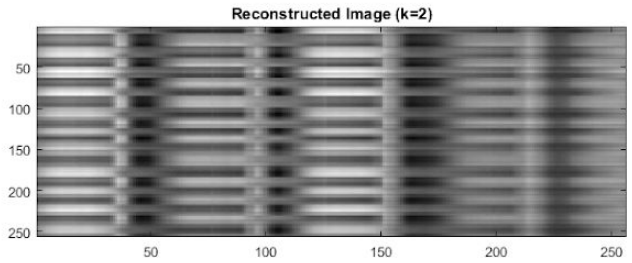
q1.m

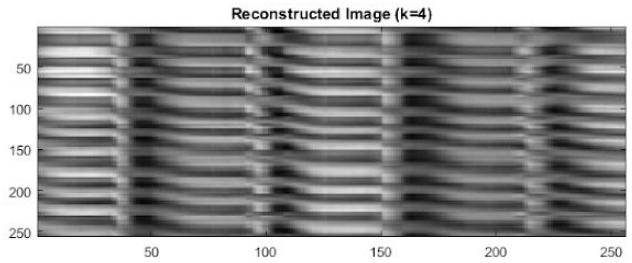
```
A=[1,2;2,3]
N=20
n=10000

%create points on unit circle
angle = 2*pi/N;

u = zeros(2,N);

for i = 0:N
    %parition unit cirle
    u(1,i+1) = cos(angle*i);
    u(2,i+1) = sin(angle*i);
end

%ellipse points
e = A*u;
%random unit cirle points
points = randn(2,n);
%random ellipse points
transPoints = A*points;

figure();
%scatter(u(1,:),u(2,:))
scatter(points(1,:),points(2,:),'r')
hold on
scatter(transPoints(1,:),transPoints(2,:),'b')
plot(u(1,:),u(2,:),'k')
plot(e(1,:),e(2,:),'k')
xlabel('e1-axis');
ylabel('e2-axis');
title('Geometry of Action of A on the Unit Circle and n Randomly Generated Points (n=10000)');
legend('Red dot = random points in the domain','Blue dot=image under action of A of the random points');
xlim([-10 10]);
ylim([-10 10]);
hold off;
```

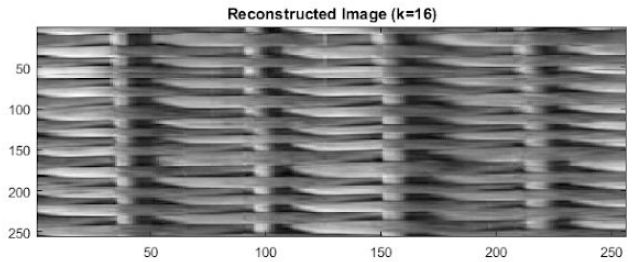
Q2)a)

