Machine Learning Project: Student Achievement Classification Project Database

Songming.PING (2033685)

Lab-2-Group-7

April 26, 2022

Machine Learning Project: Student Achievement Classification Project Database

Author:Songming.PING
Student number:2033685

Abstract: In this paper, feature engineering has been carried out to optimise the dataset features for the data characteristics of student achievement and classification datasets. The open set problem is effectively solved by the idea of binary classification. Three supervised learning algorithms are optimised to obtain higher accuracy results and to evaluate the final model. The data-level unsupervised models are evaluated based on contour coefficients to obtain optimal clustering result.

This experiment was done using python code.

1 Introduction

Targeted education based on student performance is widely used as one of the methods to improve the quality of teaching and learning.

It is costly and inaccurate for teachers to classify students based on their report cards one by one. The development of machine learning algorithms can be an effective solution to this problem. This report considers data characteristics and dimensionality reduction through PCA,NMF and other methods. The experimental process focuses on the use of binary classification ideas to solve the open set problem. The logistic regression classifier is optimised by several binary classification iterations, which effectively improves the model accuracy. Unsupervised learning and outlier removal based on the distance from the cluster centre were performed. A practically applicable xun was derived.

2 Data Observation

Feature engineering is the process of using specialist background knowledge and skills to process data so that features can be used to better effect on machine learning algorithms.

Significance: can directly affect the effectiveness of machine learning.

2.1 Feature Extraction

Convert arbitrary data (e.g. text or images) into digital features that can be used for machine learning.

The first step is to import the data, naming the imported file "CW_Data".

Read .csv file, comma separated, read all contents.

Through the import process above, it was analysed that the data was presented as numpy arrays. It contains "ID", "Programme" and five features corresponding to student scores, with the last row being a null row. There are outliers with different classifications for the same feature and "Programme=0" which is outside the normal range. These issues are dealt with below.

The existence of "Programme=0" will be considered and resolved in a subsequent open set.

2.2 Feature pre-processing

The process of converting feature data into feature data that is more suitable for the algorithmic model by means of some conversion functions.

In this study, the student's id number did not affect the classification results and was an irrelevant feature that needed to be removed.

Data with the same features but different classification results will be considered outliers that affect the study and should be removed.

Null values in the dataset will cause training anomalies and need to be removed. The basic characteristics of the data are displayed after pre-processing, including the mean, variance, etc.

In the code, the "isnull" method specifies the location of the null value, the "dorpno" method removes the null value, the "drop_duplicates" method removes the outlier, the "dorp" method removes the column and the "describe" method obtains the basic characteristics of the data.

Before deleting	515 data items
After deletion	439 data items

Table1: Comparison of before and after outlier removal

Save the processed data as "deleteerror.csv"

For better analysis, the data are dimensionless (normalised, normalised).

Considering that features vary greatly in units or size, or that the variance of a feature is several orders of magnitude larger compared to others, it is easy to influence (dominate) the target result, making some algorithms unable to learn other features. Therefore, some methods are needed to dimensionlessly transform data of different sizes to the same size.

Normalisation is first performed, mapping the data between (default [0,1]) by applying a transformation to the original data.

$$X' = \frac{x - min}{max - min} \qquad X'' = X' * (mx - mi) + mi$$

Standardisation of the data is also required. The data is transformed to a mean of 0 and a standard deviation of 1 by transforming the original data.

2.3 Feature Dimensionality Reduction

Dimensionality reduction is the process of reducing the number of random variables (features) to obtain a set of "uncorrelated" primary variables under certain constraints

Reducing the number of random variables.

This is because in training, features are used for learning. If there are problems with the features themselves or if the features are highly correlated, this will have a greater impact on the algorithm's ability to learn predictions and therefore requires dimensionality reduction.

2.3.1 Low variance feature filtering

Considering the small variance of the features, a feature is mostly close to the sample value and the data is first filtered with low variance features.

In the above code, initialise VarianceThreshold = 10,specify the threshold variance and derive Q2 as a low variance feature and delete it.

2.3.2 Spielman correlation coefficient

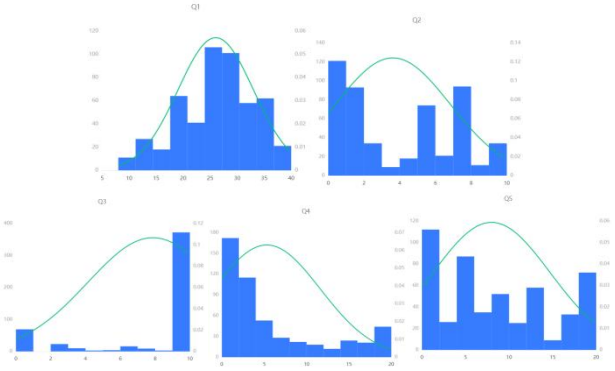


Figure1: Normal calibration chart

By testing the histograms for normality, all data normality plots do not show a bell shape, indicating that the data do not obey a normal distribution, so instead of using Pearson's

correlation coefficient, Spearman's correlation coefficient is used.

The Spearman coefficient evaluates the correlation of two statistical variables using a monotonic equation.

$$\rho = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}}.$$

The first test is whether there is a statistically significant relationship between XY; then the positive and negative correlation coefficients and the degree of correlation are analysed.

