Homework # 3

CS 550 - Machine Learning

Yusuf Dalva, Bilkent ID: 21602867
yusuf.dalva@bilkent.edu.tr

INTRODUCTION

In Machine Learning, clustering is known as labeling the data without using any kind of knowledge about the data. Considering the concept of clustering, it is a natural use of unsupervised learning. In this assignment, two different unsupervised clustering algorithms are examined which are k-means clustering algorithm and agglomerative clustering algorithm. This report involves details about the implementation of both of the mentioned algorithms and examining different factors in them. All of the factors examined and the outputs of the algorithm are provided in the part dedicated to that specific algorithm. In addition to the implemented algorithms, insights about the data clustered is given in the section *Data Exploration*. The implementation and the discussion mentioned in this report mainly originates from [1].

DATA EXPLORATION

As the initial stage of the assignment, a data exploration stage has been performed to gain some idea about the distribution of the pixels in the given image. This stage has been performed in two ways. As the first way, the pixels in the image are plotted as a scatter plot in 3D. Here the dimensions are based on the color information (Red, Green, Blue). The second way of data analysis is based on image visualization where the RGB components of the image are visualized. In this analysis, it is observed that some parts of the color space in terms of RGB colors (axes) do not contain any kind of data. This fact gives the motivation of using clustering to represent the colors in the image. By using clustering, the image can be represented with a smaller encoding compared to RGB space where each color is in the range [0, 255] computationally. As another issue that leads to using clustering to use the dimensionality of the data is the varying densities of unique pixels. The details of this data analysis stage can be found at the submitted file *Data Exploration.ipynb*.

PART 1: K-MEANS CLUSTERING ALGORITHM

For this assignment, the first algorithm implemented is K-Means clustering algorithm. There are some specific choices made which can be considered as the details of the implementation. After the implementation stage, the algorithm is tested with different K values to see whether it can apply any kind of clustering. Following the success of this step, the optimal K value is searched and the corresponding results are reported. The implementation of the K-Means algorithm is given in *k_means.py* whereas the experiments are implemented in *K-Means Clustering.ipynb* files.

A. Implementation Details

The algorithm consists of three main stages may differentiate from other approaches. The details of these stages are given in the following subsections.

1) Initializing the Centroids (Mean Vectors): In order to identify the clusters, as the first step the mean vectors(centroids) are initialized. As the literature proposes a random initialization as a possible option for the centroids, this approach is followed. This random initialization is done by picking random RGB pixel values from the pixels present in the image to be clustered. In order to prevent the existence of empty clusters, this selection is performed after finding the unique pixel values in the image. With this distinction, the rare case of facing an empty cluster in the initial step is prevented. This case can be obtained if any two centroids have an equal value. In order to randomize the selection of the pixel values, the unique RGB pixels are shuffled first and then the selection is made. This randomization step is performed by the help of numpy random module.

2) Defining Similarity Between Samples: In order to be able to determine that which pixel is going to be assigned to which cluster, a similarity metric has been defined. Considering the nature of the data, the Euclidean Distance is considered as a good metric for measuring similarity.
The formulation of this metric D(x, m_i) for a pixel value x and mean vector m_i is given in the equation below:

$$D(x, m_i) = ||x - m_i||_2 \tag{1}$$

This equation can be expanded to RGB values of the pixel x and mean vector m_i as follows:

$$D(x, m_i) = \sqrt{(x_R - m_{iR})^2 + (x_G - m_{iG})^2 + (x_B - m_{iB})^2} \tag{2}$$

In the equation above, the subscripts R, G and B shows the red, green and blue pixel intensity. Here the pixel intensities are within the range [0, 255].

3) Stopping Criterion: In order to finalize the cluster values, a stopping criterion has been defined. In order to do that, a global approach that covers all of the mean vectors are followed. The stopping criterion can be formulated as follows where M is the matrix of centroid values after update where M_prev is the matrix of centroid values before the update. The dimensions of this matrix is $K \times (Pixel\ Dimensions)$.

$$\Delta M = |M - M_{prev}| \tag{3}$$

$$D = \frac{1}{3K} \sum_{i=1}^{K} \sum_{j \in (R,G,B)} \Delta M_{i,j} \tag{4}$$

TABLE I
CLUSTERING ERROR AND EXECUTION TIME VALUES WITH CHANGING K VALUES

K	2	3	4	5	6
Clustering Error	52.083	41.7107	33.3258	28.8923	27.7400
Execution Time	0.6882	1.3451	1.0648	2.0047	1.3545

If the condition $D \leq 0.1$ is satisfied, the centroid values are finalized. The 0.1 value is selected after several trials and it is considered that 0.1 is a negligible value for the pixel intensities where the intensity values are within the range [0, 255].

