MAT 343 Lab 5 - Jordan Ledbetter

```
A=imread('gauss.jpg'); %load the picture
B=double(A(:,:,1));  %convert to double precision
B=B/255;             %scale the values of B
[U S V]=svd(B);      %compute the SVD decomposition of B
```

Problem 1

Compute the dimensions of U, S and V

```
size(U)
```

```
ans = 1×2
   636   636
```

```
size(S)
```

```
ans = 1×2
   636   482
```

```
size(V)
```

```
ans = 1×2
   482   482
```

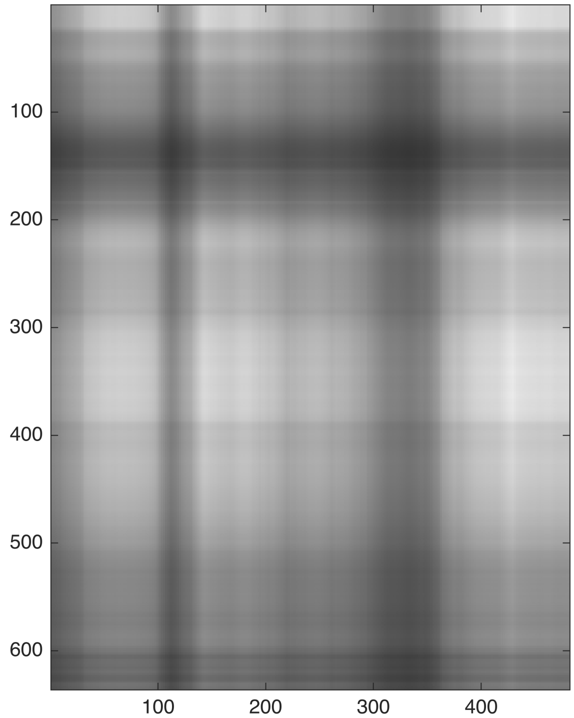
Problem 2

Compute the best rank-1 approximation and store it in `rank1`

```
[U,S,V] = svd(B);
k = 1;
rank1 = U(:,1:k)*S(1:k,1:k)*V(:,1:k)';
```

Visualize rank1 by performing steps 3 -6

```
C = zeros(size(A));

C(:,:,1) = rank1;
C(:,:,2) = rank1;
C(:,:,3) = rank1;

C = max(0, min(1,C));

image(C), axis image
```



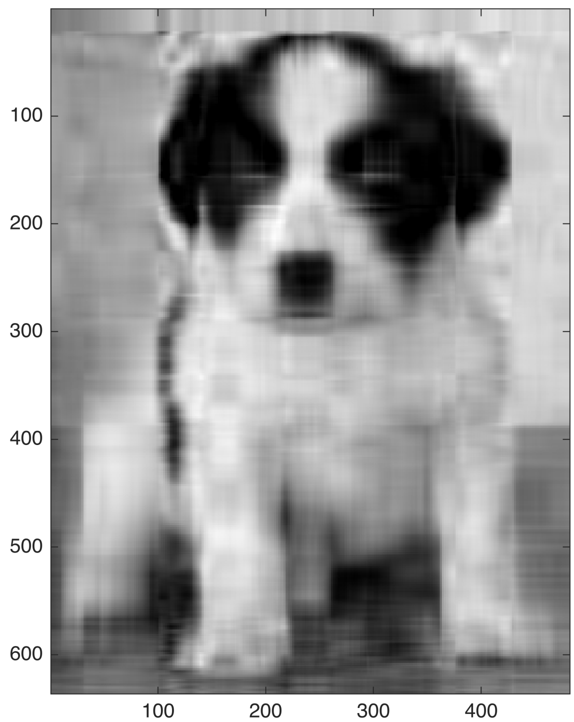
Problem 3

Create and view a rank-10 approximation to the original picture

```
k = 10;
rank1 = U(:,1:k)*S(1:k,1:k)*V(:,1:k)';

C = zeros(size(A));
C(:,:,1) = rank1;
C(:,:,2) = rank1;
C(:,:,3) = rank1;

C = max(0, min(1,C))
```

```
C =
C(:,:,1) =

    0.4741    0.4754    0.4767    0.4785    0.4828    0.4873    0.4912    0.4935    0.4972    0.5019    0.
     ⋮
```

```
image(C), axis image
```



Problem 4

Experiment with different ranks until you found one that gives, in your opinion, an acceptable approximation.

```
k = 40;
rank1 = U(:,1:k)*S(1:k,1:k)*V(:,1:k)';

C = zeros(size(A));
C(:,:,1) = rank1;
C(:,:,2) = rank1;
C(:,:,3) = rank1;

C = max(0, min(1,C))
```

```
C =
C(:,:,1) =

    0.5430    0.5410    0.5380    0.5369    0.5421    0.5449    0.5459    0.5449    0.5482    0.5484    0.
    ⋮
```

```
image(C), axis image
```

Problem 5

What rank-r approximation exactly reproduces the original picture? Explain,

Answer: I believe that a rank-r approximation of 100 reproduces the original picture. After increasing the rank-r approximation by 10 each run, it seems that the image doesn't appear any clearer after a rank-r approximation of 100.

Problem 6

(i)

How much data is needed to represent a rank-k approximation? Explain.

Answer: The total amount of data needed to represent the rank-k approximation is given by k(m+n+1), where m represents rows and k is columns for matrix U, k represents rows and n is columns for matrix V, and matrix S is a diagonal matrix with k entries. With this information, we create the formula kxm + k + kxn, which simplifies to k(m+n+1). In this expression, 1 accounts for the diagonal entries stored separately from the off-diagonal entries.

(ii)

Find the compression rate for the value of the rank you determined in problem 4. Explain.

Answer: The compression rate is calculated by kx(m+n+1)/(mxnxd), which represents the amount of data required for the rank-k approximation divided by the amount of data required for the original photo. Therefore, the compression rate would be 40x(636+482+1)/(636x482x8) = 0.01825 x 100 = 1.8%.

What does the compression rate represent? Explain.

Answer: The compression rate refers to the ratio of the amount of data needed to represent the original matrix to the amount of data needed to represent the rank-k approximation. The compression rate measures the degree of data reduction achieved by the rank-k approximation, indicating how much the size of the data can be reduced while preserving most of the essential information contained in the original data.

Problem 7

Find the smallest value of k such that the rank-k approximation uses the same or more amount of data as the original picture. Explain how you obtained the answer.

Answer: To find the smallest value of k such that rank-k approximation uses the same or more amount of data as the original photo by k = ceil(3mn / m+n+1). By subsituting the variables with k = ceil(3*636*482 / 636+482+1), our k value will be 824.