Experimental report on the use of artificial intelligence methods to classify patients

Xian Wang (2141698)

Lab-D-Group-2

May 15, 2023

1 Introduction

This coursework consists of three tasks: data dimensionality reduction, construction of classifiers and unsupervised classification of patients, which should be carried out on a dataset of patient questionnaire scores with the aim of providing a comprehensive analysis and assessment of the patient's physical condition. By classifying and clustering the data, we will gain insight into the health status of patients and the influence of relevant factors. This report uses three classifier models (logistic regression, random forest and support vector machine) to classify patients and uses principal component analysis (PCA) to reduce the dimensionality of the input features. In addition, two algorithms were used to cluster the patients. With this report, it is hoped that it will help healthcare professionals to better understand the physical condition of their patients and to determine the appropriate method of anaesthesia for them.

1.1 Data Analysis

The patient questionnaire contains 15 questions that reflect the patient's physical condition. The data set contains the scores obtained by more than 5000 patients for each question and whether their label is '0' or '1'. In addition, a small number of patients have a label of '2'. This data can be removed as noise and can improve accuracy. However, after removing label '2' the categories are left with '0' and '1', which becomes a dichotomous problem, the image becomes less rich and causes the later loss function and other methods will have to be adjusted.

1.2 Classification Selection

Analysis of the data set led to the conclusion that the samples in each process did not conform to a Gaussian distribution and that there was no correlation between the markers for each question in the questionnaire. Considering the relatively large variance of the dataset, a support vector machine classifier (SVM) can be built. Since the dataset after PCA dimensionality reduction is linearly divisible, it is possible to generalize the logistic regression classifier (LR). The random forest based classifier (RF) is an integrated learning algorithm and consists of multiple decision trees that can provide accurate classification results by way of integrated decision trees and robustness to processing PCA reduced dimensional data. The two clustering methods DBSCAN and K-means were compared in the third phase. It is found that the DBSCAN clustering method has more noise, and K-means have a higher silhouette coefficient.

1.3 Classification Result

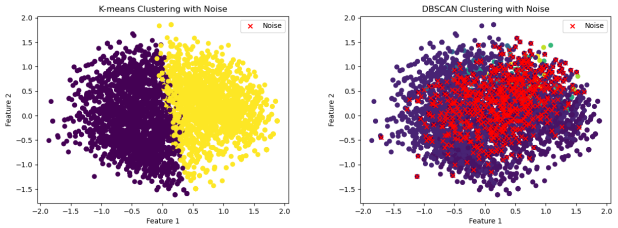
The cross-validation score of the supervised learning classifier is shown below:

Table 1: **Cross-validation Score**

Classifier	Score on Train Set	Score on Test Set
LR	70.21%	71.65%
RF	48.71%	66.71%
SVM	45.47%	67.22%

Figure 1 shows the grouping result of two clustering algorithms.

Figure 1: Clustering result of DBSCAN and K-Means



(a) K-means (b) DBSCAN

The silhouette score of the two algorithms above is shown below:

Table 2: **Silihouette Score**

Algorithm	Silihouette Score	Noise Point
K-means	0.1545	None
DBSCAN	-0.1673	693

2 Dimensionality reduction

2.1 Pre-processing

Data pre-processing was first performed. Because some patients are special and belong to other outcomes, the last column of their dataset has a 'label' of '2', such data can be considered as outliers and can be deleted directly. Apart from that, since the scores between the 15 questions are not correlated, there is no need to delete the columns in this. Also, the 5000+ patients are different people and do not need to be deleted even if they have the same data. Finally, all rows were checked and there was no empty data; if there was empty data, it was to be considered as noise and did not make sense and needed to be deleted. At the same time, we separated the input from the labels.

2.2 PCA

The given pre-processed dataset is then subjected to dimensionality reduction analysis. Data dimensionality reduction plays an important role in the field of machine learning and data analysis, by reducing the dimensionality of features, it can simplify the complexity of the data, reduce storage space and computational costs, and improve the training efficiency of the model. Principal Component Analysis (PCA) was chosen for this report to perform dimensionality reduction, which transforms the original features into a new, uncorrelated set of principal components through a linear transformation. It achieves dimensionality reduction by finding the principal components with the highest variance in the data and projecting the data onto these principal components [1].

