E0-270 Assignment-2

Smit Radadiya
Department of Computer Science and Automation,
IISc, Bangalore.
smitradadiya@iisc.ac.in

April 2023

1 K-means Clustering

The task is to cluster the pixels of the image using the K-Means algorithm, with the given number of clusters k = 2, 5, 10, 20, 50.
Here, we are taking one image(3D) and converting it into a 2D matrix to fit into our clustering algorithm.

1.1 Fit

In the fit function, I first define cluster centres using random data points from the matrix. Then using those centres, I call predict function, which returns the closest centre for each data point. And then, I update the cluster centres by taking the mean of all the data points that fall into a particular cluster.

1.2 Predict

In the predict function, I am defining a 1D array of length that is the same as the number of data points to store the cluster number to which it belongs. To do so, I also define a 2D matrix to store the distance of each point from each cluster. Where the index (i, j) represent the distance of i^{th} data point from j^{th} cluster. And finally, take the minimum value and return the column index corresponding to the minimum value.

1.3 Replace with cluster centre

In this section, I first call predict function that returns a 1D array of cluster numbers for each data point. Then replace each row with the cluster mean to which it belongs.

2 Results

I have run the algorithm for the different values of k $[2, 5, 10, 20, 50]$ and the MSE which I get is shown in the table1. Figure3 shows the clustered images w.r.t the values of K.

Value of K	**MSE**
2	0.032
5	0.01
10	0.0048
20	0.0026
50	0.0012

Table 1: MSE for different value of K

2.1 MSE

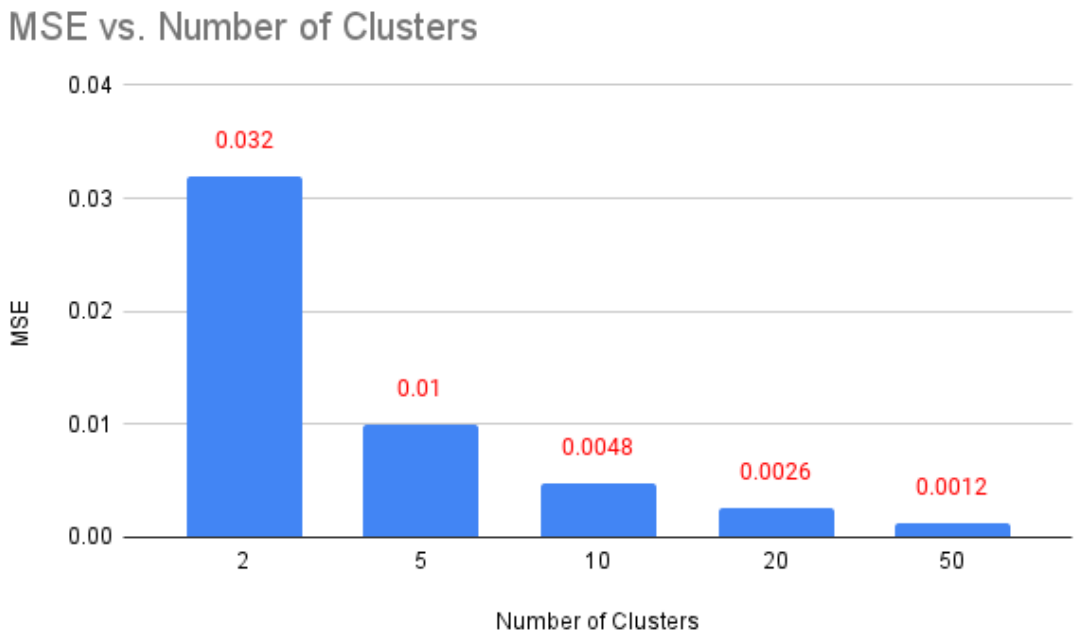

Figure 1: MSE vs Number of clusters

From fig1, we can say that as we increase the value of K the MSE is decreasing. It is because the image gets more colour(centres of clusters), and we reach towards the original image. So MSE is decrease with the increase in the number of clusters. As we can see from fig3 with K=50, it almost looks like same as the original figure. Note that the clustered image with K=20 also captured most of the features of the original image. Send a clustered image with K=20 is require less number of bits to send as compared to the original image.

3 Conclusion

We can use K-means clustering for image segmentation and image compression. As we can get the reasonably same image as the original image after applying the K-means clustering algorithm. Here the choice of K is depend on how much error we can proceed with and, of course, the quality of the image, more clusters lead to finer details. Here it clusters images based on the colour. We can say that clustering is useful for image segmentation, but it does not always produce the most accurate or visually pleasing results.

Figure 2: K=2

Figure 3: K=5

Figure 4: K=10

Figure 5: K=20

Figure 6: K=50

Figure 7: Original image