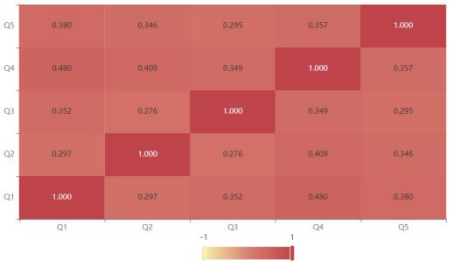


Figure2: Correlation coefficient heat map

Analysis of the heat map shows that Q1 is relatively highly correlated with Q4 and Q2 with Q4, with a correlation coefficient greater than 0.4, but the correlation coefficient is still less than 0.5, which is not a highly correlated feature, so the corresponding features are not deleted or combined.

2.3.3 PCA

The KMO and Bartlett's tests were first performed to determine if principal component analysis could be performed. For the KMO value: on 0.8 is very suitable for principal component analysis, between 0.7-0.8 is generally suitable, between 0.6-0.7 is less suitable, between 0.5-0.6 means poor, under 0.5 means extremely unsuitable, for Bartlett's test ($p < 0.05$,strictly speaking $p < 0.01$), if the significance is less than 0.05 or 0.01 and the original hypothesis is rejected, it indicates that principal component analysis can be done, if the original hypothesis is not rejected, then these variables may provide some information independently and are not suitable for principal component analysis;

KMO test and Bartlett's test		
KMO values		0.758
Bartlett's test of sphericity	Approximate cardinality	300.389
	df	10.000
	p	0.000***

Note: ***, **, * represent 1%, 5%, 10% level of significance respectively

Table 2: KMO test and Bartlett's test

The results of the KMO test showed a value of 0.758, while the results of the Bartlett's spherical test showed a significant p-value of 0.000***, a correlation between the variables and a valid principal component analysis of average magnitude.

The PCA analysis starts by evaluating the original variables and calculating the variance of each variable and the correlation between the two variables to obtain a covariance matrix. In this covariance matrix, the diagonal values are the variances of each variable and the other values are the respective covariances of the two variables. Then, its eigenvalues and eigenvectors are calculated. The product of the original variables and the eigenvectors (a linear combination of the original variables) is the new variable; the covariance matrix of the new variable is the diagonal covariance matrix, with the variances on the diagonal sorted from largest to smallest; the top two or three new variables with the greatest information, i.e. the variance, are then selected from the new variables to visualise the principal components[1] .

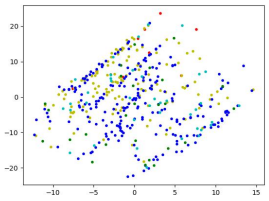


Figure 3: PCA downscaled to 2-dimensional scatter plot
PCA downscaled to 2-dimensional scatter plot.

After an initial PCA downscaling was performed, the features of at least 80 percent of the original data were considered in the experiment, in which case the original data needed to be downscaled to a 3-dimensional case and visualised in 3D.

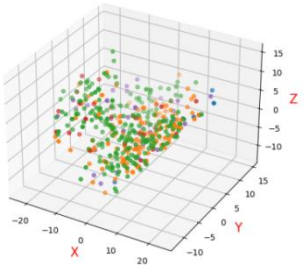


Figure 4: PCA retention 80% feature scatter plot (3D)

component	Explanation of variance	Cumulative variance explained	Weighting
1	0.442	0.442	58.747%
2	0.161	0.603	21.378%
3	0.149	0.752	19.875%

Table3: Results of principal component weights

The results of the weight calculation of the principal component analysis showed that the weight of principal component 1 was 58.747%, the weight of principal component 2 was 21.378% and the weight of principal component 3 was 19.875%, with the maximum value of the indicator weight being principal component 1 (58.747%) and the minimum value being principal component 3 (19.875%).

Deviation values are removed with the help of a clustering algorithm.

The distance between each data point and its nearest clustering centre is calculated. The largest distance is considered to be an outlier.

outliers_fraction is set to 1% because in the case of a standard normal distribution (N(0,1)), research generally considers data with more than 3 standard deviations to be outliers, and data within 3 standard deviations contains more than 99% of the data in the dataset, so the remaining 1% can be considered as outliers.

The number of outliers is calculated from the outliers_fraction number_of_outliers. Set a threshold to determine the outliers. Use the threshold to determine if the data is an outlier. Visualisation of the data (both normal and abnormal data).

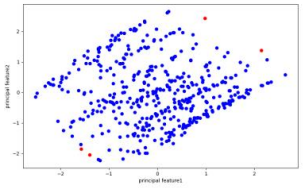


Figure 5: Anomalous features after PCA dimensionality reduction and clustering process

The red dots in the graph above are the identified outliers, which represent approximately 1% of the total data volume.These four outliers can be removed.

After obtaining the dimensionality reduction structure of the PCA, the scatter plot of the Cogito words reveals that the partitioning of the data is not particularly clear, so other dimensionality reduction methods need to be tried in the experiment in order to find more alternate and more representative features for subsequent training.