2.3 Feature dimension selection

To select the appropriate feature dimension, the principal component explained variance ratio was used to assess the proportion of the variance explained by each principal component to the data. The explained variance ratio indicates how much of the information in the original data is retained by the feature after dimensionality reduction. By plotting the cumulative sum of the principal component variance ratios, it can help to determine the appropriate dimensionality of the feature [2]. The cumulative sum of explained variance ratios curve shows the relationship between the number of principal components and the explained variance ratio. By looking at the shape of the curve, ‘ 8 ‘ was chosen as the principal component because its explained variance ratio was both over 70% and close to 80%, which allowed the study to reduce the dimensionality of the data while retaining most of the variance of the data.

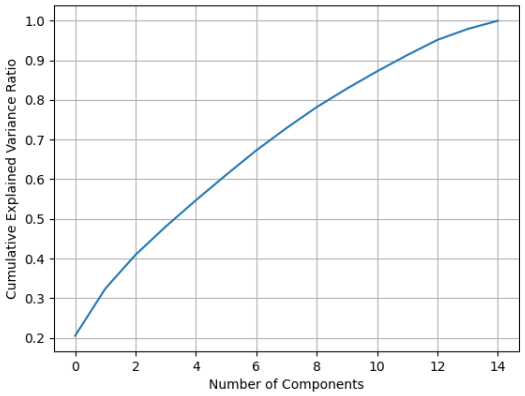


Figure 1: Cumulative Explained Variance Ratio Curve

2.4 Investigating Explained Variance Ratio Bar Chart

The explained variance ratio bar chart shows the explanatory variance ratio for each principal component. The explained variance ratio was observed to decrease as the principal components increased. Those principal components with high explained variance ratios were selected to retain the features that had a significant impact on the data. In addition to this, the rate of decline in the explained variance ratio can be determined by looking at the histogram [3]. For example, the histogram shows a clear decreasing point at PCA features ’1’ and ’2’, indicating that after that point adding more principal components contributes less to explaining the variance ratio. Choosing the appropriate principal components can reduce the dimensionality while retaining information about the data. The histogram of the survey explained variance ratio is shown in Figure 2.

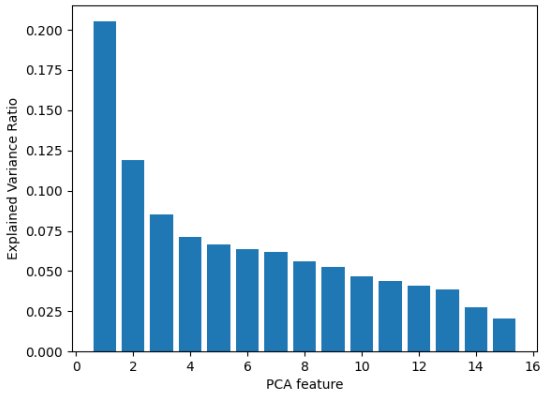


Figure 2: Explained Variance Ratio Bar Chart

2.5 Row Bias Removal

The reduced dimensional data is then subjected to row bias removal. This improves data comparability, with differences between features reflecting more of the true differences rather than those due to overall bias. In addition to this, it reduces redundant information in the dataset, providing cleaner and more accurate data, reducing computational costs and avoiding over-fitting, helping to better understand and model the data [4]. The image after bias removal for the dimensionality reduction data is shown in Figure 3.

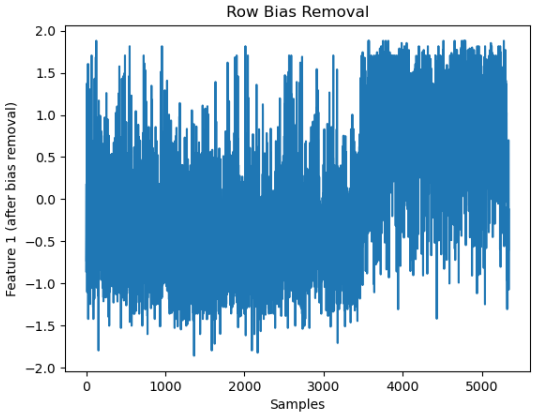


Figure 3: Row Bias Removal

3 Training Classifiers in a Supervised Way

3.1 Classifier selection and principle

1. Logistic regression is based on a linear regression model and is used to predict the class of a sample by mapping a linear combination to a probability value through a logistic transformation [5]. In logistic regression, we assume that there is a linear relationship between the input feature x and the output variable y. This linear relationship can be expressed as follows:

