

Assignment 3: Clustering

Due: November 4 2019 11:59PM

Four datasets (Iris, YeastGene, Example, Utilities) can be found on Piazza. In each dataset, each row corresponds to an object and each column corresponds to an attribute. The attribute values are comma-separated. You don't need to do normalization for any of the datasets.

The initial centroids to be used in K-means for Iris and YeastGene datasets are provided. In each file, each row denotes the initial centroid of a cluster.

In this assignment, you are asked to implement K-means algorithm and Agglomerative algorithm (using Min to define inter-cluster distance). Templates (*kmeans_template.py*, *hierarchical_template.py*) are for Python 3.

In *kmeans_template.py*, you are asked to fill in two functions: *assignCluster* and *getCentroid*. In *assignCluster*, each object is assigned to the cluster whose centroid is the closest to the object, based on Euclidean distance. In *getCentroid*, cluster centroids are updated based on current assignment.

In *hierarchical_template.py*, you are asked to fill in *merge_cluster* and *update_distance*. In *merge_cluster*, you need to merge two closest clusters. In *update_distance*, you need to update the distance matrix after the merging using Min as the inter-cluster distance.

You cannot directly call a function or package that implements K-means algorithm and Agglomerative algorithm. You need to implement these algorithms by yourself. If you are not sure about whether it is OK to use a certain function, please post your question on Piazza.

Please take the following steps:

1. Implement K-means algorithm as follows:

Repeat T times:

- For each object x_i
 - Calculate Euclidean distance between x_i and each of the K centroids
 - Assign x_i to the cluster whose centroid is the closest to x_i
- For each cluster
 - Calculate its centroid as the mean of all the objects in that cluster

2. Implement Hierarchical clustering algorithm (with Min as inter-cluster distance definition):

- Obtain the distance matrix by computing Euclidean distance between each pair of objects

- Let each object be a cluster (Assign the cluster index as 1 to N , where N is the number of objects)
- Set the current index as $T=N+1$
- **Repeat**
 - Find the smallest entry in the distance matrix—suppose the entry is i -th row and j -th column
 - Merge the clusters that correspond to the i -th row and j -th column of the distance matrix as a new cluster with index T
 - Remove the rows and columns of the two old clusters and add new row and column for the new cluster to the distance matrix by computing the distance between the new cluster and each of the remaining clusters
 - $T=T+1$
- **Until only one cluster remains**

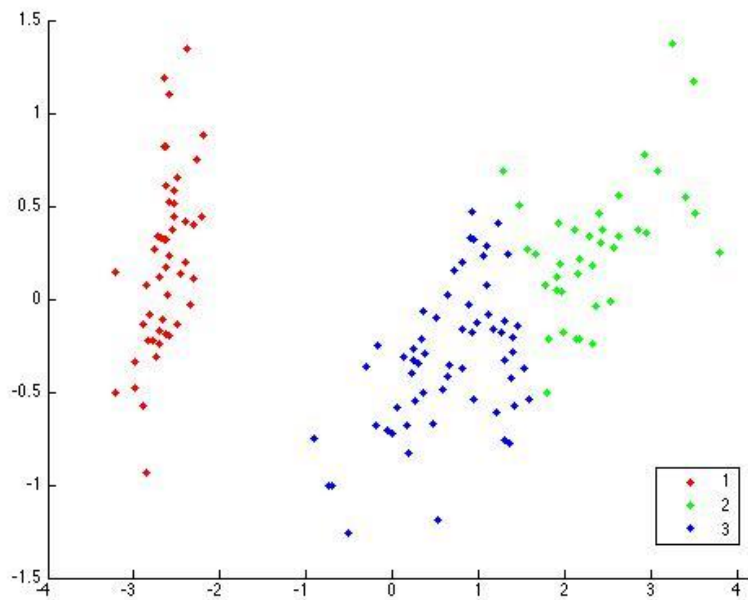
3. Test your K-means on Iris dataset. Use the provided initial centroids, and set the number of iterations (T) as 12, and the number of clusters (K) as 3. The final cluster centroids should be:

Cluster 1: 5.006,3.418,1.464,0.244

Cluster 2: 6.8538,3.0769,5.7154,2.0538

Cluster 3: 5.8836,2.741,4.3885,1.4344

4. After running the filled *kmeans_template.py* code file, you can obtain a data file *Iris_kmeans_cluster.csv*, which contains the Iris data and the assigned clusters. Then, apply the PCA and plotting functions you used in Assignment 1 to project Iris data to 2 dimensions and draw the scatter plot. Use different colors for different clusters in the scatter plot. The plot should look like the one below:



5. If you get the correct final cluster centroids and the plot, then repeat steps 3 and 4 on the YeastGene dataset. Use the provided initial centroids, and set the number of iterations (T) as 7, and the number of clusters (K) as 6.

6. Apply Hierarchical Clustering algorithm implemented in Step 2 on the Example dataset. The order of the merging should be:

1	2	7
3	7	8
4	5	9
6	9	10
8	10	11

Each row here denotes an operation of merging two clusters to form a new cluster. The first two indices denote the clusters to be merged, and the last one denotes the index of the new cluster.

7. If you get the correct order in Step 6, then apply the hierarchical clustering algorithm on the Utilities dataset.

8. Prepare your submission. Your final submission should be a zip file named as Assignment3.zip. In the zip file, you should include:

- A folder “Code”, which contains all the codes used in this assignment. You are required to fill in the template.
- Report: A doc or pdf file named as Assignment3.doc or Assignment3.pdf. The report should consist of the following parts: 1) The cluster centroids obtained on YeastGene dataset after the T iterations. 2) The scatter plot obtained on YeastGene dataset after applying PCA and plotting points using different colors for different clusters. 3) The order of merging in the hierarchical clustering on the Utilities dataset. 4) The codes of your K-means and hierarchical clustering algorithm implementation.

9. Log in any CSE department server and submit your zip file as follows:
submit_cse469 Assignment3.zip

Please refer to Course Syllabus for late submission policy and academic integrity policy. We will take the submission time recorded by the server as the time of your submission. This assignment must be done independently. Running your submitted code should be able to reproduce the results in your report.