2.3.4 NMF:

Considering that all student scores are non-negative and that NMF has a better explanation of the local characteristics of things, an attempt was made to reduce the dimensionality to 2 dimensions using NMF.

The basic idea of NMF can be simply described as follows: for any given non-negative matrix A, the NMF algorithm is able to find a non-negative matrix U and a non-negative matrix V such that satisfy , thus decomposing a non-negative matrix into the product of two non-negative matrices on the left and right.

Through matrix decomposition, the dimensionality of the matrix describing the problem is reduced on the one hand, and on the other hand a large amount of data can be compressed and generalised[2].

$$argmin1/2||V - WH||^2 = \sum_{ij}(v_ij - wh_ij)^2$$

In the experiment it is based on decomposing the large matrix into two small matrices such that the two small matrices can be reduced to the large matrix when multiplied together.

In this experiment for the NMF implementation, the number of iterations is specified as 500, the gradient descent constant is usually taken to be smaller, here 0.0002, and the learning rate is 0.02.

The core code for matrix decomposition is

```
P[i][k] = P[i][k] + alpha * (2 * eij * Q[k][j] - beta * P[i][k])
Q[k][j] = Q[k][j] + alpha * (2 * eij * P[i][k] - beta * Q[k][j])
```

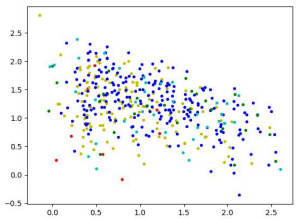


Figure 6: NMF downscaled 2-dimensional scatter plot

As can be seen from the results above, NMF obtained relatively good results and a visual scatter plot, successfully obtaining a set of 2-dimensional features.

2.3.5 Isomap:

As the PCA results were not particularly satisfactory and the MDS results were not good, an attempt was made to use isometric feature mapping for dimensionality reduction. Research assume that the data are not distributed in a traditional Euclidean space, but are embedded on a potential manifold in an outer dimensional Euclidean space, or that these data points can form such a potential manifold.

The principle is essentially the same as for MDS, but in isomap the distance between two points is the shortest path of two points on the way, and the derivation is in the form of an inner product[3].

The algorithm flow is

(1) Set the number of nearest neighbour points k at each point, construct the connectivity graph and the adjacency matrix.

(2) Construct the distance matrix in the original space by the shortest path of the graph.

(3)Calculate the inner product matrix .

(4) Decompose the matrix B into eigenvalues to obtain the eigenvalue matrix and the eigenvector matrix.

(5) Take the first [Eq.] term of the largest eigenvalue matrix and its corresponding eigenvector .

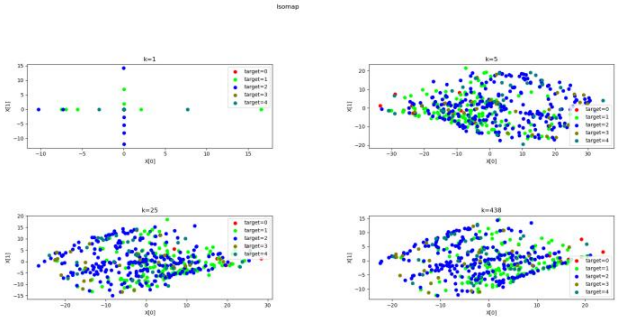


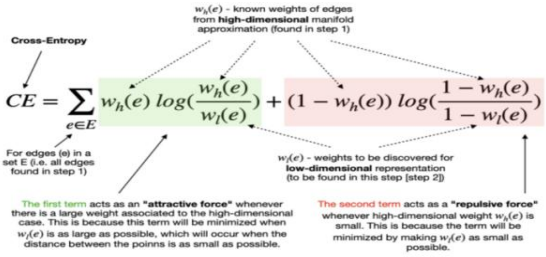
Figure 7: Scatterplot of Isomap dimensionality reduction results for different k values

In this experiment, the hyperparameters n_neighbors=1, 5, 25, and size-1 were chosen for testing and visualisation respectively, and clearer results were obtained, but there were still major flaws from which the dataset could be largely judged to be inadequate. As can be seen from the above figure, the dimensionality reduction has successfully obtained two-dimensional data.

2.3.6 UMAP:

UMAP assumes that the available data samples are uniformly distributed in the topological space and can be approximated and mapped from these finite data samples to a lower dimensional space.

UMAP first uses the Nearest Neighbour-Bailey algorithm to find the nearest neighbours. The hyperparameter n_neighbors to refers to the number of nearest neighbours and can be used to specify how many nearest neighbours are used. distances in UMAP are standard Euclidean distances relative to the global coordinate system. The conversion from variable to standard distances also affects the distances of the nearest neighbours. Therefore, another hyperparameter min_dist (default value = 0.1) is used to define the minimum distance between the embedding points. With the minimum distance specified, the algorithm can begin to find a better representation of the low-dimensional manifold.UMAP achieves this by minimising the following cost function, also known as cross-entropy (CE) [4].



The ultimate goal is to find the optimal weights of the edges in a low-dimensional representation.