$$z = \theta^T x + b \tag{1}$$

This is where z is the result of a linear combination, θ is the parameter vector, and b is the bias (intercept). To convert the linear combination into probabilities, we use the logistic function (also known as the Sigmoid function):

$$h_\theta(x) = \frac{1}{1 + e^{-z}} \tag{2}$$

Here, $h_\theta(x)$ is the prediction function for the input feature x for classification as a prediction function. The logistic function maps the result of the linear combination to a probability value ranging from 0 to 1.
Parameter estimation is a key part of logistic regression [6]. Maximum likelihood estimation is used to estimate the parameters of the model. Suppose there is an observed data set containing n samples. Each sample has an input feature x_i and a corresponding output label y_i, where i=1,2,...,n. The objective is to find a set of parameters θ that maximises the likelihood function of the observed data set. The likelihood function can be expressed as function 3.

$$L(\theta) = \prod_{i=1}^{n} h_\theta\left(x_i\right)^{y_i} \cdot \left(1 - h_\theta\left(x_i\right)\right)^{(1-y_i)} \tag{3}$$

Usually, the log-likelihood function is used to simplify the calculation. The log-likelihood function is:

$$\ell(\theta) = \sum_{i=1}^{n} \left(y_i \log\left(h_\theta\left(x_i\right)\right) + \left(1 - y_i\right) \log\left(1 - h_\theta\left(x_i\right)\right)\right) \tag{4}$$

The goal is to find the parameter θ that maximizes the log-likelihood function. Optimization algorithms such as the gradient ascent method can be used to maximize the log-likelihood function and thus find the optimal parameter. In the prediction phase, we use the estimated parameter θ to make predictions. For a given input feature x, we compute the probability value of $h_\theta(x)$. Usually, when $h_\theta(x) \geqslant 0.5$, the prediction label is set to 1; when $h_\theta(x) < 0.5$, the prediction label is set to 0.

2. Random Forest is a powerful integrated learning method that consists of multiple decision trees, each constructed from a random subsample of the original training set. This randomness allows RFF to avoid overfitting problems and to improve prediction accuracy and generalization. By sampling the training set without putting back, each tree uses a different sample, increasing the diversity of the model. In addition, each node randomly selects a portion of features in the feature set, giving each decision tree a different selection of features.
A decision tree is a classifier or regressor based on a tree structure. It performs prediction by recursively dividing the data set into subsets. Each decision tree consists of a root node, internal nodes and leaf nodes [7].
Gini impurity is a measure of node impurity used in the decision tree construction process. It measures uncertainty and confounding based on the distribution of sample labels in the dataset. By calculating the probability that two samples are randomly selected in the dataset and their labels are inconsistent, Gini impurity can reflect the degree of impurity within a node. During the construction of the decision tree, selecting the features with the most reduced Gini impurity for segmentation can lead to an increase in the purity of the nodes after segmentation, which in turn improves the classification accuracy. Specifically, the algorithm calculates the

Gini impurity of each feature and selects the feature that reduces the Gini impurity the most as the optimal segmentation feature to divide the data set into different subsets. The following is the formula for Gini impurity:

$$Gini = 1 - \sum_{i=1}^{k} \left(\frac{N_i}{N} \right)^2 \tag{5}$$

3. Support Vector Machine (SVM) is a powerful supervised learning algorithm for binary and multiclassification problems. It separates different classes of samples as much as possible by finding an optimal hyperplane with good generalization ability. In SVM, samples are considered as points in a high-dimensional space, each represented by a feature vector. the goal of SVM is to find a hyperplane that maximizes the classification boundary (or interval) while minimizing the classification error. This hyperplane can be considered as a decision boundary, and classification is performed by determining on which side of the hyperplane the sample lies.

