Exploring and Evaluating Classification Algorithms for Predicting Optimal Anesthesia Methods

Abstract:

The purpose of this study is to design and compare multiple classification algorithms to classify patients into the categories they belong to.

The dataset contains more than 5000 patients and is preprocessed by removing outliers and normalizing features. In the experiment, principal component analysis (PCA) was used for dimensionality reduction, and four supervised classification models of support vector machine (SVM), logistic regression, decision tree, and k-nearest neighbor (KNN) were trained and evaluated, and then the test set was used to evaluate each the accuracy of a model. In addition, the K-means algorithm is used for unsupervised learning training. Experimental results show that the logistic regression classifier has the highest classification accuracy in supervised learning training. In unsupervised learning, the author uses the elbow method and the silhouette coefficient method to determine the optimal number of clusters for K-means clustering.

I. INTRODUCTION

In the field of anesthesia, providing personalized and optimized medical services is a crucial development direction. The choice of drugs and anesthetic methods should be based on each patient's individual circumstances and specific treatment goals. To meet this need, a questionnaire consisting of 15 questions was used to assess the physical condition of the patient, aiming to determine the most appropriate method of anesthesia based on the scores obtained.

The dataset used in this study contains over 5000 patients with two primary outcomes labeled "0" and "1" in the spreadsheet. The classification task is to train different classifiers to predict the anesthesia method category using the questionnaire scores as features. Four classification algorithms were used in the experiments: support vector machine (SVM), logistic regression, decision tree, and k-nearest Neighbor (KNN). The authors evaluate the performance of each algorithm using accuracy scores on separate test datasets.

In addition, this experiment uses k-means algorithm for unsupervised learning, which divides data samples into different clusters and provides effective tools for data classification, grouping and feature extraction. In unsupervised learning, the experiment uses the elbow method and the optimal clustering method to determine the optimal number of clusters for the K-means algorithm. The results of cluster analysis reveal the intrinsic structure of the data, which can further guide the development of classification models.

This study focuses on the evaluation of multiple classification algorithms to classify patients into appropriate anesthesia method categories based on their questionnaire responses.

II. EXPERIMENTAL METHOD

Data observation and processing

As mentioned in the introduction, the Data used for the experiment in this paper is a CSV file named “Data” given by the task requirements, which contains the answers to the questionnaire of all 5344 patients. The first row, first column, and last column of the data describe the problem label, patient label, and outcome label. The authors first classified and counted patients with different labels and visualized the above data. (Fig.1)

It can be seen from the figure that most of the data labels are composed of “0” and “1”, and only a very small number of “2” (14 in total). In the experiment, the author chose to delete the data labeled “2” as an outlier to ensure the accuracy of the final result. Only the data with two labels “0” and “1” were retained. The above data are used in the following research analysis. After removing the data with label 2, the classification task becomes a binary classification.

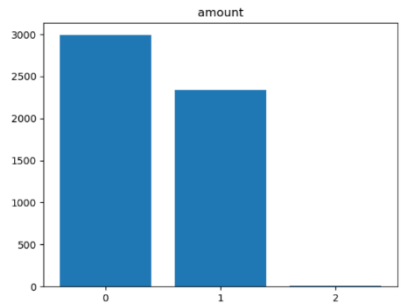

Fig. 1: Visualization of the number of tags

Then the experiment analyzed whether the data belonged to the Gaussian distribution, and the author made a probability density function curve of the Gaussian distribution based on the processed data. (Fig.2)

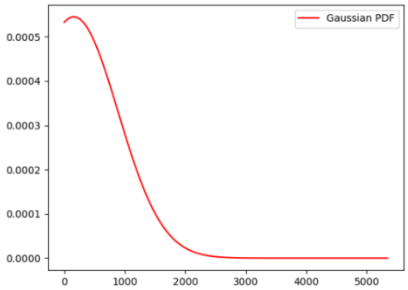

Fig. 2: Probability density function curve

From Fig.2, we can find that the data as a whole presents the shape of the second half of a normal distribution. However, it is not accurate enough to judge whether the data conforms to the Gaussian distribution based on the shape of the Gaussian curve. Therefore, the author decided to use the Shapiro-Wilk test for further normality testing. The test results show that the p-value is less than the significance level (0.05). Therefore, the data does not obey the normal distribution.