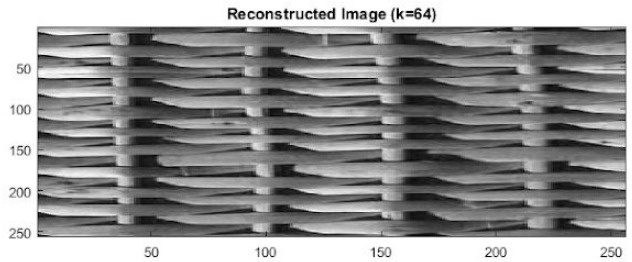










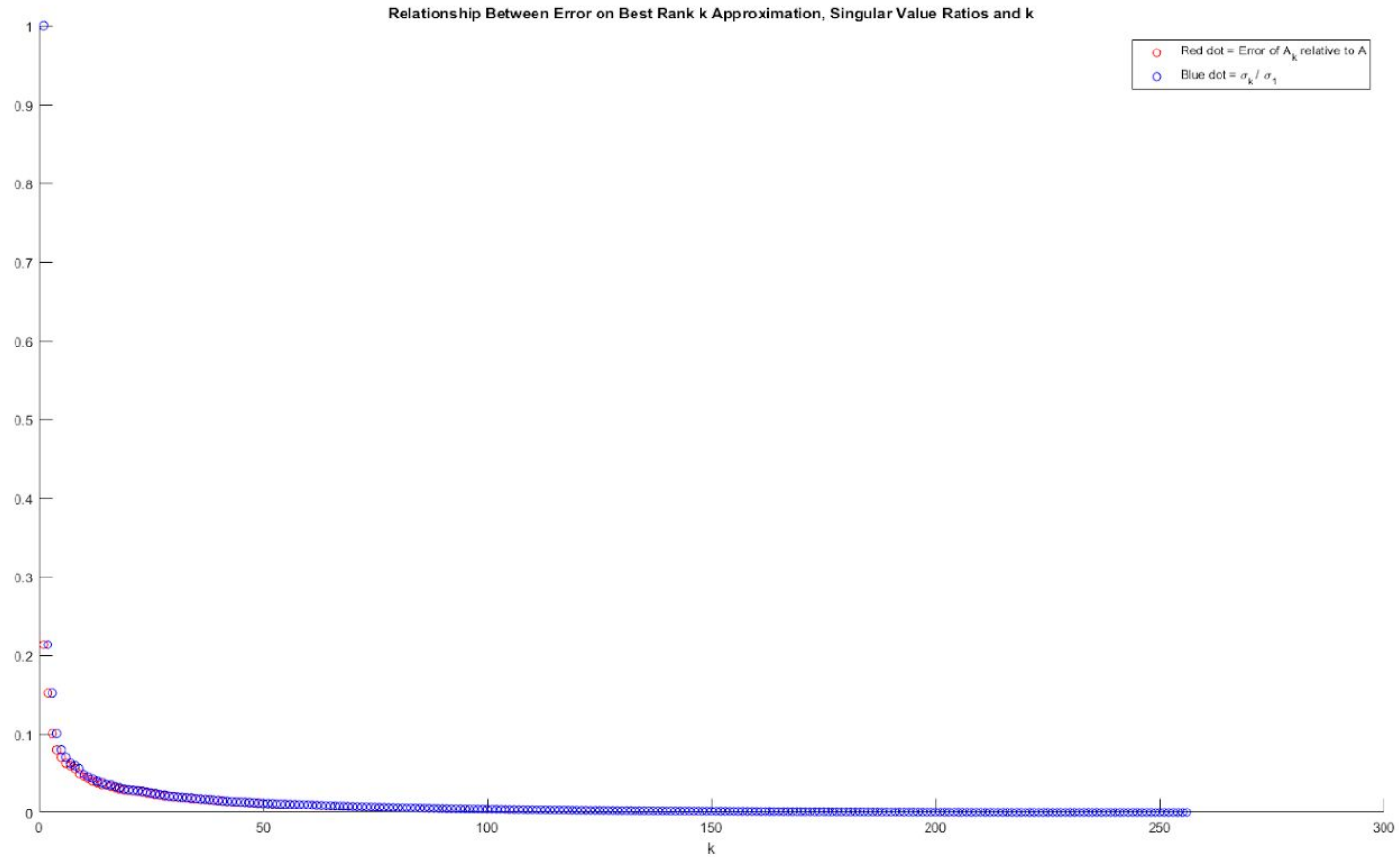


Observations:

