INT104 CW3 Lab Report

Mingxuan Hu 2252534
TA: Jiahui Hu

***Abstract*—This lab report presents an exploration of unsupervised learning through the application of three distinct data clustering methodologies: Gaussian mixture model, K-means and hierarchical clustering, with the assistance of Python. The aim was to identify patterns and structures within a given 11-dimensional data set, specifically focusing on the distribution of Programme column information. The data set was first subjected to t-SNE dimensionality reduction to simplify the clustering process. Each method's effectiveness was evaluated using various metrics, including Bayesian information criteria, Akaike information criteria, silhouette coefficient and cophenetic coefficient. Despite the successful implementation of these algorithms, the study found little correlation between the clusters produced and the original programme affiliations.**

***Index Terms*—Unsupervised Learning, Gaussian Mixture Model, BIC, AIC, K-means, WCSS, Silhouette Coefficient, Hierarchical Clustering, Cophenetic Coefficient, Pearson Correlation Coefficient**

I. Data Clustering

Data clustering is an unsupervised learning method in machine learning that is used to discover inherent patterns and structures in data sets. The main goal of this technique is to categorize the data in such a way that data points within the same group have high similarity, while those from different groups exhibit low similarity. There are many algorithms for data clustering, including Gaussian mixture model, K-means clustering, hierarchical clustering, DBSCAN, spectral clustering and so on. In this experiment, three different data clustering algorithms will be used to cluster the given data set, in order to observe the distribution of Programme column information.

It is worth noting that since the data set given in this experiment is a 11-dimensional data (excluding the Index column), data clustering is difficult to achieve. Therefore, the data set used in the subsequent experiment is the data after t-SNE dimensionality reduction of the original data to two dimensions. The scatterplot of data after dimensionality reduction is shown in Figure 1.

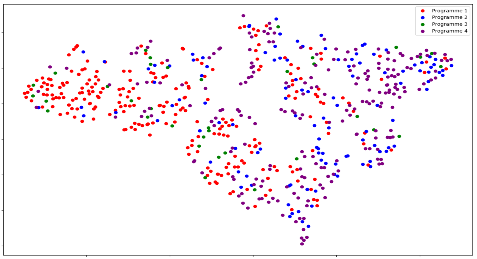

Fig. 1. Data Distribution after t-SNE Dimension Reduction

II. Gaussian Mixture Model

The first method of data clustering is Gaussian mixture model.

Gaussian mixture model, or GMM for short, is a soft clustering model that describes the statistical properties of a data set by linear combinations of multiple Gaussian distributions. In GMM, each Gaussian distribution is treated as an independent cluster with its own specific mean and covariance, which together describe the probability distribution properties of the data. Although a single Gaussian distribution can only model single-peak data, by mixing multiple Gaussian distributions, the GMM is well adapted to multi-peak data and is able to accommodate different shapes and sizes of the data.

In addition, when determining the number of clusters in a Gaussian mixture model, Bayesian information criteria (BIC) and Akaike information criteria (AIC) can be used to measure the advantages and disadvantages of models with different numbers of clusters. The model can be trained with varying numbers of clusters, after which the BIC and AIC values for each model can be calculated. The model with the lowest BIC or AIC value is generally considered the best model. This is because a low BIC or AIC value indicates that the model avoids overfitting caused by over-complexity while maintaining a good degree of fitting to the data. For the data set given in this experiment, the corresponding BIC and AIC values are shown in Figure 2.

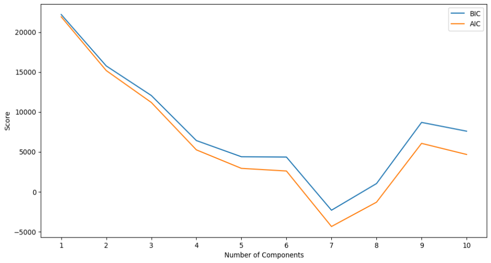

Fig. 2. AIC and BIC Scores for Gaussian Mixture Model

According to the data in the figure, for the data set of the experiment, the optimal number of clusters of the Gaussian mixture model is 7. However, considering the practical significance of this experiment, the clusters finally divided need to be connected with students' programmes. Therefore, the number of clusters in the Gaussian mixture model is finally determined to be 4.

Therefore, the Gaussian mixture model was applied to the data set, and the four clusters obtained were shown in Figure 3. The figure also shows the silhouettes of the four clusters, represented by black ovals. It can be seen that the obtained four clusters are obviously separated, and the data point aggregation trend within each cluster is obvious, with a certain degree of differentiation, which is convenient for subsequent analysis of information distribution.