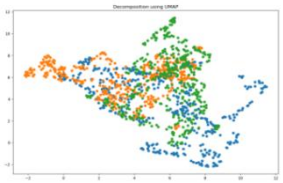


Figure 8: Decomposition using UMAP

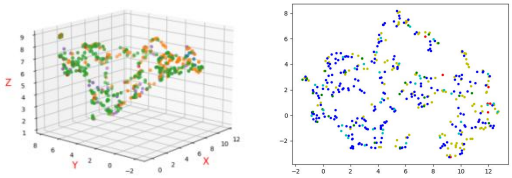


Figure 9 : Scatter plot of UMAP dimensionality reduction results (3D,2D)

The results show that the original data is not evenly distributed in the topological space and the method is excluded

2.4 Data bias(Isolated forests)

From the above analysis, it is concluded that this experimental dataset is chaotic and scattered, and therefore the unsupervised learning algorithm of isolated forest is used for anomaly detection.

In an isolated forest, the dataset is recursively and randomly segmented until all sample points are isolated. With this random segmentation strategy, it usually takes very few segments to make the anomalies isolated. Clusters with very high densities are required to be cut many times to be isolated, but those points with very low densities can be isolated easily.

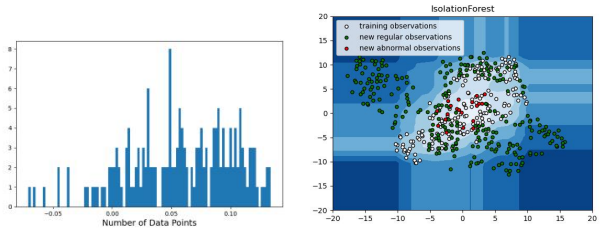


Figure 10: Data shortest path and isolated forest outlier reality

From the analysis of the graph on the left, the shortest paths for most of the data are long, but there are some individual data with very short paths (e.g. less than -0.08), so in this experiment the data with shorter paths are considered to be anomalous data, and it is assumed that those with path lengths less than -0.08 are considered to be anomalous data.

Visualisation of the training set data, the test set data and the anomalies is carried out through the post-training model (right panel). Where red is the outlier, white is the training set and green is the test data.

The above analysis resulted in good model training results and accurate differentiation of anomalous data. The anomalous data was obtained and the anomalous data was removed to improve the feature accuracy.

2.5 Feature selection

After the above investigation, it is concluded that for this dataset, PCA dimensionality reduction and NMF dimensionality reduction present relatively good results, so the 3-dimensional features obtained by PCA dimensionality reduction and the 2-dimensional features obtained by NMF dimensionality reduction are selected to form a new 5-dimensional feature, forming the new dataset "finaldata". The new dataset is then used together with the original dataset for subsequent supervised training.

3 Training Classifiers in a Supervised Way

3.1 Dividing the data set

In machine learning, the data set is generally divided into two parts: training data, and test data. In this study, the dataset is divided using sklearn's train_test_split.

For better subsequent use of the dataset, the data to be used in the study was converted into the dataset common storage form bunch by means of the Bunch package.

In the train_test_split method, upload the 'feature' and 'target' of the data, control the test_size to 0.6 and the random number seed to 22 in this experiment.

3.2 Open set problems

Before the model can be trained, the open set problem involved in this dataset first needs to be solved.

According to the requirements, this is an open-set classification problem due to the existence of the "0" category, which contains not only normal categories from 1 to 4, but also other unknown categories such as 0. However, these unknown categories are not specifically labelled, and the classifier cannot know the specific categories of these unknown categories according to the secular data, which together constitute one category: the "0" category.

The set S contains N finite categories with specific labels and S contains K finite or infinite unknown categories, the open set classification problem is to divide these N categories and reject these K unknown categories.

$$\text{openness} = 1 - \sqrt{\frac{2 \times |\text{training classes}|}{|\text{testing classes}| + |\text{target classes}|}}$$

Based on the above problem, the support vector machine was chosen for the open set problem, as support vector machines define the half space and classify data from any training sample. This results in dividing the half-plane so that the training samples are on one side and the non-training samples on the other.

The data labels were first changed to group 1-4 into one class and class 0 into a separate class, which was fed into the support vector machine for training[6].

During training, the C value is set to 2, the kernel function is 'rbf'(Linear kernels), the gamma parameter is 10, and the strategy is set to a one-to-many strategy for training.

Subsequent further solutions to the open set problem require algorithmic optimisation to address empirical risk as well as open set risk, and are not considered in this experiment.

The open set identification problem is defined as minimising both empirical and open set risk, while this experiment focuses on the SVM method to optimise its open set risk with an f-score as the objective.

	precision	recall	accuracy	F1
Training set	0.989	0.989	0.988	0.988
Cross-validation set	0.989	0.989	0.986	0.986
Test set	0.966	0.966	0.955	0.960

Table5:Open set problem training results

It can be concluded from the training results that the training model can effectively exclude data of types other than 1-4 and label them as class 0, which will not be considered in the subsequent training. And the test accuracy is 0.966, which is an excellent test result.Unknown classes in student performance are accurately distinguished from the original dataset and the above-mentioned empirical and open-set risks are decreased. Save this model, this dataset open set problem is basically solved.

Subsequent optimisation requires the addition of a decision plane on the other side of the training sample in addition to the decision plane of the original SVM. and minimising the empirical and open space risks by adjustment (specialisation or generalisation) of these two planes.

