MACHINE LEARNING PROJECT

Airline Passenger Segmentation using Single Linkage Divisive (Top-Down) Clustering Technique

22CL60R16 – Akshay Ramesh Bhivagade

OBJECTIVE:

An Indian airline is conducting a customer segmentation analysis to understand their customer's behavior better. For that, they have created a dataset of their customers.

Given the dataset, your task is to cluster the dataset into an optimal number of clusters.

RESULTS:

Optimal Silhouette score (k=3): 0.8586594889992042

Jaccard similarity between cluster 0 of KMeans and cluster 0 th of SLDC: 0.011904761904761904

Jaccard similarity between cluster 0 of KMeans and cluster 1 th of SLDC: 0.1314935064935065

Jaccard similarity between cluster 0 of KMeans and cluster 2 th of SLDC: 0.0008103727714748784

Jaccard similarity between cluster 1 of KMeans and cluster 0 th of SLDC: 0.0

Jaccard similarity between cluster 1 of KMeans and cluster 1 th of SLDC: 0.0076726342710997444

```
Jaccard similarity between cluster  1 of KMeans and cluster  2 th of SLDC:  0.07065217391304347

Jaccard similarity between cluster  2 of KMeans and cluster  0 th of SLDC:  0.0

Jaccard similarity between cluster  2 of KMeans and cluster  1 th of SLDC:  0.185997171145686

Jaccard similarity between cluster  2 of KMeans and cluster  2 th of SLDC:  0.7606456043956044

Average Jaccard similarity: 0.12990846943279744
```

APPROACH:

KMeans:

The CustomKMeans class is defined with k (number of clusters) and max_iteration (maximum number of iterations) as input parameters. The fit method is used to fit the input data X into K clusters by iteratively computing the mean of each cluster until convergence.

In the fit method, the algorithm starts by randomly initializing K centroids, and then assigning each data point to its nearest centroid based on the cosine similarity distance metric. After all data points are assigned to their closest centroid, the centroids are updated by taking the mean of the data points in each cluster. The process of assigning data points to centroids and updating centroids is repeated for a maximum number of iterations until convergence.

The save_final_clusters method is used to save the resultant cluster indices in a file named "KMeans.txt". This method first calculates the distance and cluster index for each point, and then groups the data points according to their cluster index. Finally, it saves the indices of data points for each cluster in the file "KMeans.txt".

Single Linkage Divisive Clustering Algorithm:

The class SingleLinkageDivisiveCLustering has the following methods:

__init__(self, k) : Constructor method which initializes the number of clusters k.

generate_proximity(self, X): This method takes in a dataset X and computes the proximity matrix between all pairs of data points using cosine similarity. The resulting proximity matrix is returned.

generate_mst(self, prox_mat): This method takes in the proximity matrix and generates a Minimum Spanning Tree (MST) using Prim's algorithm.

remove_min_edge(self): This method removes the minimum weight edge (least similarity) from the MST and returns the indices of the two points it connects.

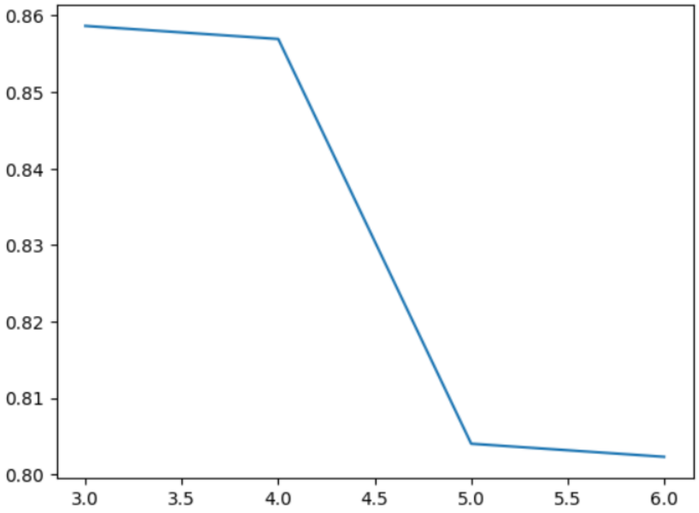
dfs(self, start): This method performs a Depth First Search (DFS) to find all connected data points in a cluster, given a starting point.

fit(self, X): This method performs the actual clustering algorithm. It first generates the proximity matrix and computes the MST. It then iteratively removes the minimum weight edge from the MST until k clusters are formed. Points are added to the corresponding cluster using DFS.