   The decision function of the SVM can be expressed as:

$$f(x) = sign(\mathbf{w}\mathbf{x} + b) \tag{6}$$

   where $\mathbf{w}$ is the normal vector (weight vector) of the hyperplane, $\mathbf{x}$ is the eigenvector of the input sample, b is the bias (intercept), and $f(x)$ is the class of the predicted output.
   In SVM, the support vectors are the nearest training sample points to the hyperplane. They play an important role in the location and spacing of decision boundaries. Predicting the class of a new sample can be achieved by computing its projection on the hyperplane. It can handle high-dimensional feature spaces and nonlinear problems, has less impact on small sample data and outliers, and has better generalization performance.

3.2 Data features selection

The data that were dimensioned down using PCA and removed by row bias were fed into the classifier as training data, and histograms were generated for different numbers of labels. The histogram of the number of labels is shown in Figure 4.

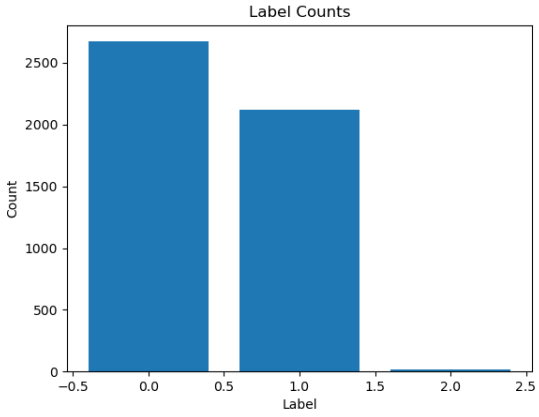


Figure 4: Label Counts

According to the images, there is a clear imbalance in the dataset, where the number of data samples with label '0' far exceeds the number of data samples with label '1' and label '2'. Considering the large difference in the number of samples, undersampling may be a reasonable choice as one of the methods to deal with unbalanced datasets, which can balance the dataset by reducing the number of samples in most categories and can improve the learning of the classifier for a few categories.
However, undersampling may lead to underfitting problems. By reducing the number of samples in most categories, undersampling may lose some important information and sample diversity, thus preventing the model from fully learning the features and patterns of the data. Underfitting means that the model does not fit the training data well, resulting in poor prediction performance. If undersampling is used too heavily, it may cause the model to rely excessively on samples from a few categories, thus failing to capture important information from most categories. This may lead to poor generalization of the model to practical applications and the inability to correctly predict unseen samples.

3.3 Classifier training and Cross-Validation

1. In the logistic regression classifier, the default setting is selected for this report. For example, the regularization type 'penalty' is chosen as 'l2' and the pairwise form 'dual' is set to 'False' because the number of samples is larger than the number of features, which gives better performance. The convergence tolerance 'tol' is set to default '1e-4' to control the stopping tolerance of the algorithm. The inverse of the regularization strength 'C' is the default value '1.0'. In addition to this, the hyperparameters of the logistic regression classifier are

changed. Set it to use 'solver', the optimisation algorithm for 'lbfgs', which works for large data sets and is quite efficient. The maximum number of iterations 'max_iter' is also changed to 1000, which can increase the optimization Chengdu of the model on the training data, improve the accuracy and performance of the model, and also control the training time of the model properly. If the maximum number of iterations is set too high, it may cause the model to overfit the training data and its ability to generalize to new data is reduced.
Grid-CV is used to systematically search for the best combination of model parameters to obtain the best model performance. It evaluates the performance of each parameter combination by searching exhaustively for specified parameter combinations and using cross-validation to finally determine the best combination of parameters. In this paper, we use K-fold cross-validation to evaluate the performance of the classifier by dividing the dataset into five parts, each in turn as the validation set and the rest as the training set. Then, the model is trained and evaluated for each validation set, and data containing the results of each evaluation is returned. In this paper, the average of the F1-score is used as the CV score.
The result of Grid-CV and CV score is shown in Table 3.