Dimensionality reduction

In the dimensionality reduction task, due to the large scale of data, this experiment chooses PCA as the means of dimensionality reduction. PCA dimensionality reduction can reduce high-dimensional data to lower dimensions, thereby reducing storage and computing costs, and is more effective when dealing with large-scale data. Moreover, the data after dimensionality reduction by PCA can be visualized more conveniently, so as to better understand the distribution and characteristics of the data.

At the beginning of the experiment, since the specific dimension reduction is not clear, the authors decided to try to reduce the data to 2 and 3 dimensions after a simple normalization of the data, and use the corresponding Python packages to achieve a simple data visualization in two and three dimensions. (Fig.3 & Fig.4) The aim of this step is to get an intuition of how the data is distributed to the data at the 2D and 3D levels.

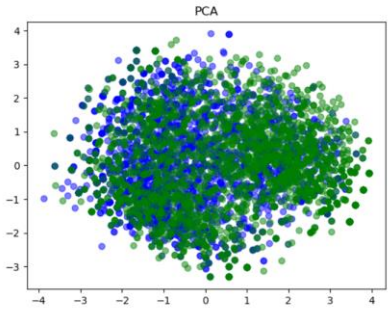

Fig. 3: Visualization of the data in 2D

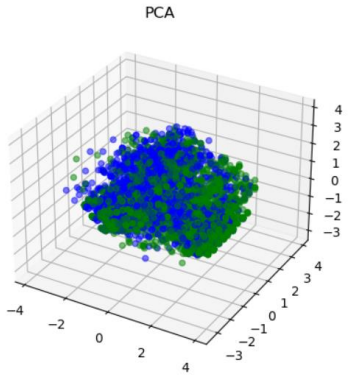

Fig. 4: Visualization of the data in 3D

From the above visualization, the separation of the data at the 2D and 3D levels is low, indicating that the feature dimension after dimensionality reduction cannot capture enough changes and differences in the data. Dimensionality reduction may lose some of the distinguishing feature information in the original data, resulting in overlapping samples in the dimensionality reduction space. So it is not possible to simply reduce the data directly to these two dimensions.

Next, in order to retain more information in the dimensionality reduction, this paper calculated and plotted the explanatory variance of each principal component, and then summed the variance of the principal components to determine the percentage of retained information and the appropriate dimension for dimensionality reduction.

For the experiment, the authors created a chart where the horizontal axis is each principal component of the data and the vertical axis is the explained variance of each principal component. (Fig.5) Within the chart, the bars represent the explained variance of the current principal component, and the stepped lines represent the cumulative explained variance of the first principal component up to the current principal component. That is, the amount of information retained.

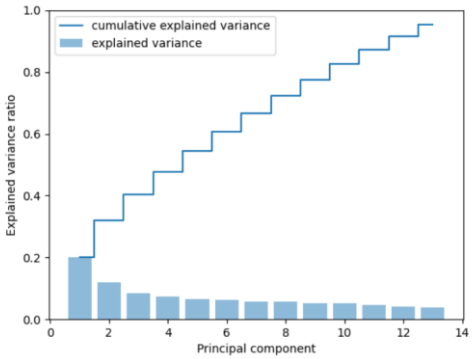

Fig. 5: Principal Components Explanatory Variance Plot

From the explanatory variance table and the sum of the data, the researcher can notice that to retain more than 95% of the information, the author needs to choose a higher dimension, such as 13D. Therefore, in the first round of

experiments, this paper selected 13 dimensions as the target dimension to reduce the dimensionality of the original data. In the process of reducing 15-dimensional data to 13-dimensional data, we need to drop two columns from the original data. The authors chose to create a line plot of eigenvalues to decide which two principal components to leave out. (Fig.6)

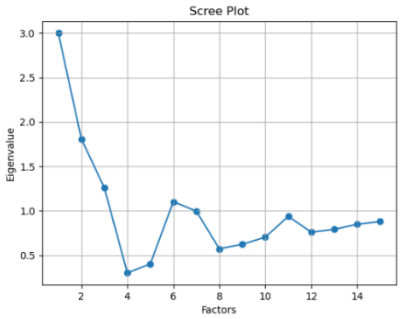

Fig. 6: Eigenvalue Line Chart

In general, the larger the eigenvalues are, the more information is contained in the corresponding principal components. Principal components with small eigenvalues have less influence on the data. By selecting principal components with larger eigenvalues, the data can be mapped into a lower dimensional space while retaining a higher amount of information. Based on this theory, the PCA discards the two principal components with the smallest eigenvalues, the fourth and fifth columns.