Category 0 is still set in subsequent experiments, reflecting the open set results as well as the collection

of model prediction anomalies (prediction deviation values).

3.3 K proximity algorithm

Note: The experiments were conducted using k-fold cross-validation, where the initial sample was split into K sub-samples, a single sub-sample was retained as the data for validating the model, and the other K-1 samples were used for training. The cross-validation was repeated K times, once for each sub-sample, and the results were averaged.

The principle is that if a sample belongs to a category if the majority of the k most similar samples in the feature space belong to that category as well[7].

In this experiment, the Euclidean distance is used to calculate the distance between two points.

In the process of model tuning, a k-fold cross-validation approach was adopted in order to make the evaluated model more accurate, where k was taken to be 5.

Also, in order to select and tune the parameters, a hyperparameter-grid search is performed for the number of neighbours (n_neighbours hyperparameters) in the k-critical algorithm in order to find the best value of k.

	result	Programme	outcome 1	2	3	4
0	2	1	0.2	0.4	0.2	0.2
1	2	1	0.2	0.4	0.2	0.2
2	1	1	0.6	0.4	0	0
3	1	2	0.4	0.4	0.2	0
4	2	1	0.2	0.6	0.2	0

Table 6: Chart of selected poll predictions

The results of training the features selected in task1 is as follows.

	1	2	3	4	5
K=2	0.5142	0.5142	0.5428	0.4857	0.4285
K=3	0.5428	0.4857	0.4571	0.5428	0.4285
K=4	0.5428	0.5142	0.5714	0.5142	0.5428
K=5	0.5428	0.5142	0.6000	0.5142	0.4857
K=6	0.6285	0.5714	0.6571	0.5714	0.4285
K=7	0.6285	0.6000	0.6000	0.6000	0.3714
K=8	0.6857	0.6000	0.7142	0.6571	0.4857
K=9	0.6571	0.6285	0.6571	0.6285	0.4571

Table7: KNN superparametric grid search and cross-validation

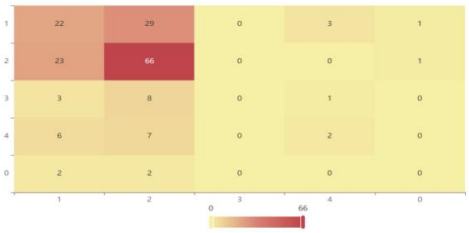


Figure 11: KNN classification results confusion matrix heat map

Based on the above heat map of the confusion matrix for the KNN classifier's classification results (the sum of each row indicates the number of true samples in that class, and the sum of each column indicates the number of samples predicted to be in that class) it can be concluded that a total of 55 sets of data with label 1 were tested, of which 22 were correctly predicted and 29 were predicted as 2. A total of 90 sets of data with label 2 were predicted, with 66 A total of 90 sets of data labelled 2 were predicted, with 66 data predicted correctly. A total of 12 sets of data with a label of 3 were predicted, with no correct predictions, and 8 were predicted as 2. A total of 15 sets of data with a label of 4 were predicted, and 7 were predicted as 2.

From the column analysis, it was concluded that the data predicted as type 2 had the most data and the highest hotness, indicating that the type 2 data accounted for a huge proportion of data with a large proportion of features, which to some extent affected the prediction accuracy of the model, while the number of type 3 was 0 indicating that the three types of features were extremely insignificant and would basically not be predicted.

After obtaining the data, the average accuracy, average variance and error index of the cross-validation of the KNN classifier at different values of k are visualised by means of line graphs to determine the range of values of k and the variation of accuracy.

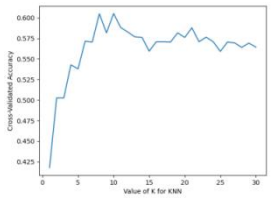


Figure 12: Average accuracy of cross-validation

By analysing the average accuracy images obtained through cross-validation, it can be concluded that the best training results are obtained when the value of k is between 6 and 12.

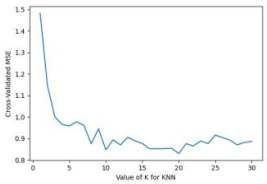


Figure 13: KNN model training mean square error

By image analysis of the mean square error, a smaller mean square error is needed to make the classification model more accurate, so the k-value should be between 8 and 20.

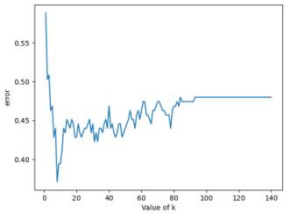


Figure 14: Error line graph for different K values of the model

From the above graph it follows that there are lower error values between 0, 15.

Using the visualisation results above and the results generated by the super-reference search and cross-validation, the analysis shows that the optimal parameter is K = 8, at which point the accuracy is 0.62857.

This best training model was saved for subsequent use, and given that the accuracy was still low in the optimal case, the distribution of points and clustered regions was plotted for visualisation of the classifier results (2-dimensional data was chosen here for visualisation).

