Clustering

CS 550 Assignment III

Emirhan Koç
Bilkent University
Electrical and Electronics Engineering
emirhan.koc@bilkent.edu.tr

I. Introduction

In this assignment, it is asked to implement K-Means and Hierarchical Agglomerative Clustering (HAC) algorithms and perform experiments on an image with dimension (435, 510, 3). After implementating the algorithm, experiment with different K (cluster number) values and choose K value which has the least error. In addition, display the error values with respect to cluster number and for the HAC; experiment with different initial cluster numbers. Lastly, clustered images are exhibited for different cluster numbers and for both algorithm in each section.

II. K-Means Algorithm

A. Description of Algorithm

Initially, determine the number of the cluster (K) and randomly select K centroids. In this regard, **”Forgy’s Initialization”** is used to initialize centroids such that K random points are selected from the data. Then, Expectation-Maximization is used to determine cluster of each data sample. In Expectation part, each data sample is assigned to the closest centroid and centroids are updated according such that new centroid is determined as the mean of sample points in the corresponding cluster. This process is repeated iteratively until the stopping criterion is met. As a stopping creterion, I decided to check the difference between two consequtive error. If error difference is less than 0.0001, it is wisely to stop as the centroids does not change considerably and algorithm is converged.

Distance metric is determined as L_2-norm in Eq.1 and error metric is determined as mean-squared error in Eq.2:

$$d(x,y) = \sqrt{\left\| \sum_{i=1}^{d} (x_i - y_i)^2 \right\|} \qquad (1)$$

$$MSE = \frac{1}{N_s} * \sum_{j=1}^{K} \sum_{i=1}^{N_j} (x_{ij} - c_j)^2 \qquad (2)$$

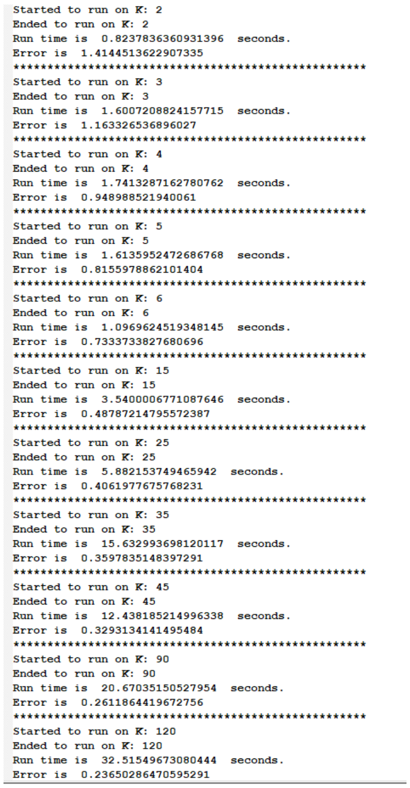
where N_s is the total number of samples. After the centroid of each cluster is determined, each pixel of corresponding cluster is assigned to this centroid.

B. Data Preprocessing and Image Clustering

Initially, we have an image with a shape MxNx3 such that in total MxN pixels exist for each channel. I converted this 3D image matrix into 2D MNx3 matrix and each 1x3 elements is evaluated as a sample point. Therefore, centroids are located in 3D space in the form of (x_i,y_i,z_i). Then, normalization is performed such that each channel’s mean and standard deviations are calculated; from each sample point, mean is substracted and it is divided by standard deviation. Eventually, each pixel value is multiplied with its standad deviation value and its mean is added to it.