B. Results for K values (2,3,4,5,6)

In order to check whether the algorithm is able to perform a sensible clustering, the K-Means algorithm implemented has been tried out with k values (2,3,4,5,6). The clustering error and execution time values obtained with different K values are given in Table I. Here the error value is given in average value of pixel differences and time is given in terms of seconds.
The centroid values can also be seen as follows, each 3D vector given represents a centroid with RGB values respectively:

- **K = 2**: ([245.01 102.93 97.02], [164.47 22.69 20.87])
- **K = 3**: ([208.02 35.89 32.79], [247.75 128.83 124.17], [103.34 14.50 12.30])
- **K = 4**: ([179.5 20.36 21.04], [237.78 64.84 57.83], [72.29 16.17 10.32], [249.35 151.21 146.66])
- **K = 5**: ([243.8 95.61 90.29], [64.59 16.38 9.74], [224.02 41.31 35.57], [250.65 174.25 169.24], [165.65 16.87 18.69]) [165.65 16.87 18.69])
- **K = 6**: ([166.63 14.88 17.45], [64.59 15.87 9.46], [250.66 175.68 170.7], [229.26 40.02 34.49], [246.99 97.18 92.59], [187.49 61.16 50.21])

Finalizing the analysis of applying clustering with K values (2,3,4,5,6), the obtained images after applying clustering are given in Fig. 1.

C. Results for the optimal K value

After trying out different K values , the optimal value for K is searched. In this search the key factor of optimality was both preserving the the variability in the image and not to use too many clusters for memory constraints. In order to perform such a search, a larger variety of K values are tried out. The K values tried out are $K \in (2, 3, 4, 5, 6, 10, 15, 20, 30, 50, 100, 150, 200)$. In order to make a sensible prediction about the count of the natural clusters in the image, the clustering error values (measured in pixel values) are plotted. Referring to [1], it is expected that the drop in the clustering error would be small as the number of clusters are larger than the number of natural clusters. The plot of changing clustering error values and changing amount of clusters is given in Fig. 2. After inspecting this plot, the number of optimal clusters is approximated as K = 30. After running K-Means clustering algorithm on the sample image a clustering error of **13.0161** is obtained. This error is obtained in the by averaging the difference in individual pixel values in the predicted image and the ground truth image. The results obtained are given in Fig. 3.

Fig. 1. Images with 2, 3, 4, 5, 6 clusters and the original image

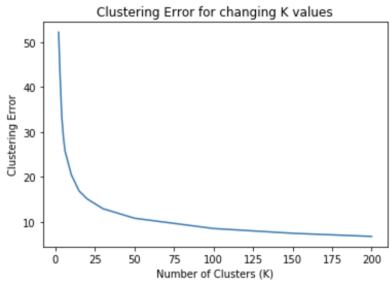

Fig. 2. Plot of changing clustering error with changing amount of clusters

Fig. 3. Ground truth image and image clustered with K = 30

AGGLOMERATIVE HIERARCHICAL CLUSTERING ALGORITHM

The second algorithm implemented for this assignment is agglomerative clustering algorithm. This algorithm is an example of an hierarchical clustering algorithm. Just like the K-Means algorithm, this algorithm also involves some details about the implementation which effects the way that the algorithm works. The implementation details and the results obtained by running this algorithm are explained in this section.

D. Implementation Details

From the nature of the algorithm, this methodology can be effectively used to find hierarchies among clusters. The algorithm starts with N clusters and then continues by merging different clusters depending on how similar they are. In this section, the necessary details about the algorithm implemented are provided.

1) Solving the problem of high computational time: According to the agglomerative clustering algorithm, the main aim is defined as merging cluster that are most similar to each other, one at a time. In an image with high resolution, the most basic form of a cluster would be a single pixel. However, considering the algorithmic complexity of the algorithm, a high amount of clusters would effect the overall process negatively in terms of computational time. In order to overcome this issue, the implemented algorithm makes use of K-Means clustering algorithm implemented before. Considering Fig. 2, the error nearly converges at K = 100. Making use of this fact, the initial N clusters are defined as the 100 clusters obtained from the standard K-Means algorithm with number of clusters equal to 100 (no selection is performed among the clusters).

