CW3 report

Enze Zhou, Student ID:2254411
TA:Zhejun.Yang

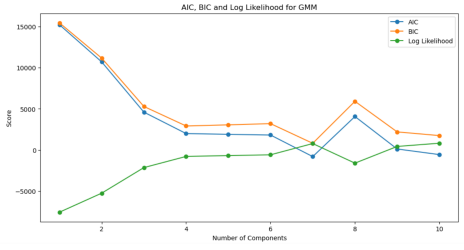
I. Gaussian Mixture Model

In this analysis, I first selected gender, grade, total score, MCQ, Q1, Q2, Q3, Q4, Q5 and other features, and divided the data set into training set and test set, as shown in Figure [1]. Next, I used Gaussian mixture model (GMM) to fit the training data, and selected the optimal number of model components through AIC, BIC and likelihood evaluation, as shown in Figure [2]. According to the line chart, I could judge that the optimal number of GMM components should be 7, so I used 7 components in the subsequent training of the model. In order to evaluate the clustering effect, I calculated the posterior probability of the training set sample, and used the contour coefficient, Calinski-Harabasz index and Davies-Bouldin index to measure the clustering quality.

[1]

```
#分离特征

from sklearn.model_selection import train_test_split
# 2. 选择特征
features = ['Gender', 'Grade', 'Total', 'MCQ', 'Q1', 'Q2', 'Q3', 'Q4', 'Q5']
X = data1[features]  # 特征

# 3. 分割数据集为训练集和测试集（不需要分割成X和y）
X_train, X_test = train_test_split(X, test_size=0.2, random_state=42)
```

[2]



Then, I associate each cluster in the training set with the program label, as shown in Figure [3], and use the voting method to determine the program label corresponding to each cluster, and verify it in the test set. In addition, in order to see the clustering results more directly, I conducted PCA dimensionality reduction on the data,like Figure [4], and visualized the data and clustering results after dimensionality reduction, which showed that the effect was not good.

[3]

```
sil_score = silhouette_score(X_train, y_train_pred)#轮廓系数
calinski_harabasz = calinski_harabasz_score(X_train, y_train_pred)#Calinski-Harabasz指数
davies_bouldin = davies_bouldin_score(X_train, y_train_pred)#Davies-Bouldin指数
print(f'Silhouette Score: {sil_score:.2f}')
print(f'Calinski-Harabasz Score: {calinski_harabasz:.2f}')
print(f'Davies-Bouldin Score: {davies_bouldin:.2f}')

# 投票方式关联聚类簇与项目
cluster_programme_mapping = {}
for cluster_label in set(y_train_pred):
    cluster_indices = (y_train_pred == cluster_label)
    cluster_programme = y_train[cluster_indices].mode().values[0]
    cluster_programme_mapping[cluster_label] = cluster_programme

# 在测试集上进行投票
y_test_pred_associated = np.array([cluster_programme_mapping[cluster_label] for cluster_label in y_test_pred])

# 计算准确率
accuracy = accuracy_score(y_test, y_test_pred_associated)
print(f'Accuracy: {accuracy:.2f}')
```

```
Silhouette Score: -0.29
Calinski-Harabasz Score: 76.07
Davies-Bouldin Score: 4.37
Accuracy: 0.60
```

[4]

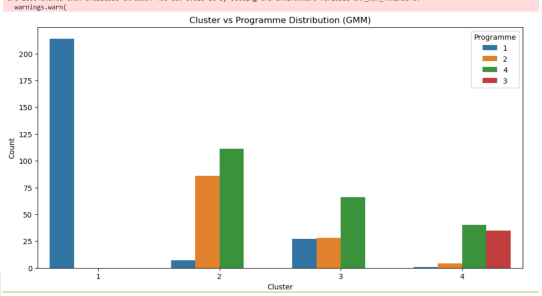
```
# 绘制PCA降维后的数据和聚类结果
plt.figure(figsize=(12, 6))
scatter = plt.scatter(X_train_pca[:, 0], X_train_pca[:, 1], c=y_train_pred, cmap='viridis')
plt.xlabel('PCA Component 1')
plt.ylabel('PCA Component 2')
plt.title('PCA of GMM Clusters (Training Set)')
plt.colorbar(scatter, label='Cluster Label')
plt.show()
```