C. Results

In the figure below, you can see results of error, training time and clustered images.



```
Started to run on K: 2
Ended to run on K: 2
Run time is  0.8237836360931396  seconds.
Error is  1.4144513622907335
****************************************************
Started to run on K: 3
Ended to run on K: 3
Run time is  1.6007208824157715  seconds.
Error is  1.163326536896027
****************************************************
Started to run on K: 4
Ended to run on K: 4
Run time is  1.7413287162780762  seconds.
Error is  0.948988521940061
****************************************************
Started to run on K: 5
Ended to run on K: 5
Run time is  1.6135952472686768  seconds.
Error is  0.8155978862101404
****************************************************
Started to run on K: 6
Ended to run on K: 6
Run time is  1.0969624519348145  seconds.
Error is  0.7333733827680696
****************************************************
Started to run on K: 15
Ended to run on K: 15
Run time is  3.5400006771087646  seconds.
Error is  0.48787214795572387
****************************************************
Started to run on K: 25
Ended to run on K: 25
Run time is  5.882153749465942  seconds.
Error is  0.4061977675768231
****************************************************
Started to run on K: 35
Ended to run on K: 35
Run time is  15.632993698120117  seconds.
Error is  0.3597835148397291
****************************************************
Started to run on K: 45
Ended to run on K: 45
Run time is  12.438185214996338  seconds.
Error is  0.3293134141495484
****************************************************
Started to run on K: 90
Ended to run on K: 90
Run time is  20.67035150527954  seconds.
Error is  0.2611864419672756
****************************************************
Started to run on K: 120
Ended to run on K: 120
Run time is  32.51549673080444  seconds.
Error is  0.23650286470595291
```

Fig. 1. Error and training time with respect to K

From the figure above, it can inferred that increasing number of clusters bring about the decline in error but increment in training time. According to these results, it is best to choose K between 30 and 45 as it has good trade-off between time and loss

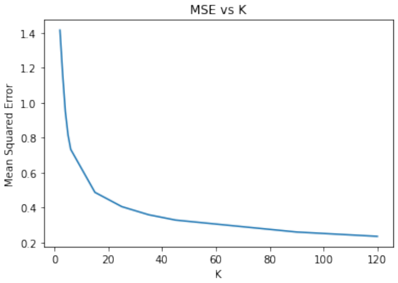


Fig. 2. Increasing number of clusters gives better results in MSE but causes high computational complexity

