CSDS 435 Project #2

Ethan Ho

March 2025

1 Introduction

This report analyzes the integrity of three different clustering algorithms applied on two datasets representative of real-world data. The layout of this exploration will be constructed as follows. First, a theoretical description of each of the three algorithms will be provided along with a brief explanation of each algorithm's application in the context of the datasets. Then a visual analysis of the three clustering algorithms applied on the datasets will be shown. Finally, a technique of evaluation will be discussed, along with its result applied to the three algorithms and two datasets.

2 Clustering Algorithms

2.1 K-means Clustering

The basic idea behind the K-means clustering algorithm is to iteratively select "centroids", or points in the data that best represent the central location of a cluster of data. For each iteration, each centroid is refined (if possible), such that each is positioned more optimally and accurately with respect to the determined clusters of data. The "K" component in K-means clustering defines the number of clusters to create, and consequentially, the number of centroids to pick. Initially, the dataset chooses K random centroids among the set of data. These centroids are then refined by recomputing the centroid of each cluster per iteration. This process continues until the centroids remain unchanged after a second iteration or an enforced limit (e.g., $n = 100$ iterations) has been reached.

2.2 Agglomerative Clustering

The agglomerative clustering algorithm is a type of hierarchical clustering algorithm where the clusters are formed by merging smaller clusters. Initially, each data point is considered its "own" cluster. Then for each iteration, the two "closest" clusters are merged into one cluster. This process repeats until only a single cluster remains, but a hierarchy of clustering has formed.

The process of merging clusters with respect to distance is intuitive when the clusters are individual data points, but complications arise when determining the distance between clusters with multiple data points each. As a result, there are multiple different strategies that can be employed to determine the distance between two arbitrary clusters. The two strategies applied in this report's specific analysis are Ward's Method of squared error and group average. Ward's Method calculates the similarity of two clusters by the increase in squared error when the two clusters are merged. This strategy is less susceptible to noise, making it an ideal algorithm in certain datasets. Group average is another strategy for calculating the distances between clusters. Essentially, the average location of the cluster points is used as a "central" point to determine distance against group averages of other clusters. The group average technique is also generally less susceptible to noise, making it a solid choice in some cases.

2.3 Density-based Clustering

Density-based clustering works in creating clusters from adjacent groups of dense cells. First, a "grid system" is defined among the data points. Each data point is then assigned to the cell in which it resides in the grid. Finally, clusters are formed and defined from contiguous groups of dense cells. A "dense cell" is defined by a cell that has a sufficient density level defined by threshold τ. Density-based clustering is generally very efficient, as it is not an iterative process and can determine clusters in a single pass through the data. However, its precision is highly dependent on the determined threshold τ, making it not an ideal model in some cases.

3 Visual Analysis

In order to visualize these clusters, the data points and their attributes had to be reduced into a form that could be visually seen. To achieve this, a technique of principal component analysis was applied on the values, reducing the dimensions of the data while keeping important patterns and traits of the set. For each of the following figures, a specific color represents a cluster.

3.1 K-means Clustering Visualization

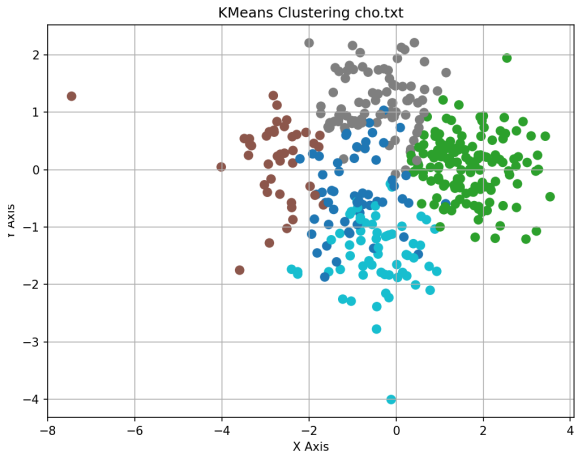

Figure 1: K-means clustering visualization on "cho.txt"

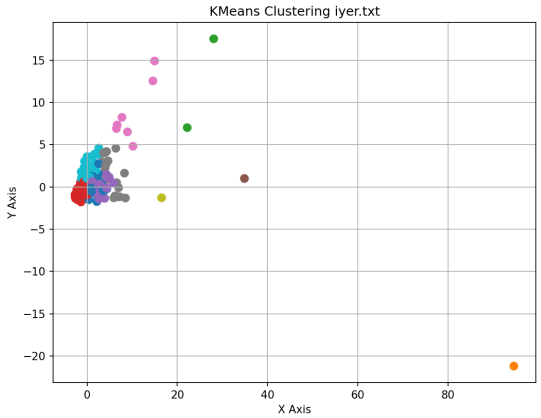

Figure 2: K-means clustering visualization on "iyer.txt"

3.2 Agglomerative (Ward) Clustering Visualization

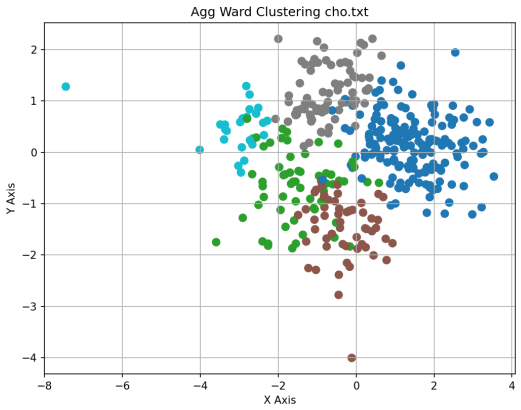

Figure 3: Agglomerative (Ward) clustering visualization on "cho.txt"