Training Classifiers

After reducing the dimensionality of the original data to 13, the author removed the patient numbers from the reduced data in order to separate the feature data from the label data for the unnumbered data.

When dividing the training and test sets, this research used a "training set : test set = 7 : 3" partition. The training set was put into four classifiers: support vector machine (SVM), logistic regression, decision tree, and k-nearest Neighbor (KNN).

a) Support Vector Machine

SVM is a supervised learning algorithm that is mainly used for binary classification problems, but can also be extended to multi-class classification. The basic principle is to separate samples from different classes by finding an optimal hyperplane.

In SVM, samples are represented as points in the feature vector space, and an attempt is made to find a separating hyperplane such that the points of different classes are maximally separated. The key idea is to select the best-separating hyperplane by maximizing the margin. Margin refers to the distance from the hyperplane to the nearest training sample point, and the goal of maximizing the margin is to improve the generalization ability and robustness of the model.

b) Logistic Regression

Logistic regression is a machine learning algorithm used to solve binary classification problems. The basic principle is to build a logistic regression model by fitting the sample data, which can classify new samples.

Logistic regression uses a logistic function (also known as the sigmoid function) to map the results of a linear regression to a probability value (between 0 and 1) for classification.

c) Decision Tree

The basic principle of decision trees is to partition the data based on the features and build a tree based on the partition result. In a decision tree, each internal node represents a condition for a feature, and each leaf node represents a class or a prediction.

The decision tree construction process starts from the root node and selects a best partition feature to partition the dataset into different subsets. The selection of the best split feature is usually based on some kind of evaluation metric (such as the Gini index) that aims to maximize the purity or reduce the uncertainty of each subset. Then, for each subset, the partitioning process is repeated until a stopping criterion is met, such as the maximum tree depth is reached, the number of samples in a node reaches a threshold, or no more features are available.

d) K-Nearest Neighbors

KNN (K-Nearest Neighbors) is a basic supervised learning algorithm, often used in classification and regression problems. The principle is that similar samples will be close to each other in the feature space. KNN first calculates the distance between the sample to be classified and each sample in the training set according to a given distance measurement method (such as Euclidean distance, Manhattan distance, etc.). Then according to the calculated distance, select the K samples closest to the samples to be classified as their nearest neighbors.

For classification problems, by counting the number of occurrences of each category in the nearest neighbor, the category with the highest frequency of occurrence is selected as the predicted category of the sample to be classified. For regression problems, the output values of the nearest neighbors can be averaged as the predicted value of the sample to be classified.

The above four classifiers have their own advantages and disadvantages: SVM works well in high-dimensional space and can handle high-dimensional feature data, but noise and overlap may reduce its performance; Logistic Regression has high computational efficiency and fast training speed and it is suitable for solving binary classification problems, but it is more affected by feature correlation; Decision Tree can automatically handle feature selection, but it is prone to overfitting; KNN is suitable for multi-classification and binary classification problems, but it may make mistakes for a few categories for unbalanced data sets Classification. The dimension of the experiment belongs to the higher dimension, and the classification is a binary classification problem after removing the outliers. Based on the advantages of the above four classifiers, the author believes that these four classifiers are more suitable for this data, so they are selected.

The experiment uses the data after PCA dimension reduction and bias removal through Random Forest as the training data input classifier. The author calculated the

accuracy of the four classifiers based on the predicted values and the test set. The test results are as follows (keep three decimal places):

Tab 1. The accuracy of the four classifiers in 13D

Classifier	Accuracy
SVM	0.685
Logistic Regression	0.704
Decision Tree	0.640
KNN	0.664

After the above comparison, the logistic regression classifier has the highest accuracy among the four classifiers.