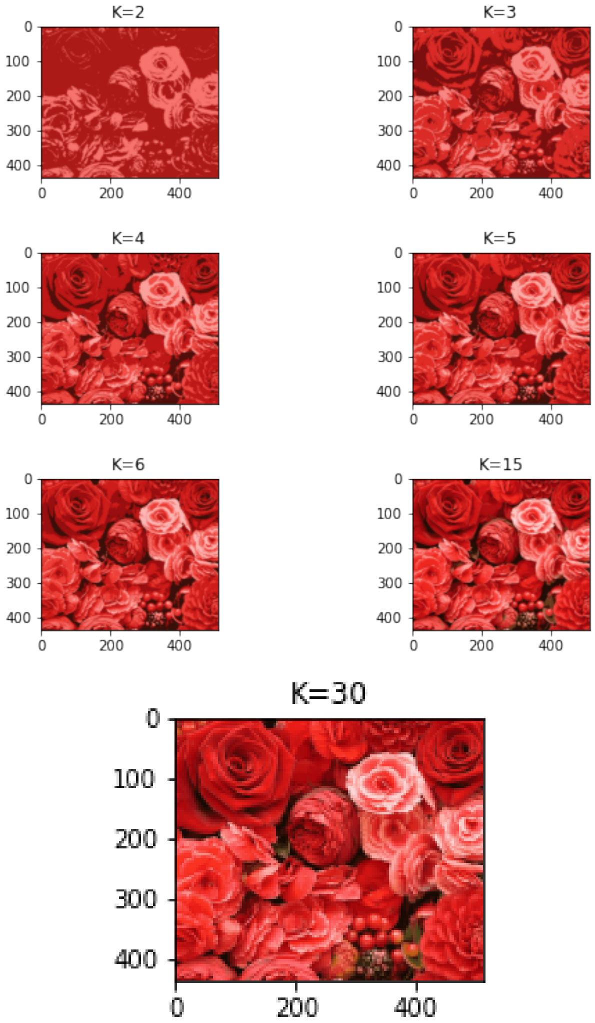


Fig. 3. Clustered images with different cluster numbers

III. Hierarchical Agglomerative Clustering (HAC)

In this part of the assignment, it is asked to implement Hierarchical Agglomerative Clustering algorithm. It follows a bottom-up process such that samples are clustered to N clusters in the beginning and clusters are merged according to their similarity. It is much more complex in terms of time and memory compared to K-means algorithm. Yet, I initially used K-Means clustering to determine centroids and memberships of pixels before merging clusters. Merging clusters depends on the similarity between clusters and two similarity metrics are used: **single linkage** and **complete linkage**

Single linkage

$$S_{min}(c_i, c_j) = \underset{x \in D_i, y \in D_j}{min} \|(x-y)\|^2 \tag{3}$$

Complete linkage

$$S_{max}(c_i, c_j) = \underset{x \in D_i, y \in D_j}{max} \|(x-y)\|^2 \tag{4}$$

A. Description of Algorithm

In the beginning, initial centroids are determined using K-Means clustering and distance matrix where d[i,j] corresponds to minimum sample distance between c_i and c_j. Afterwards, repeatedly, merge two most similar according the distance.Then, update the distance matrix deleting the one of the clusters.Stop this process until you have previously defined number of clusters and the change of error between two consequtive iterations is zero.

B. Data Preprocessing and Image Clustering

Here, similar path as followed in K-Means clustering, normalization and denormalization operations are performed to visualize image at the end.

C. Results

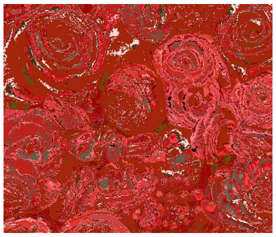


Fig. 4. Clustered images with K=60 in single linkage

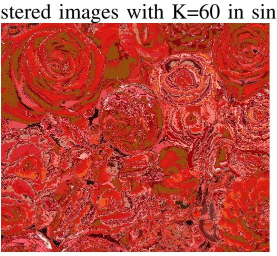


Fig. 5. Clustered images with K= 60 in complete linkage

In all of these experiments, initial K value is taken as K=100. Below, I showed how images are clustered in different

K values.
Clustered images with different cluster numbers

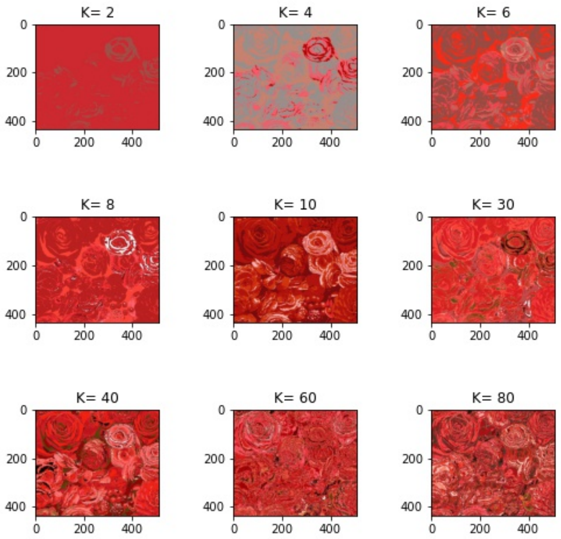


Fig.6 shows how MSE changes with different clustering count

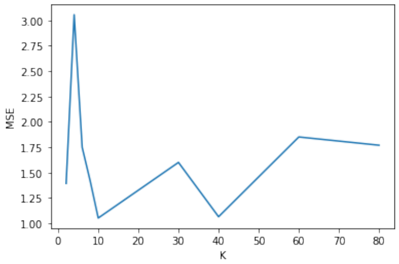


Fig. 6. MSE with respect to different K values

TABLE I
TOTAL RUN TIME WITH DIFFERENT K-VALUES

K	Time (s)
2	272
4	231
6	220
8	210
10	264
30	248
40	248
60	282
80	258