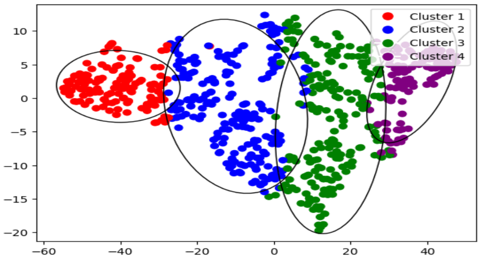

Fig. 3. Clusters of Gaussian Mixture Model with Silhouette

In addition, when implementing a Gaussian mixture model in Python, the choice of some parameters is crucial. For example, the ”covariance_type” parameter is used to specify the covariance type for each Gaussian distribution. This parameter determines the distribution shape of the data points in each component. It has four possible values: ’full’, ’tied’, ’diag’, and ’spherical’. By detecting the silhouette coefficient (which will be detailed in the third part of the report) of the clusters, the best clustering effect is achieved when the parameter value is set to ’spherical’, where the average silhouette coefficient between the four clusters is 0.34.

III. K-means

The second data clustering algorithm is K-means.

K-means is an iterative and partition clustering algorithm, whose core idea is to minimize the sum of squares of distances between sample points in the cluster, so as to take it as a clustering measure. It starts with a random initial cluster center and iterates by assigning each sample to the nearest cluster center and then recalculating the center point of each cluster. The algorithm stops until the position of cluster center no longer changes.

It is worth noting that the number of clusters k is a hyperparameter. Therefore, the optimal value of k needs to be determined before K-mean algorithm is carried out. There are generally two ways to choose the best k value, the first way is called the elbow method. By drawing the clustering cost curve corresponding to different k values, the method finds the ”elbow” of the curve, that is, the inflection point of the curve, as the best clustering number. This inflection point represents that adding more clustering leads to a significant slowdown in the rate of decline in clustering costs.

Clustering costs are usually measured using the intra-cluster sum of squares (WCSS). For each cluster, WCSS is the sum of the squares of the Euclidean distances from all points in the cluster to the cluster center. The smaller the value of WCSS, the better the clustering, because the data points are closer to their clustering centers.

Another indicator for determining the best k value is the silhouette coefficient. The silhouette coefficient is a kind of index to measure the clustering effect, which takes into account the similarity within clusters and the difference between clusters. Its value is between -1 and 1. When the silhouette coefficient is close to 1, it means that the sample has high similarity with other samples in the cluster, but low similarity with other samples in the cluster, and the clustering effect is better. Therefore, when using the silhouette coefficient to select the best k value, the k value that makes the silhouette coefficient maximum should be selected.

For the data set given by the experiment, the best k value can also be determined by the above two methods. Figure 4 records the WCSS for different k values in the elbow method, and Figure 5 calculates the silhouette coefficients for different k values. Combining the information of the two figures, the optimal k value for the experimental data set should be 2. However, for the same reason as before, because the clusters should be associated with the student’s programmes, the choice of k value here is still 4.

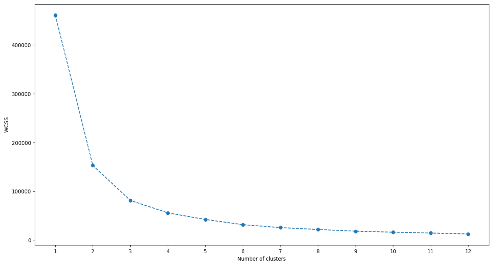

Fig. 4. Elbow Method for Optimal k

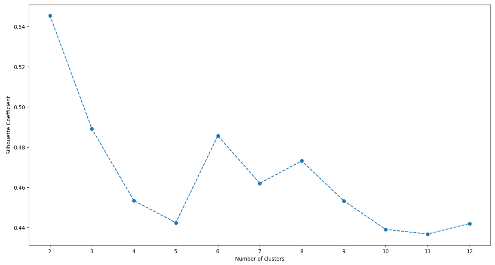

Fig. 5. Silhouette Method for Optimal k

Therefore, Figure 6 shows the result of K-means clustering on the dataset when the value of k is 4. The figure also shows the locations of the four centroids. The average silhouette coefficient between the four clusters is 0.45.

IV. Hierarchical Clustering

The third method of data clustering is hierarchical clustering.

Hierarchical clustering is a clustering algorithm that divides data sets into small groups or subgroups by establishing a hierarchical structure. This method can be carried out in a top-to-bottom or bottom-to-top manner, which are called agglomerative and divisive respectively.