Finally, I displayed the clustering results through principal component reduction and visualization, as shown in Figure [5], and analyzed the distribution of each program label in each cluster. However, it can be seen that except cluster1, there are only programme 1, and other clusters have many programmes. This means that it has little to do with the programme, almost nothing to do with it.

[5]

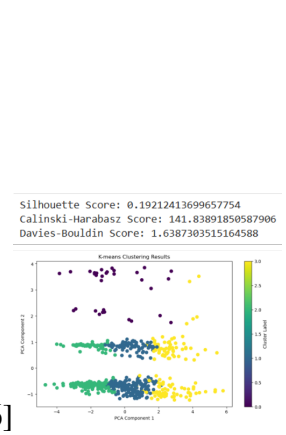
```
# 计算每个聚类中每个程序的数量
cluster_programme_count = cluster_programme_df.groupby(['Cluster', 'Programme']).size().reset_index(name='Count')

# 可视化每个聚类中各程序的数量分布
plt.figure(figsize=(12, 6))
sns.barplot(x='Cluster', y='Count', hue='Programme', data=cluster_programme_count)
plt.title('Cluster vs Programme Distribution (GMM)')
plt.xlabel('Cluster')
plt.ylabel('Count')
plt.legend(title='Programme')
plt.show()
```

```
G:\AIvoice\Anaconda\Lib\site-packages\sklearn\cluster\_kmeans.py:1382: UserWarning: KMeans is known to have a memory leak on Windows with MKL, when there are less chunks than available threads. You can avoid it by setting the environment variable OMP_NUM_THREADS=3.
  warnings.warn(
```

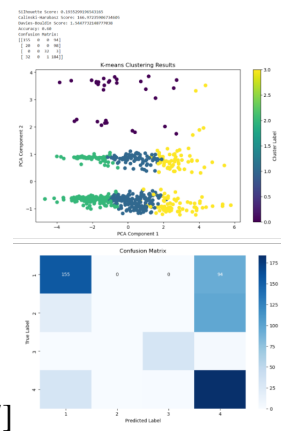


I do cluster analysis mainly by using GMM,

where we are able to reveal hidden patterns and structures in the data set, especially the relationships between different features. By associating the clustering results with program labels, we can better understand the classification information in the data, helping to improve the classification model and make more accurate decisions in practical applications. This analysis method not only improves the interpretability of the data, but also provides a valuable reference for the subsequent model optimization.



[6]



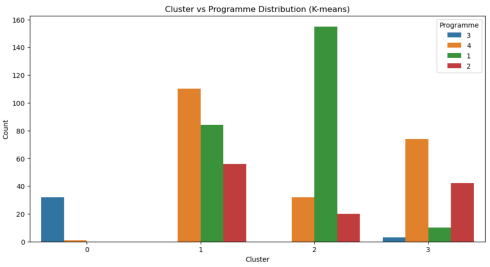
[7]

II. K-MEANS

I performed K-means clustering analysis on the data to explore relationships between features and map the results to actual item labels. First of all, I standardized the feature data to eliminate the influence between different dimensions. Then I use K-means algorithm to cluster the data, and since there are four programmes, I set four clusters. As shown in Figure [6], I calculated the contour coefficient, Calinski-Harabasz index and Davies-Bouldin index to evaluate the clustering quality, assess the compactness and separation of clusters, and reduce and visualize the clustering results.