2) Finding the most similar clusters: In order to find the clusters that are closest in terms of similarity, the similarity of the centroids are considered. To find the similarity of two mean vectors m_i and m_j the euclidean distance is used. The formulation of the dissimilarity of two centroids is given as follows:

$$dissimilarity(m_i, m_j) = D(m_i, m_j) \tag{5}$$

Here the metric D is already defined as Eq. (1) in this report. While selecting a pair to combine with each other the dissimilarity measure is attempted to be minimized. In order to merge two clusters i and j, the maximization stage of the Expectation-Maximization algorithm implemented for K-Means clustering is used. The new cluster D_{new} which is formed by merging D_i and D_j is formulated as follows, where m_{new} is the mean vector for the newly formed cluster and n_{new} is the number of data points included in D_{new}:

$$D_{new} = D_i \cup D_j \tag{6}$$

$$m_{new} = \frac{\sum_{x \in D_{new}} x}{n_{new}} \tag{7}$$

E. Results for K values (2,3,4,5,6,10)

Considering the nature of the algorithm used, which is based on the distances between the mean vectors (centroids), the results obtained were different from the standard K-Means Clustering algorithm. Since the algorithm tries to find the minimum amount of clusters with maximum distance (merging the ones that are close to each other), with too few clusters the mean vectors would have color values that do not represent the majority of the original colors but the centroids that are farthest from each other. The results obtained for execution time and clustering error in terms of pixel values is given in Table II and the resulting images obtained are given in Fig. 4.

TABLE II
CLUSTERING ERROR AND EXECUTION TIME VALUES FOR AGGLOMERATIVE CLUSTERING ALGORITHM

K	2	3	4	5	6	10
Error	82.327	66.996	71.018	41.232	59.450	55.212
Time	81.759	75.347	88.842	86.663	74.725	85.837

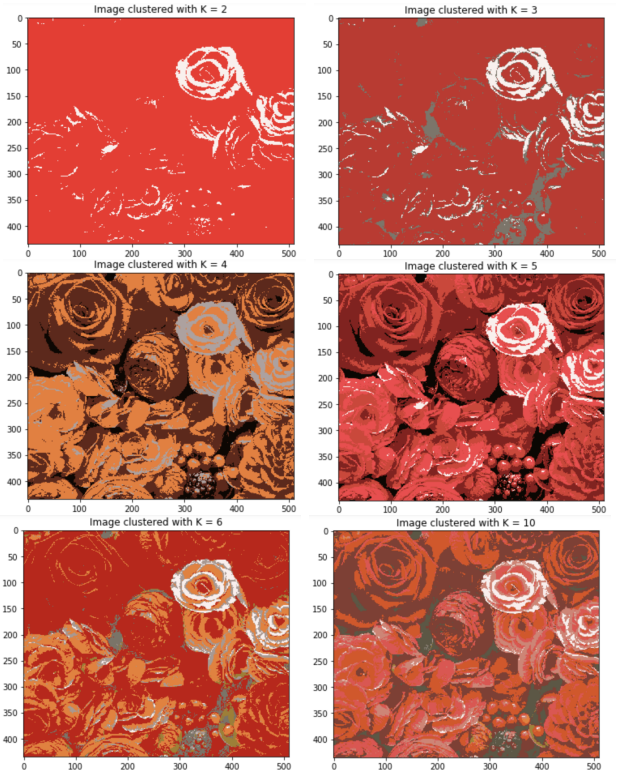

Fig. 4. Images with 2, 3, 4, 5, 6, 10 clusters using Agglomerative Clustering Algorithm

The mean vectors obtained for each K value listed here are also as follows:

- **K = 2:** ([251.81 240.09 237.56], [246.17 36.5 37.4])
- **K = 3:** ([199.49 44.08 41.66], [251.77 241.02 238.33], [126.81 117.63 105.12])
- **K = 4:** ([240.63 123.83 44.2], [14.05 7.14 2.7], [100.53 37.62 23.54], [175.68 161.36 159.11])
- **K = 5:** ([218.35 59.1 49.25], [249.1 62.03 72.48], [251.81 240.11 237.56], [12.01 7.78 2.86], [140.67 23.82 26.57])
- **K = 6:** ([123.11 116.71 103.51], [163.62 127.02 39.11], [171.58 152.09 145.37], [239.73 125.91 43.84], [251.78 239.61 237.1], [197.52 8.89 7.54])
- **K = 10:** ([93.48 87.62 67.51], [174.02 152.05 144.99],

[134.9 60.42 49.32], [224.55 79.97 21.14], [234.9 75.57 76.12], [126.17 119.42 106.71], [217.22 129.49 114.59], [251.78 238.58 236.1], [239.89 125.79 44.05], [163.71 130.08 41.85])