There may be several reasons for this result:

1. As mentioned before, the classification problem is a binary classification, and the logistic regression classifier performs well on binary classification.

2. Standardization (StandardScaler) is used in the code to preprocess the data. Logistic regression is sensitive to the scale of features. Standardization can eliminate the scale difference between features, making the model more stable and accurate.

3. It can be seen from Fig.1 that the category distribution of data samples after removing label 2 is relatively balanced, and the number is not much different. Logistic regression can correctly capture the boundaries and relationships between categories

Therefore, of the four classifiers used, the author recommends logistic regression.

Unsupervised Classification

K-means is an unsupervised learning algorithm used for cluster analysis. The main goal is to divide a set of data points into K different clusters such that the data points within the same cluster are similar to each other while the data points between different clusters are quite different. This is the unsupervised learning algorithm used in this experiment.

In this experiment, the elbow method was first used to determine the optimal number of clusters. (Fig.7) When using the Elbow Method to determine the optimal number of clusters, the horizontal axis usually represents the number of clusters, and the vertical axis represents the clustering error or the total internal sum of squares (Within-Cluster Sum of Squares, WCSS). WCSS refers to the sum of squares of distances between all samples in each cluster and the centroid of that cluster. The number of clusters corresponding to the elbow is considered to be the optimal number of clusters, because after this point, the contribution of increasing the number of clusters to reduce WCSS is relatively gradual.

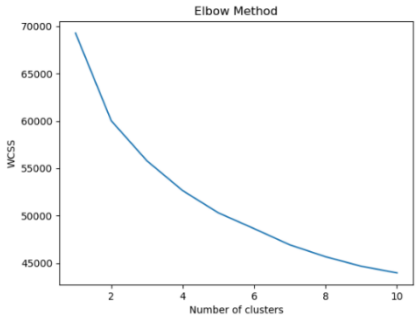

Fig. 7: Elbow Method to find the best cluster number

In Fig.7, we can find a relatively obvious inflection point when the number of clusters is 2. But after careful observation, we can also find that there are small inflection points at other clustering numbers. In order to make the experiment more rigorous, it is therefore not intended to determine the optimal number of clusters by the elbow method only.

Therefore, this experiment decided to determine the best clustering again by the silhouette coefficient method. (Fig.8) This method evaluates the clustering quality by calculating the silhouette coefficient for each data point and then calculating the average silhouette coefficient. The silhouette coefficient takes into account how similar a point is to its cluster and how different it is from other clusters and has values between -1 and 1. The best number of clusters corresponds to the clustering result with the largest average silhouette coefficient.

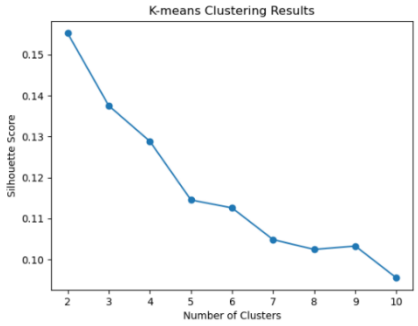

Fig. 8: Silhouette Coefficient in 13D to find the best K

From the above figure, we can find that when the cluster is 2, its silhouette coefficient is the highest, so the best cluster is 2. When the cluster is 2, the silhouette score is 0.157 (keep three decimal places). Finally, visualizing the clustering results under this condition, we can get the following plot. (Fig.9)

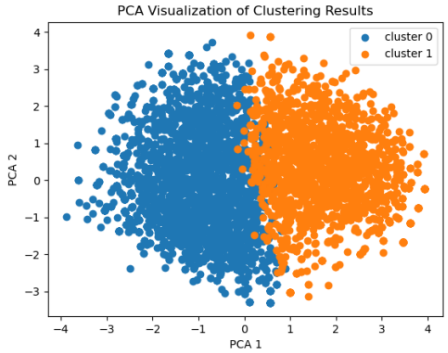

Fig. 9: Visualization of clustering results at the 13D level

From the above figure, we can see that when the dimension is equal to 13 and the best clustering is equal to 2, The two clusters are clearly divided by the middle line but there is still a significant overlap at the junction of the two clusters in the visual clustering result. This may mean that there are some data points with ambiguous boundaries or ambiguous samples.