I want to know whether each cluster is associated with the programme, so I associate each cluster with the actual programme label to determine the most common project label in each cluster. Finally, I categorize the data set, predict the cluster label for each sample, and map it back to the actual item label. The accuracy of classification is calculated and confusion matrix is generated to judge whether my clustering effect is good, as shown in Figure [7].Moreover, I used PCA to reduce the dimension of the data to two dimensions, and visualized the clustering results and confusion matrix, so that the results could be seen more intuitively.

In order to find out whether each cluster has any relationship with the Programme, I analyzed the quantity distribution of each project label in each cluster and presented it visually, as shown in Figure [8]. However, it can be seen from the results that this is similar to the effect of GMM, only cluster1 may have a little relationship with the programme. Other clusters are almost unrelated, and we can say that this cluster is not associated with the programme.



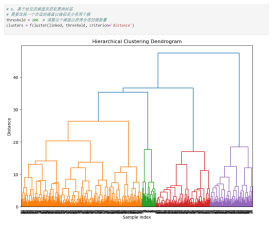
[8]

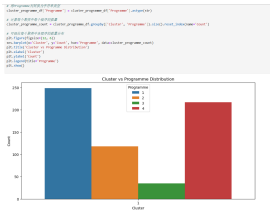
The significance of these efforts is that by using K-means clustering analysis, we are able to uncover hidden patterns and structures in the data set, revealing relationships between features. By associating the clustering results with the item labels, the classification information of the data can be better understood, helping to improve the classification model and make more accurate decisions in practical applications. In addition, the clustering quality index can be used to objectively evaluate the clustering effect and ensure the reliability of the model. The final visual display helps to intuitively understand the clustering results and item label distribution, and provides strong support for subsequent data analysis and decision making.

III. HIERARCHICAL CLUSTERING

I then performed cluster analysis on the data using hierarchical clustering methods to explore the relationships between features and map the results to actual item labels. First, I use Ward method and

Euclidean distance to perform hierarchical clustering on the data. The hierarchical clustering model is constructed by calculating the link matrix. Based on this model, a tree diagram like Figure [9] is drawn to show the hierarchical clustering results, and an appropriate threshold is selected to determine the clustering label. Then I obtain the cluster label of each sample according to the selected threshold value, and associate it with the actual item label to determine the quantity distribution of each item label in each cluster. Moreover, the quantity distribution of each item label in each cluster is analyzed and visually displayed, as shown in Figure [10]. The result shows that, like the previous two models, it has almost nothing to do with Programme.

[9] 

[10] 

The significance of these efforts is that by using hierarchical clustering analysis, we are able to uncover hidden patterns and structures in the data set, revealing relationships between features. Hierarchical clustering provides an intuitive way to understand the hierarchical structure and similarity relationships of data. By associating the clustering results with the item labels, the classification information of the data can be better understood, helping to improve the classification model and make more accurate decisions in practical applications.

IV. Conclusion

By using the Gaussian mixture model (GMM), K-Means, and Hierarchical Clustering for data clustering analysis, I delve into the relationships between features and map the results to actual item labels. In Gaussian mixture model analysis, we selected the optimal number of model components by AIC, BIC and logarithmic likelihood assessment, and found that the correlation between the samples in the training set and the item label was low. When K-means method is used for clustering, the clustering quality is evaluated by contour coefficient, Calinski-Harabasz index and Davies-Bouldin index, and the mapping relationship with item labels is found in the clustering results, but the results show that most clusters are not strongly correlated with item labels. In the hierarchical cluster analysis, I used Ward method and Euclidean distance to build a hierarchical cluster model, visualized the clustering results through the tree graph, and found that most clusters still had a low correlation with item labels after selecting appropriate thresholds.

Although the correlation between the clustering results of each method and the item labels is not strong, through these analyses, I revealed the hidden patterns and structures in the data set, and improved the understanding of the data classification information. These analysis methods not only enhance the ability of data interpretation, but also provide valuable reference for improving classification models, which helps to make more accurate decisions in practical applications. The final visual display enables us to intuitively understand the distribution of clustering results and item labels, which provides strong support for subsequent data analysis and decision making