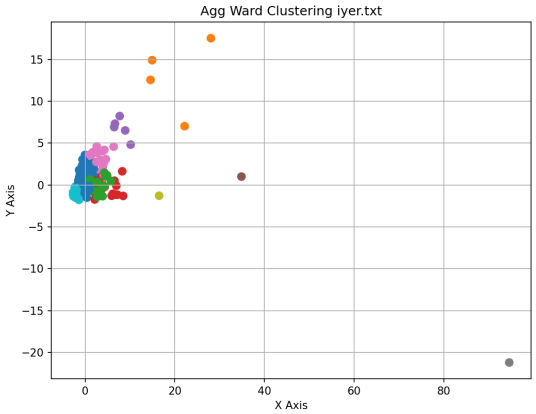

Figure 4: Agglomerative (Ward) clustering visualization on "iyer.txt"

3.3 Agglomerative (Avg) Clustering Visualization

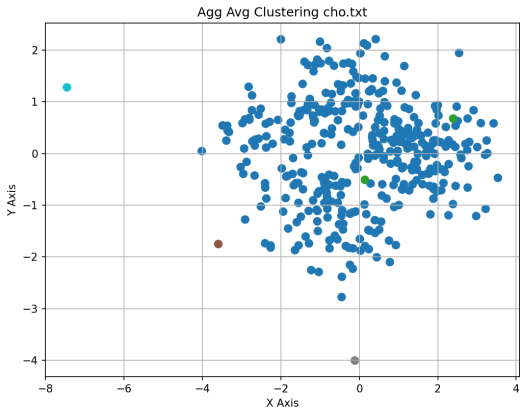

Figure 5: Agglomerative (Avg) clustering visualization on "cho.txt"

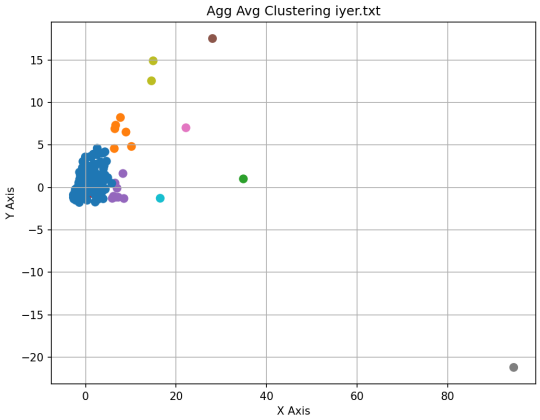

Figure 6: Agglomerative (Avg) clustering visualization on "iyer.txt"

3.4 Density-based Clustering Visualization

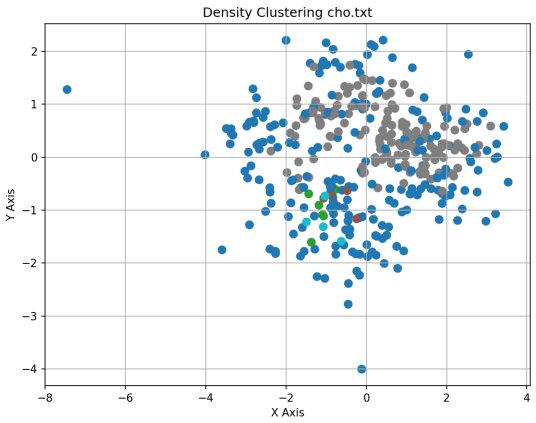

Figure 7: Density-based clustering visualization on "cho.txt"

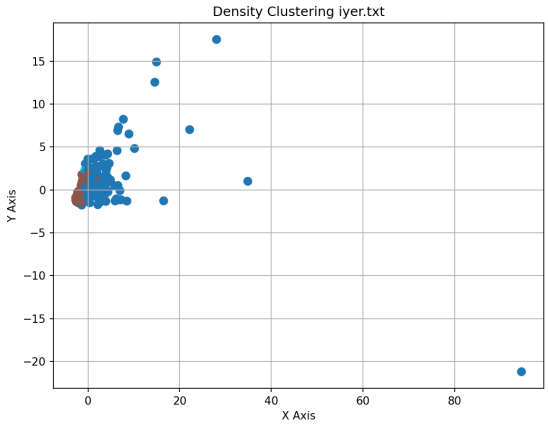

Figure 8: Density-based clustering visualization on "iyer.txt"

4 Result Evaluation

To evaluate the performance of each of the models, the obtained clusters were compared against ground true cluster values provided. The measurement used to achieve this goal was the Rand index (or Rand score), which gives a quantitative value from 0 to 1 indicating how similar the clustering is. A score of 0 indicates no match, while a score of 1 indicates a perfect match. Below is a table showing the Rand score for each of the algorithms against each of the data sets.

Method	cho.txt	iyer.txt
K-Means	0.787	0.678
Agg (Ward)	0.784	0.701
Agg (Avg)	0.243	0.228
Density	0.534	0.544

For K-Means, experimenting with the number of clusters revealed that for "cho.txt", the ideal number of clusters was 5, and for "iyer.txt", the ideal number of clusters was 10. These clustering values were also used for agglomerative clustering, revealing decent Rand scores for Ward (squared error) Method and poor scores for group average method. For density-based clustering, the ideal eps (maximum distance values) for the "cho.txt" and "iyer.txt" were 1.02 and 1 respectively.

It is also worth noting that in terms of visualization, results with higher rand scores more distinctly cluster the data points into isolated units. For example, for "cho.txt", the K-Means Rand score against the ground true values was calculated to be 0.787, which is relatively high compared to the rest of the scores. Figure 1 shows the visualization of this result, and as expected, the clusters are visually separated by color in true clustering fashion.