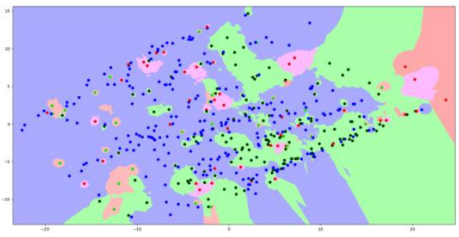


Figure 15: KNN Visual comparison

The results from the above figure show that the prediction area can cover most of the training data, but

there are still some training data distribution anomalies (some blue points into the green area and green points into the red area) which may be caused by the data set not being good enough.

The KNN model is trained and gives relatively good classification results for the dataset.

3.4 Random Forest Classifier

The Random Forest algorithm has good accuracy, can operate effectively on large data sets, handles input samples with high-dimensional features, and does not require dimensionality reduction and can evaluate the importance of individual features on the classification problem.

It works by generating multiple classifiers/models that each learn and make predictions independently. These predictions are finally combined into a combined prediction and therefore outperform any single classification in making predictions. In this experiment it is hoped that the integrated learning approach will improve prediction accuracy.

For each tree. Use M to denote the number of training use cases (samples) and N to denote the number of features: Randomly select one sample at a time and repeat M times. Randomly select n features, n << N, and build a decision tree

Adopt bootstrap sampling:

1. If there is no random sampling and the training set is the same for each tree, then the final trained tree classification result will be exactly the same.

2. If there is not a put-back sampling, then each tree is trained with different samples, all without intersection, which means that each tree is trained with a large variation. This can seriously affect the final vote.

The importance of each feature in the case of the random forest classifier was obtained by first obtaining a bar chart of the importance of the features derived from the classifier[8].

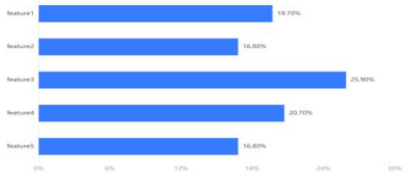


Figure 16: Importance of random forest classifier features

Based on the above graphs, the analysis shows that in the random forest, feature 3 occupies 25% of the feature importance and largely influences the final prediction results, while features 2 and 4 have a relatively small impact on the prediction results.

In the random forest model training, a 3-fold cross-validation was chosen.

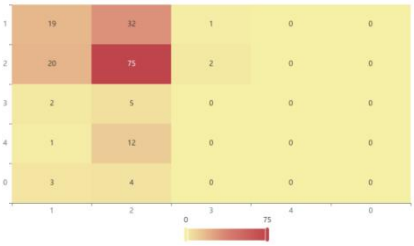


Figure 17: RFC classification results confusion matrix heat map

Analysis of the above heat map led to similar conclusions to those of the KNN classifier. A total of 52 sets of data labelled 1 were predicted, 32 were predicted 2 and 19 were predicted correctly. 97 sets of data labelled 2 were predicted and 75 were predicted correctly. 20 were predicted 1. 13 sets of data labelled 4 were predicted and no data were predicted correctly, 12 were predicted 2 and 1 was predicted 1. It was further concluded that the data labelled 2 or 1 were clearly characterised by a It is further concluded that the data labelled as 2 or 1 have obvious characteristics and are over-represented, which affects the accuracy of the model, while the data labelled as 3 and 4 have extremely insignificant characteristics and are difficult to be predicted.

Observation of the longitudinal heat in the graph shows that very few data were predicted to be 3 and none were predicted to be 4, suggesting that the distinctness and proportion of features vary considerably between labels, seriously affecting the results of the experiment.

In this experiment, the hyperparameters "n_estimators" and "max_depth" in the random forest are searched in a hyperparametric grid to visualize the results and find the model with the highest accuracy. An iterative approach is used to calculate the effect of different parameters on the model and to return the average accuracy.

First fix the max_depth hyperparameter to none and restrict the parameter n_estimators to the range 50,1500, resulting in an image.

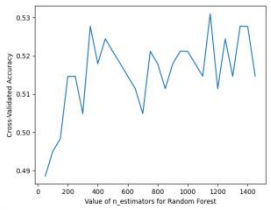


Figure 18: The n_estimators parameter affects accuracy

From the above figure, the number of trees (classifiers) of the random forest affects the accuracy of the results to a large extent, and with max_depth of none, the accuracy is peaked between 1100-1200, obtaining a precision of about 0.53.

Next, fix the number of trees in the forest to 100, max_depth to 1,20, and visualize the accuracy.

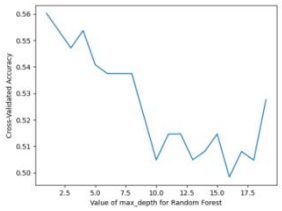


Figure 19: The max_depth parameter affects accuracy

Analysis of the above graph shows that this dataset has a higher training accuracy when the max_depth is small, which is not suitable for deep mining and can exclude subsequent training with neural networks.

Combining the above analysis and the results of the hyperparametric grid search, the iterative validation of "max_depth" and "n_estiimators" shows that the best model is obtained at a maximum depth of 1 and n_estimators=100, with an accuracy of 0.668 in the training set and 0.576 in the test set, and an F1 index of 0.53. The training result is relatively KNN The training result is relatively poor compared to KNN.

It can be seen that this dataset does not perform well on the random forest classifier, the accuracy is low, it is not suitable for deep analysis, the features are poorly represented and other classifiers should be considered.