Table 3: **CV result of LR**

	Raw data	PCA
C	1.0	2.0
Score on Train Set	70.23%	70.21%
Score on Test Set	71.82%	71.65%

2. For Random Forests, the number of decision trees 'n_estimators' is an important parameter. Larger values can improve the performance of the model, but also increase the computation time.' criterion' is a measure of the quality of node splitting, and optional values include the Gini coefficient (gini) and the information gain (entropy). The maximum depth of the decision tree 'max_depth' is used to control the complexity of the tree and avoid overfitting. Larger values can increase the complexity of the model, but may also lead to overfitting. Besides, the random forest has many hyperparameters, for example, 'n_jobs' is used to specify the number of jobs to run decision tree construction and prediction in parallel, which can speed up the training. As with logistic regression, random forests can be parameter tuned with GridSearchCV.
The result of Grid-CV and CV score is shown in Table 4.

Table 4: **CV result of RF**

	Raw data	PCA
n_estimators	100	500
Score on Train Set	68.58%	48.71%
Score on Test Set	64.98%	66.71%

3. For support vector machines, its parameter setting 'C' is a regularization parameter like logistic regression, which is used to control the degree of penalty for misclassification. The choice of kernel function 'kernel' is used to map the data to a high dimensional space. The optimal parameters are continued to be detected using Grid-CV [8].
The result of Grid-CV and CV score is shown in Table 5.
In summary, the average cross-validation score of LR is higher than that of the other classifiers (71.65%)and therefore the logistic regression classifier would be recommended, presenting the classification results more concretely through the confusion matrix in Figure 5.

Table 5: **CV result of SVM**

	Raw data	PCA
C	1.0	2.0
Score on Train Set	62.80%	45.47%
Score on Test Set	64.20%	67.22%

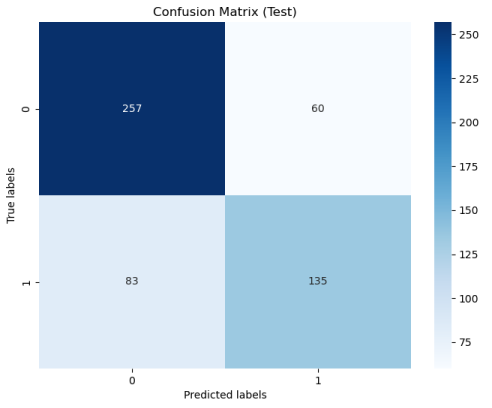


Figure 5: Confusion Matrix of LR

4 Unsupervised Classification

1. In this report, the process of K-means will be explained in detail. Because the density-based clustering algorithm (DBSCAN) has noise and unsatisfactory performance, K-means is used to divide the dataset into K distinct groups or clusters. Each cluster consists of samples within it, while the samples between different clusters have a large variability. It is an iterative optimization algorithm and the core idea is to minimize the sum of squared distances between samples within a cluster to make the samples within the same cluster more similar and the samples between different clusters more different. By iteratively updating the cluster centres and reassigning samples, the K-means algorithm finds the optimal cluster centres and cluster partitioning. However, K-means relies on initial random values, and different initial values may lead to different clustering results, so the algorithm needs to be run several times to select the best result. Besides, it is also sensitive to outliers and noise, and outliers may affect the calculation of clustering centres and lead to wrong clustering results. And noisy points may be incorrectly assigned to a cluster, affecting the accuracy of clustering [9].

2. This report uses the Silhouette coefficient as a clustering assessment metric to measure the quality of the clustering results. It combines intra-cluster tightness and inter-cluster separation and is used to determine the optimal number of clusters [10].
   The following is the formula for the Silhouette coefficient.

$$s_i = \frac{b_i - a_i}{max\{b_i, a_i\}} \tag{7}$$

   The initial range of k is determined as 2 to 6, according to Figure 5, the silhouette coefficient when k=2 is highest(0.1545), so the patients should be divided into 2 groups.