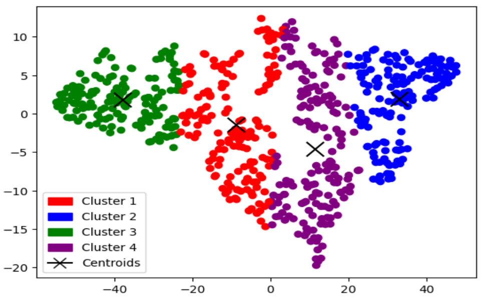

Fig. 6. Clusters of K-means with Centroids

In agglomerative hierarchical clustering, each data point is initially treated as an independent cluster, and then the algorithm continues to combine the closest pair of clusters into a new cluster until all data points are combined into a single cluster, forming a hierarchical structure. In divisive hierarchical clustering, all data is treated as one large cluster at the beginning, and then at each step the most separable of the existing clusters is split into two new clusters. This process continues until each data point becomes a separate cluster.

In this experiment, the agglomerative hierarchical clustering method is adopted, and the results are shown in Figure 7. After calculation, the silhouette coefficient of the cluster is 0.44.

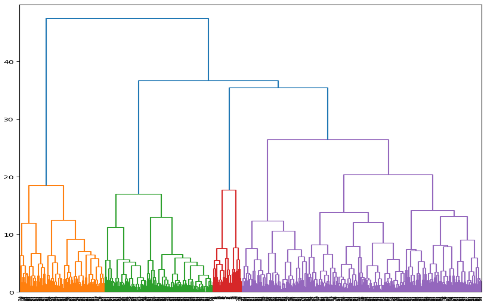

Fig. 7. Hierarchical Clustering Dendrogram

In addition, cophenetic coefficient is another index used to measure the stability and confidence of hierarchical clustering results. It is primarily used to verify the reliability of the dendrogram, measuring the consistency between the distance between the original data points and their correlation in the tree. The cophenetic coefficients are calculated by calculating the heights at which they are joined in the tree for each pair of data points, and then calculating the Pearson correlation coefficient of the Euclidean distance between these heights and the original data points. Its value is between -1 and 1, and the closer the value is to 1, the more consistent the results of hierarchical clustering are with the original distance matrix, that is, the better the quality of the clustering results.

After calculation, the cophenetic coefficient value of the hierarchical clustering dendrogram generated by the experimental data set is 0.59. This shows that the dendrogram reflects the distance relationship of the original data to some extent, but there are some deviations at the same time, and the accuracy of clustering needs to be further improved.

V. Association of Clusters and Programmes

Upon completion of the data clustering and observation of the information distribution in the Programme column, it's necessary to evaluate the relationship between the clusters and the programmes. Since the clusters generated by the K-means algorithm have the largest silhouette coefficient, they are used here for analysis.

There are many criteria to judge whether a cluster is related to a programme. The simplest method is to calculate the number of scattered points in the cluster. The result is shown in Figure 8. It can be analyzed that the number of scattered points in each cluster is relatively close, about 150. However, there is a large gap between the number of students in each programme, from as little as 30 to as much as 250. This means that the cluster at least does not correctly reflect the number of students in each programme.

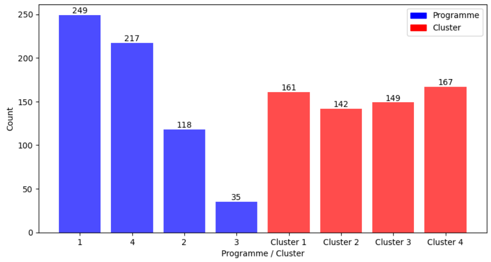

Fig. 8. Counts of Data Points for Each Programme and Cluster

In addition, as shown in Figure 9, each cluster contains students from four programmes, and students from each programme are also present in the four clusters. There is also no significant difference in the proportion between them, suggesting that the clusters also do not correctly reflect the programme situation from which the students come.

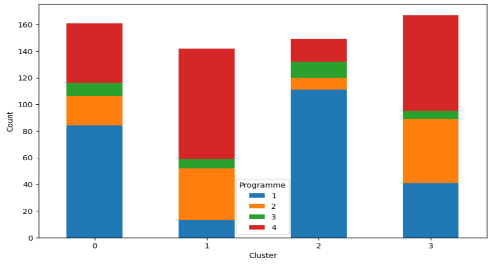

Fig. 9. Programme Distribution in Each Cluster

Finally, the Pearson correlation coefficient between the clusters and programmes can be calculated. The Pearson correlation coefficients for the four programmes are 0.15, 0.29, -0.23 and 0.007 respectively. This shows that there is very little or no linear correlation between clusters and programmes.

To sum up, for the dataset in this experiment, there is little connection between the clusters generated by the data clustering and the programmes from which the students come.