3.5 Logistic regression(Optimising binary classifiers to accommodate multi-category datasets)

After the training and testing of the classifier described above, the analysis revealed that labels one and two accounted for a considerable proportion of the total and their features were relatively obvious, which had a significant impact on the prediction results. Therefore, logistic regression, a

classification model more suitable for solving binary classification problems, was considered for prediction, and improvements were made to logistic regression to make it applicable to the dataset. At the same time, partial 3,4 labeling accuracy was sacrificed to substantially improve the overall prediction results.

The input to a logistic regression is essentially the result of a linear regression.

$$h(w) = w_1x_1 + w_2x_2 + w_3x_3 \ldots + b$$

The regression results are fed into a sigmoid function trace, which outputs a probability value in the interval [0,1].

$$g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}},$$

This experiment was conducted by simple recursive training to make logistic regression applicable to this dataset. First, the data is trained for the first time after solving the open set problem, and the output of the training is that the data belongs to class 1, 2 or 3, 4. Next, the data belongs to class 1, 2 or 3, 4 is trained twice to produce the final classification result, thus creating a new logistic regression classifier.

In using logistic regression, the problem of log-likelihood loss has to be considered.

$$cost(h_\theta(x), y) = \sum_{i=1}^{m} -y_i log(h_\theta(x)) - (1 - y_i) log(1 - h_\theta(x))$$

It is well known that log(p), the larger the value of p, the smaller the result, so the gradient descent optimisation algorithm is used to reduce the value of the loss function.

Distinguish between categories	Accuracy
1,2/3,4	0.960
1/2	0.731
3/4	0.622
1/2/3/4	0.673

Table8: Optimised logistic regression prediction accuracy

In the first step of classification, it still appears that a large amount of data is considered as class 1, 2, and its accuracy is 0.56, which is not a significant improvement, but it has a correct rate of up to 0.7 when differentiating class 1, 2, sacrificing some of the differentiation for class 3, 4, thus improving the overall accuracy, and its final test set accuracy is 0.643, which is much higher than the random forest. It is slightly higher than the optimised KNN.

For the evaluation of logistic regression, the experiments considered precision (the proportion of samples with positive predictions that are true cases) and recall (the proportion of samples with true positive predictions that are true cases), and introduced an F1-score to reflect the robustness of the model[9].

$$F1 = \frac{2TP}{2TP + FN + FP} = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

For a more visual view, the ROC curve for the data is plotted with FRP on the horizontal axis and TRP on the vertical axis.

TPR=TP/(TP+FN) FPR=FP/(FP+FN)

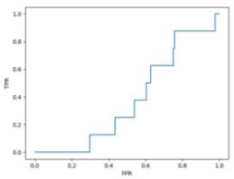


Figure 20: ROC curve

	precision	recall	f1-score	support
1	0.62	0.56	0.59	41
2	0.65	0.86	0.74	72
accuracy			0.64	113
macro avg	0.25	0.28	0.27	113
weighted avg	0.55	0.64	0.59	113

Table9: Accuracy and recall with a 1/2 class label division for example

From the analysis of the above icons, it is concluded that the improved logistic regression has a good adaptation on this dataset and can effectively improve its accuracy, especially for a large number of data with a label of 1/2.

Finally, the classifier is evaluated by the calculation of the AUC metric, where a pair of positive and negative samples is randomly taken in the result, and the probability of a positive sample being greater than a negative sample is the AUC metric, and a perfect classifier appears when the AUC is 1. In this experiment, the calculated AUC metric is 0.6, and it is known that the closer the AUC metric is to 1, the better it is. It can be seen that logistic regression has relatively good prediction results in this experiment, but still cannot achieve higher accuracy due to the data set.

3.6 Task 2 conclusion

In summary, the three optimal classifiers were selected for deeper optimization and parameter selection through several iterations of experiments, and the final accuracies obtained were as follows.

Classifier	Accuracy
K-Nearest Neighbour Algorithm	0.628
Random Forest Classifier	0.668
Optimised logistic regression classifier	0.673

Table10: Classifier accuracy

In summary, after a series of evaluations, the optimised logistic regression model had the highest accuracy of 0.673 as the optimal model due to the distinct and large number of class 1/2 features in the dataset. The KNN model tuned to the optimal parameters also showed a high accuracy of 0.62857 on the dataset and was able to draw a more distinct geographical distribution. Supervised learning is complete.

4 Unsupervised Classification

Unsupervised learning starts with unlabelled data to obtain the final classifier model to classify and predict the data.

As there are no labels, the experiments need to consider how to effectively generalise and group them. How the data should be effectively characterised in a compressed format. These will be discussed in the report.

In this experiment, PCA was chosen to reduce the dimensionality of the data and K-means was chosen for clustering.

The purpose of dimensionality reduction of data is to facilitate subsequent image visualisation. In this experiment, PCA dimensionality reduction, which performs better in supervised learning, was chosen, and the clustering algorithm was adopted to compare the highest feature retention down to 3 dimensions and the clearest visualisation down to 2 dimensions, respectively.

In this experiment, K-means clustering was performed by considering the distance between the samples in the following steps:

1, randomly set K points in the feature space as the initial

clustering centres.