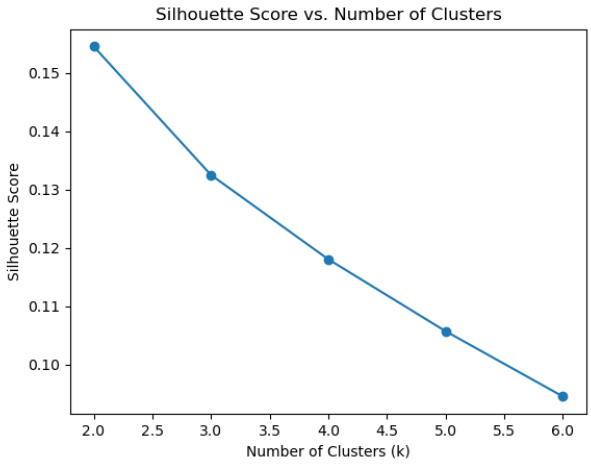


Figure 6: Silhouette coefficient and k

5 Conclusion

5.1 General conclusion

The physical condition of the patient can be seen from the scores of the 15 questions in the questionnaire, and the corresponding 'label' is given according to the score of the question. In this report, the data are used in the training of three classifiers, logistic regression, random forest, and support vector machine, after PCA dimensionality reduction and row bias removal. Among them, logistic regression has the best classification performance, with an average cross-validation f1-score of 71.65%. In the unsupervised learning phase, the number of grouped patients was determined to be 2 based on the Silhouette coefficient.

5.2 Optimization

1. The results of the cross-validation F1-score showed a large difference between the scores of the random forest classifier and the support vector machine classifier on the training and test sets. The poor performance of the training set and the better performance of the test set can be attributed to the following reasons: inconsistent data, underfitting, data quality issues and inappropriate feature representation. Inconsistent data distribution is when the data in the training set does not match the data in the actual application scenario and the model is unable to fully learn the features and patterns of the data during training, but the test set may be closer to the data distribution in the actual application scenario and therefore the model performs better on the test set. Underfitting occurs

because of poor performance on the training set, which can be caused by the model's lack of complexity to capture the complex relationships in the data and to effectively fit the data in the training set. Attempts should be made to increase the complexity of the model in the experiment, for example by increasing the number of layers, the number of parameters or the number of features in the model. But this in turn leads to an increase in time. The second issue is data quality, where the training set has a large number of noisy or mislabelled samples, whereas the test set is composed of cleaner as well as more accurately labelled samples, resulting in a large difference in performance. Finally, the feature representation in the training set may not be sufficient to capture the important information in the data, the test set uses a better feature representation and in future experiments, better feature engineering should be sought to find a feature representation that better represents the characteristics of the data.

2. The random forest and support vector machine classifier training sets differed significantly from logistic regression on the cross-validated F1-score. There may be issues with linear separability of data, data noise and complexity, sample size and data distribution. Logistic regression is a linear model and has better performance for linearly divisible datasets. The samples in this training set can be distinguished from the different categories by a linear boundary in the feature space, and logistic regression fits this relationship better. However random forests and support vector machines are non-linear models, so the scores will be lower. There is a lot of noise in the training set or the relationship between the data is more complex, and random forests and support vector machines may receive the effects of these factors. Logistic regression, on the other hand, is a simple model that is easier to fit to noisy and less complex data sets, resulting in a higher score. In addition to this, random forest and support vector machines can suffer from a small sample size in the training set or a mismatch between the distribution of the data and the distribution assumed by the model, resulting in lower performance. In contrast, logistic regression is more robust to smaller sample sizes and simple data distributions. These are issues that can be explored in depth, as well as finding ways to optimise them.

References

[1] Ian T. Jolliffe. Principal component analysis. In *International Encyclopedia of Statistical Science*, 2002.

[2] Huan Liu and Hiroshi Motoda. Feature selection for knowledge discovery and data mining. In *The Springer International Series in Engineering and Computer Science*, 1998.

[3] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:498–520, 1933.

[4] David Roi Hardoon, Sándor Szedmák, and John Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 16:2639–2664, 2004.

[5] David W. Hosmer and Stanley Lemeshow. Applied logistic regression. 1991.

[6] D. R. Cox. The regression analysis of binary sequences. *Journal of the royal statistical society series b-methodological*, 20:215–232, 1958.

[7] L. Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

[8] Corinna Cortes and Vladimir Naumovich Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.

[9] J. MacQueen. Some methods for classification and analysis of multivariate observations. 1967.

[10] Peter J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.