Image Compression using SVD

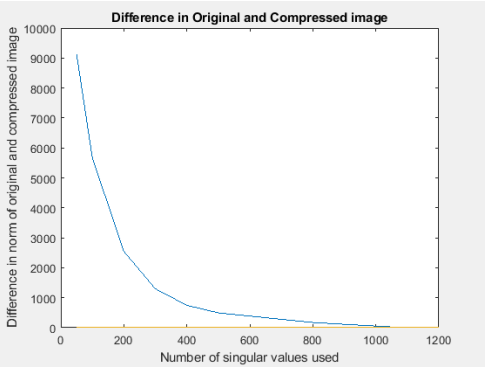
Sunday, November 17, 2019 9:58 PM

Approach :

1. Our motto is to compress the image such that the image takes less space for storage and at the same time removes extra noise and does not compromise on clarity.
2. In order to compare different number of singular values for each image, I have found the difference in 2-norm of original and compressed image.
3. Ten Linearly spaced values between 10 and rank of original matrix are used for comparison.
4. Compression ratio and picture quality is used to find the best low rank approximation of the image

Futurama.png

The rank of the given matrix a little over 1200. On using ten different rank matrices, we see that a matrix having number of singular values between 400 and 600 gives a good approximation of the image.



Singular values matrix

```
p_val =

  Columns 1 through 6

         10         155         300         445         590         735

  Columns 7 through 10

        880        1025        1170        1315
```

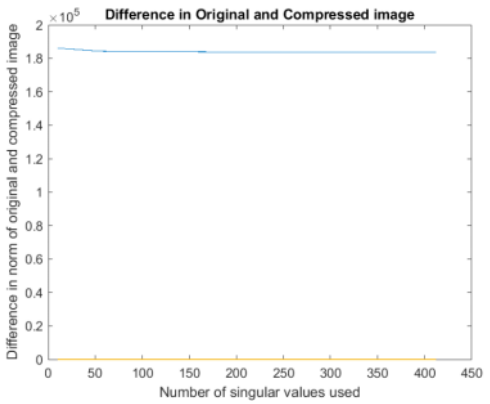
Compression Ratios:

```
compression_ratio =

  Columns 1 through 7

    0.0122    0.1897    0.3671    0.5446    0.7220    0.8995    1.0769

  Columns 8 through 10

    1.2544    1.4318    1.6093
```

We infer from the subplot below that rank = 600 picture gives a good approximation of the image without compromising on the p icture quality

UB.png

the rank of the original image is 242. On experimenting with various ranks from 10 to 242, we see that the difference of norm s of actual and compressed image is least between 50 and 100. **Best rank that retains the picture quality is 100.**



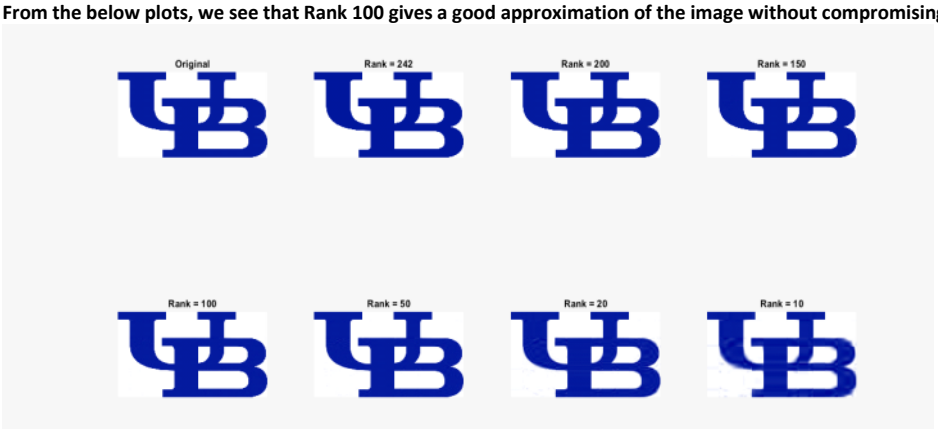
Singular values used :

```
p_val =

  Columns 1 through 7

   10.0000   54.6667   99.3333  144.0000  188.6667  233.3333  278.0000

  Columns 8 through 10

  322.6667  367.3333  412.0000
```

Compression ratios for each singular value:

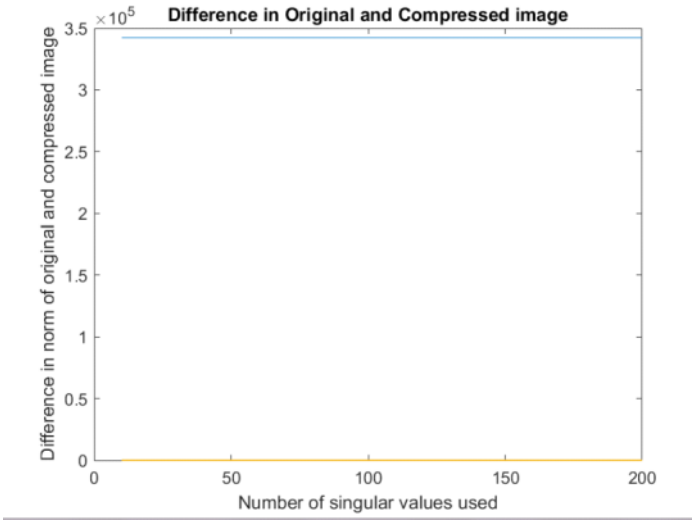
```
compression_ratio =

  Columns 1 through 7

    0.0368    0.2024    0.3680    0.5300    0.6956    0.8612    1.0231

  Columns 8 through 10

    1.1887    1.3543    1.5163
```

From the below plots, we see that Rank 100 gives a good approximation of the image without compromising on the picture qualit y.



Square.png

The rank of the given matrix is 200. From the plot, we can see that the image gives best results with the minimum number of s ingular values also.
Best rank that retains the picture quality :



The compression ratios starting from 10 to 200 in linearly spaced ten different number of singular values :
We see that the lowest number of singular value,10 gives the best compression ratio. The inference is that a number smaller t han 10 may be sufficient as well.

Singular values :

```
  Columns 1 through 7

   10.0000   31.1111   52.2222   73.3333   94.4444  115.5556  136.6667

  Columns 8 through 10

  157.7778  178.8889  200.0000
```

Compression Ratio :

```
compression_ratio =

  Columns 1 through 7

    0.1003    0.3208    0.5313    0.7419    0.9524    1.1629    1.3734

  Columns 8 through 10

    1.5839    1.7945    2.0050
```

As we see from below subplots, Rank 2 matrix does not compromise the quality of the matrix and gives a good approximation of the image. Hence we can say that **the best rank which retains the picture quality is Rank 2**. The rank is lower in this case as the picture has only one color and shape.