III. FURTHER OPTIMIZATIONS OF EXPERIMENTAL RESULTS

The author discussed with the TA in the subsequent lab based on the silhouette coefficient scores in 13 dimensions and the clustering visualizations. On the advice of the TA, the authors tried to reduce the data to 10 dimensions and observe the results in the three tasks in the second round of the experiment.

Classifier cross-validation provides a reliable assessment of the performance of a classifier, helps prevent overfitting, optimizes parameter selection, and takes advantage of the information in the dataset. In the task2, the experiment added the k-fold (k =5) cross-validation of four classifiers based on the original code, and the accuracy of the four classifiers after cross-validation is shown in the following table:

Tab 2. The accuracy of the four classifiers in 10D after k-fold cross-validation

Classifier	Accuracy
SVM	0.706
Logistic Regression	0.730
Decision Tree	0.641
KNN	0.705

In this dimension, the accuracy of SVM, Logistic Regression, and KNN all showed significant improvements compared to accuracy without cross-validation, while decision trees showed almost no change. Logistic regression is still the most accurate of the four classifiers.

In the 10-dimensional condition, the graph of task 3 applying the silhouette coefficient method to determine the best cluster is as follows. (Fig.10)

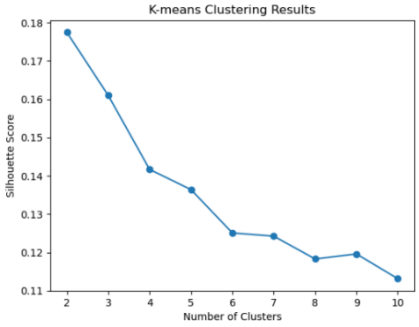

Fig. 10: Silhouette Coefficient in 10D to find the best K

Compared with the first round of the experiment, the optimal number of clusters under the condition of 10D is still 2. But the silhouette score is 0.177 when the best cluster is 2. At the level of the experimental results this time, the author believes that this is a relatively large improvement. The clustering visualization under this condition is shown below. (Fig.11)

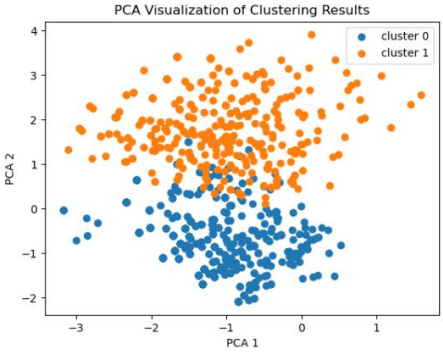

Fig. 11: Visualization of clustering results at the 10D level

Compared with the first round of the experiment, the best silhouette coefficient under the condition of 10 dimensions has a more obvious improvement. There is also a decrease in the overlap of the boundaries of the two clusters in the cluster visualization. The degree of separation between the two clusters is also more pronounced. But correspondingly, this figure has more obvious outliers, which is also the main deficiency of this figure.

IV. EXPERIMENTAL CONCLUSION

Based on the results of the conducted experiments, the following conclusions can be drawn:

1. The reduction of data to 10 dimensions in the second round of experiments produced higher contour scores compared to the reduction of data to 13 dimensions in the first round of experiments. This indicates that the 10-dimensional data representation achieves better clustering quality and inter-cluster separation.

2. Compared to the cluster visualization in the 13-dimensional data, the cluster visualization in the 10-dimensional data has less overlap along the cluster boundaries. This shows that the 10-dimensional data representation leads to sharper boundaries between clusters, indicating better discrimination and more pronounced clustering patterns.

Experimental results show that reducing the dimension of data to 10 dimensions can improve the performance of clustering and enhance the visualization effect of clustering. This means that the 10-dimensional feature space captures the underlying structure and patterns of the data more effectively than a higher-dimensional representation.

For supervised classifiers, cross-validation significantly improved the performance and generalization ability of the classifier, and the logistic regression classifier showed high accuracy in both rounds of experiments.

The above experimental results evaluate and analyze the classification algorithm for predicting the best anesthesia method. This experiment may provide ideas for building anesthesia method classifiers.