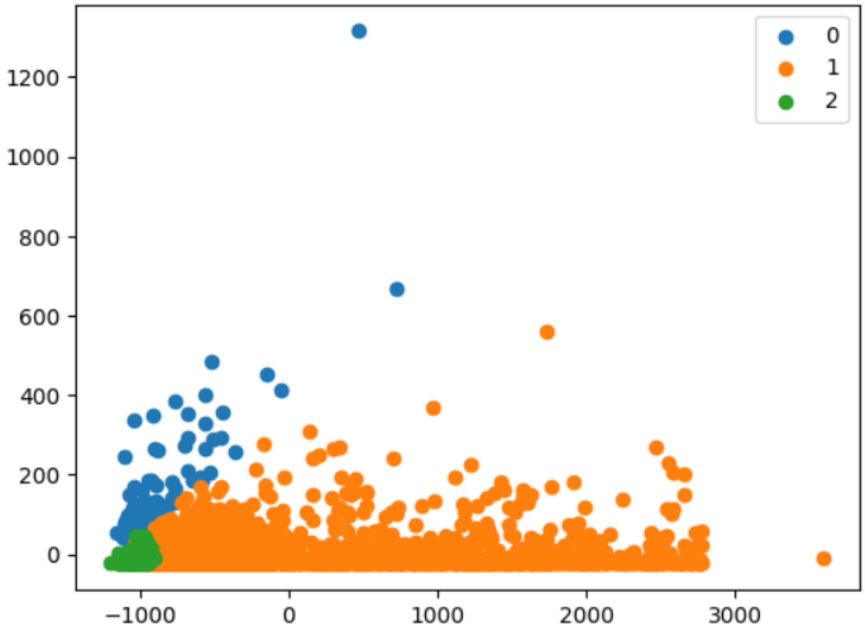
The final output of the fit method is a list of k clusters, where each cluster is represented as a list of indices of data points belonging to that cluster.

VISUALIZATION

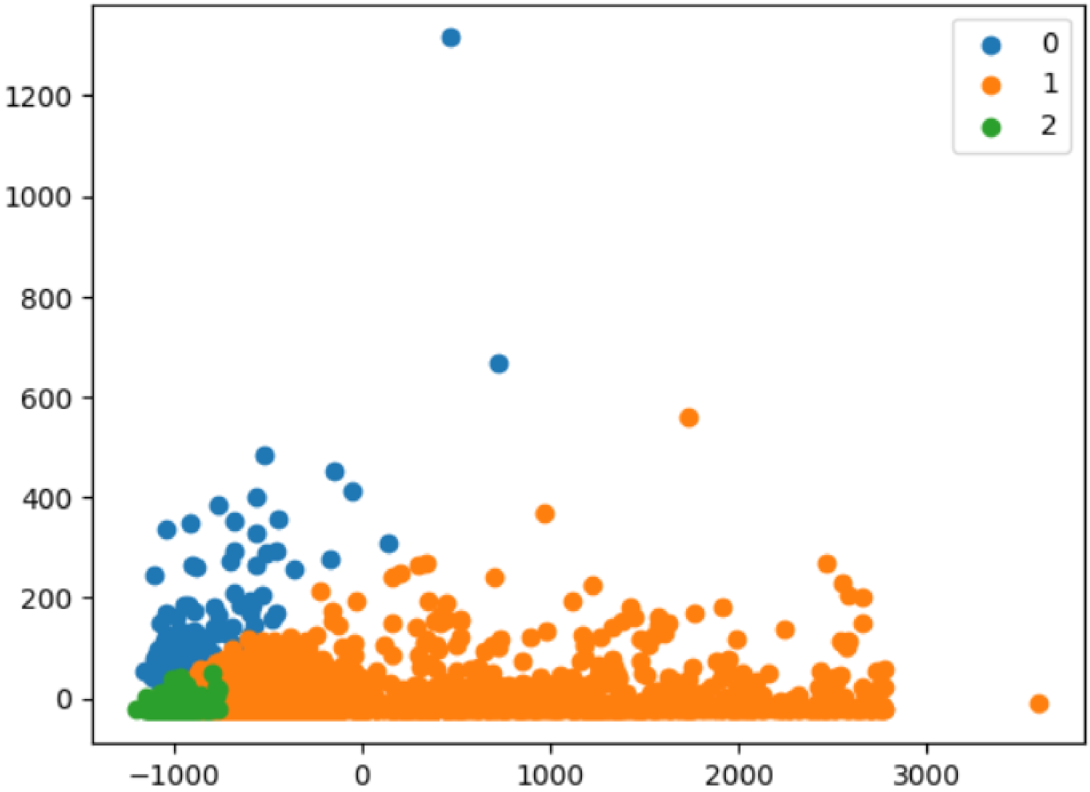
Silhouette scores for different K



Clusters formed according to **my model:**



Clusters formed according to **library function:**



0
1
2
1200
1000
800
600
400
200
0
−1000
0
1000
2000
3000