SIT384 - Secure Coding
Task 8.3HD

Name : Venujan Malaiyandi
Student ID : BSCP|CS|61|101

- **Reasons behind why Agglomerative Clustering algorithm outperformed the others**

  After using grid_search_cv and visual inspection to find the best parameters for all the three unsupervised algorithms; kmeans, agglomerative, DBSCAN. Through the resultant scatter plots and metrics such as grid_search_score, silhoute score and accuracy_score, its understood that Agglomerative Clustering Algorithm performed well.

  The one of the reasons behind this observed behavior is that Agglomerative clustering, being a hierarchical method, is less sensitive to noise compared to K-means and DBSCAN. Agglomerative Algorithm is also comparatively less sensitive to initializing parameters. And lastly, Agglomerative clustering does not assume any particular cluster shape, unlike K-means which assumes spherical clusters and DBSCAN which assumes density-based clusters. These behaviors were strongly displayed in the scatter plots , which were used to visualize the fitted models.

- **Explaining feature scaling and hyper parameter tuning**

  As evident from the attached jupyter notebook, feature scaling for the features were done using min max scaler from sklearn.preprocessing library. This was done to regulate feature 1 and feature 2 whose ranges differed slightly.

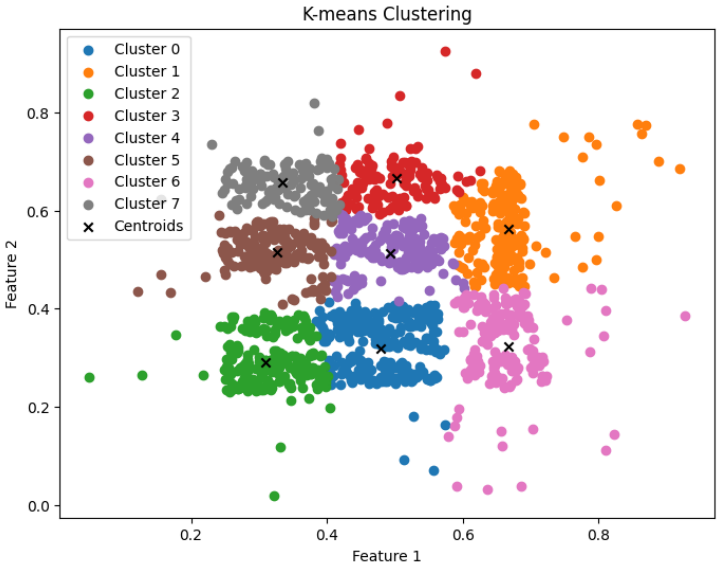
  The parameter tuning was performed through rigorous grid search cross validation with a cv value of 5. For kmeans, the grid_score was used to find the best parameters. For agglomerative, silhouette scoring was used. And lastly, for DBSCAN , visual inspection for different epsilon values were utilized, through exhaustive trial and error.

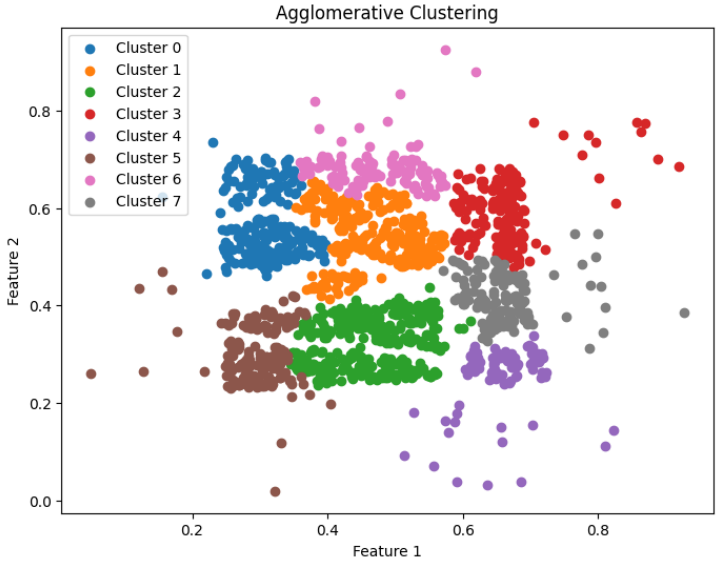
  The best parameters for KMeans
  {'algorithm': 'lloyd', 'copy_x': True, 'init': 'random', 'max_iter': 300, 'n_clusters': 8, 'n_init': 2, 'random_state': None, 'tol': 0.0001, 'verbose': 0}

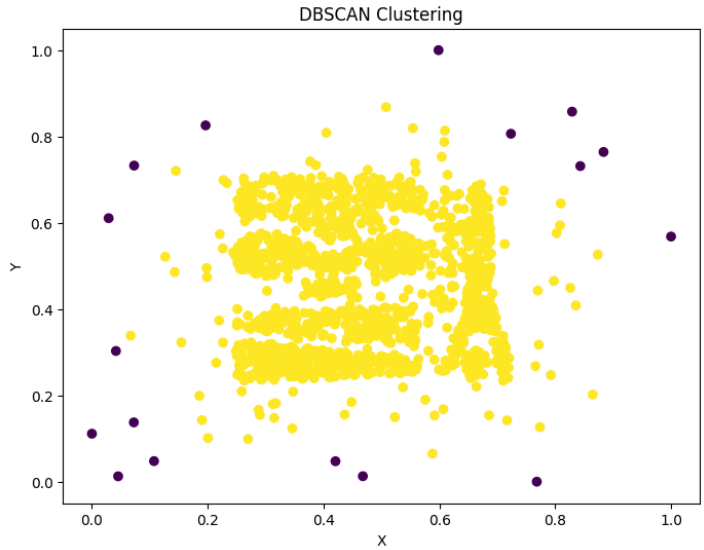
  The best parameter for Agglomerative
  {‘n_cluster’ : 8}

  The best parameter for DBSCAN
  {'eps': 0.1}

[P.T.O]


K-means Clustering
Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6
Cluster 7
Centroids
Feature 2
Feature 1
0.0
0.2
0.4
0.6
0.8


Agglomerative Clustering
Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6
Cluster 7
Feature 2
Feature 1
0.0
0.2
0.4
0.6
0.8


DBSCAN Clustering
Y
X
0.0
0.2
0.4
0.6
0.8
1.0

- **Summary**

  This task began with a a 2-dimensional dataset called Complex8_RN15, with is the variation of the Complex8 dataset with 15% gaussian noise added to the original Complex8 dataset. The dataset contained two features and a class column. To this dataset (with noise), three unsupervised clustering algorithms such as KMeans, Agglomerative, DBSCAN were fitted. The hyper parameters needed for these tasks were found using hyperparameter tuning techniques such as visually(scatter plot), elbow scaling(for clusters), and grid search (for other hyper parameters). From the finalized models for each unsupervised clustering algorithm, respectively. Agglomerative Algorithm outperformed the others.

  This is due to the properties of Agglomerative Algorithm. This algorithm does not assume any particular cluster shape. Agglomerative algorithms are also less affected by the initializing parameters. Additionally, Agglomerative clustering naturally produces a hierarchical structure (dendrogram), allowing for exploration of clusters at different levels of granularity. This is helpful when dealing with noisy data, such as the given dataset.