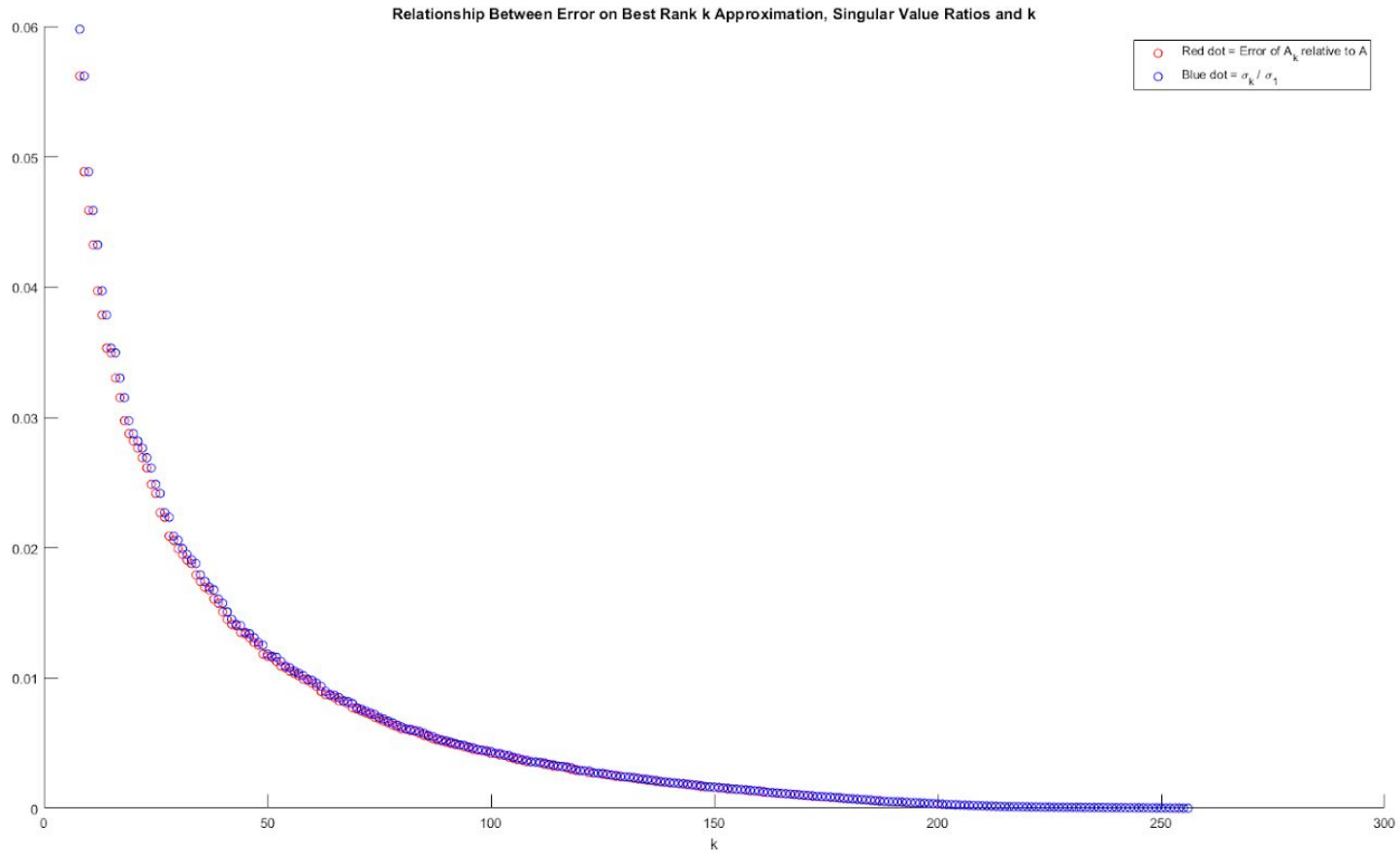
- Image reconstruction quality improves as k increases
- k=1 and k=2 image contours are far from correct
- k=4 and k=16 image contours are better but image is still quite blury
- k=64 is difficult to distinguish from the original image without careful inspection

Q2)b)

i=1,...,256



i=8,...,256



A key observation from the graphs:

$$E_k = \frac{\sigma_{k+1}}{\sigma_1}$$

Which makes sense since,

$$||A - A_k||_2 = \sigma_{k+1}$$

And,

$$||A||_2 = \sigma_1$$

Therefore,

$$E_k = \frac{||A-A_k||_2}{||A||_2} = \frac{\sigma_{k+1}}{\sigma_1} \leq 0.05$$
$$\Rightarrow \ \sigma_{k+1} \leq 0.05\sigma_1$$

In our example this achieved with k=9 (can be observed from second graph)
Q2 Code:

Editor - D:\Users\jpearc6\Documents\MATLAB\q2.m

q1.m | q2.m

```
load('A')
colormap(gray)
[m,n] = size(A);
imageSizes = [1,2,4,16,64]

[U, S, V] = svd(A);
%subplot(3,2,1);
%imagesc(A)
%title('Original Image')

for i=1:5
    k = imageSizes(i);
    U_k = U(:,1:k);
    S_k = S(1:k,1:k);
    V_k = V(:,1:k);
    %k rank approximation
    A_new = U_k*S_k*V_k';
    %subplot(3,2,i+1);
    %imagesc(A_new);
    %title(sprintf('Reconstructed Image (k=%d)', int8(imageSizes(i))));
end

i = zeros(1,256);
E_k = zeros(1,256);
singularRatio = zeros(1,256);

for k=1:n
    U_k = U(:,1:k);
    S_k = S(1:k,1:k);
    V_k = V(:,1:k);

    A_new = U_k*S_k*V_k';
    i(k) = k;
    %error
    E_k(k) = norm(A-A_new)/norm(A);
    %Singular value ratio
    singularRatio(k) = S(k:k,k:k)/S(1:1,1:1);
end

start = 8;
scatter(i(1,start:n),E_k(1,start:n),'r')
hold on
scatter(i(1,start:n),singularRatio(1,start:n),'b')
xlabel('k');
ylabel('');
title('Relationship Between Error on Best Rank k Approximation, Singular Value Ratios and k');
legend('Red dot = Error of A_k relative to A','Blue dot = \sigma_k / \sigma_1');
hold off;
```