As it can be observed from the outputs shown in Fig. 4, when the Agglomerative Clustering algorithm is used with the specified similarity criteria, the algorithm tends to find centroids that are as far as possible. Considering this fact, as the number of mean vectors decrease the colors obtained tend to be less accurate for the given image. After running this algorithm several times, different centroid values have been observed for small K values in each run. Since the algorithm tries to find the furthest clusters and consider them as hierarchies at the end, with initial clusters that have considered outliers as a cluster would harm the performance for small K values. This fact can be observed from the centroid values and their visualization given in *Agglomerative Clustering.ipynb* for Part 2 and *K-Means Clustering.ipynb* files submitted. When the pixel values of the mean vectors would be compared for small K, it is observed that the final clusters are are distant for agglomerative clustering. This fact is concluded mainly due to the merging methodology used in implementation. Following this observation, it is considered that for optimal amount of hierarchies, the agglomerative clustering algorithm performs better but when the number of clusters would be less than the number of hierarchies present in data, the final prediction would be worse compared to K-Means clustering algorithm. This pattern can be observed in Fig. 4 as K becomes less than 10. To show that Agglomerative Clıstering algorithm is trying to maximize the difference between the remaining clusters, the comparison of the 2 clusters for Agglomerative Clustering and K-Means Clustering are shown in Fig. 5.

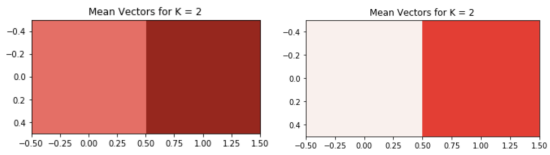

Fig. 5. Pixel values of centroids obtained by using 2 clusters with K-Means Clustering(left) and Agglomerative Clustering(right)

F. Finding the optimal K value

Just like it was performed in the first part, in order to find an optimal value for K, various K values are tried. Referring to 2, the change in the clustering error is negligible when K is greater than 100. Due to this fact, the initial clusters are obtained by using K-Means clustering algorithm where K = 100. In order to find the optimal value for the number of final clusters, the values tried out were from the set (2, 6, 10, 15, 20, 30, 50, 90). The change in the clıstering error with varying number of clusters are shown in Fig. 6. Inspecting this plot, the final significant decrease in the clustering error value is identified between the values (20, 30, 50) as the number of clusters. Following this analysis, it was predicted that with 30 clusters a clustered image which will be a close approximation to the original image can be obtained. The final image where number of clusters is equal to 30 is given in Fig. 7. In order to prove that this is a close approximation to the initial clusters and can successfully identify different clusters, the comparison between using 90 clusters and 30 clusters are provided in Fig. 6. Concluding the analysis about Agglomerative Clustering, the clustering error with 30 clusters is obtained as **23.5158** where it is measured in terms of average of pixel intensity differences.

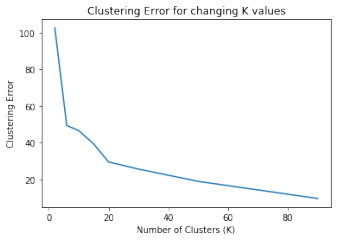

Fig. 6. Changing clustering error with changing amount of clusters using Agglomerative Clustering

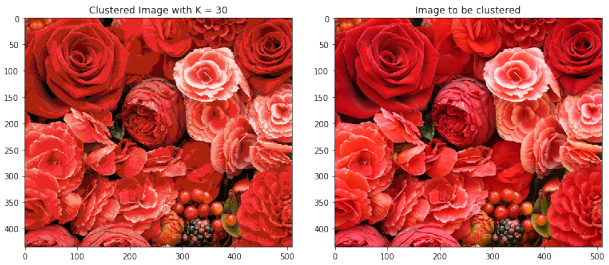

Fig. 7. Image clustered with Agglomerative Clustering with 30 clusters compared to original image

CONCLUSION

As a final remark about agglomerative clustering algorithm implemented, it is observed that the initial cluster is an important factor for performing a clustering operation where the amount of clusters aimed is small. In order to overcome this problem a better cluster selection strategy may be proposed for K-Means clustering algorithm. Also using a different similarity measure is expected to effect the final predictions for an image uses small amount of clusters.

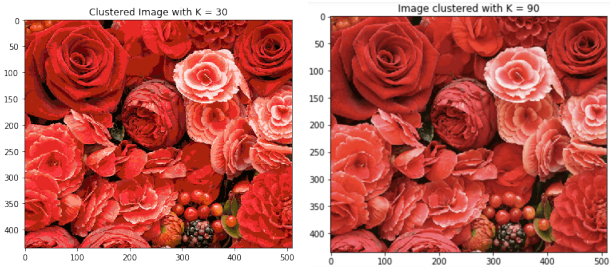

Fig. 8. Comparison of the clustered image with 30 clusters and 90 clusters using Agglomerative Clustering Algorithm

REFERENCES

[1] Ç. Gündüz Demir, *Clustering - CS 550: Machine Learning*, URL: http://www.cs.bilkent.edu.tr/~gunduz/teaching/cs550/documents/CS550_Clustering.pdf.