2, for each of the other points calculate the distance to the K centres, unknown points select the nearest one of the clustering centres as the labelled category

3. Then, after the marked cluster centres, the new centroid (mean) of each cluster is recalculated

4. If the new centroids are the same as the original centroids, then the process ends, otherwise the second step is repeated.

The effect of clustering is evaluated in this experiment by means of a contour coefficient.

$$sc_i = \frac{b_i - a_i}{\max(b_i, a_i)}$$

For each point i, which is a sample in the clustered data, b_i is the minimum of the distance from i to all samples in the other clusters, and a_i is the average of the distance from i to its own cluster. The average of the contour coefficients of all the sample points is calculated.

The profile factor requires the model to maximise external distances and minimise internal distances.

If b_i>>a_i: converges to 1 the better, b_i<<a_i: converges to -1, the worse. The value of the contour coefficient is between [-1,1], the closer to 1 means that the cohesion and separation are relatively good.

As the clustering algorithm has a large contour coefficient when the number of clusters is very small, the minimum number of clusters in this experiment is 3.

4.1 Two-dimensional

First, the research downscaled the data to 2 dimensions by PCA, performed K-means clustering, visualised the results and calculated the contour coefficients, visualised the effect of different class numbers on the clustering results and selected the optimal number of clusters based on the contour coefficients. The number of clusters is specified between 2 and 10, and the contour coefficients are visualised.

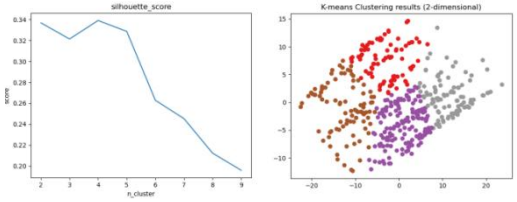


Figure 21: Relationship between profile coefficients and clustering and optimal clustering results(2-D)

Analysis of the above graphs shows that when the number of clusters is taken as 4, the contour coefficient is the largest and the clustering effect is the best. When the number of clusters is greater than 5, the contour coefficient becomes smaller rapidly and the clustering result decreases rapidly.

Taking the best clustering result, the number of clusters is 4, resulting in a schematic diagram of the clustering result, and calculating the contour coefficient at this time is 0.339145.

4.2 three-dimensional

Next, the experiments were carried out to reduce the dimensionality of the data to 3 dimensions, while retaining 80% of the original features, and K-means clustering was carried out to find the relationship between the number of clusters and the contour coefficients. The final contour coefficient clustering diagram was obtained.

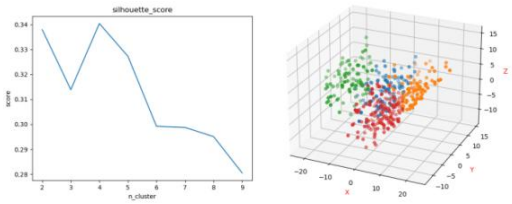


Figure 22: Relationship between profile coefficients and clustering and optimal clustering results(3-D)

The analysis yielded essentially the same results in 2 and 3 dimensions, indicating that for this data, there was little loss of data from 3 to 2 dimensions and the clustering effect was similar. On the basis of maximum feature retention, the contour coefficient is greatest and the clustering effect is best when the number of clusters is 4-dimensional, at which point the contour coefficient is calculated to be 0.340375.

4.3 Five-dimensional(Original data)

In order to maximise the retention of data features, the raw 5-dimensional data after processing was selected for clustering, the cluster centres for the optimal case were derived, and a post-clustering anomaly data display was performed.

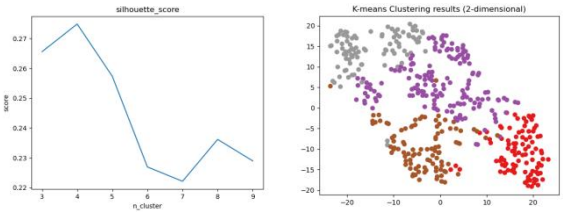


Figure 23: Original data clustering training results

	Q1	Q2	Q3	Q4	Q5
Category 1	31.20	5.31	9.40	16.10	12.23
Category 2	27.89	3.52	8.48	3.37	3.73
Category 3	28.59	4.37	8.67	2.93	15.86
Category 4	16.15	2.20	4.95	1.32	3.68

Table11: Clustering Centres

In the experiments, the number of outliers is calculated by calculating the distance between each point and the centre of the cluster, setting the percentage of abnormal data to 1%, thus setting the threshold for the outliers, judging the outliers and removing the abnormal data.

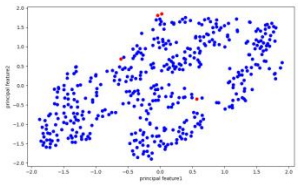


Figure 24: Clustering outlier removal

From the above study, for this dataset, the best K-means clustering model was obtained with a number of clusters of 4. The contour coefficient was the largest and the best clustering was achieved. Save the model.

5 Conclusion

The student classification dataset was assigned different categories based on scores. In this experiment, PCA and NMF dimensionality reduction results were selected as the final features. during the training of the dataset. It is crucial to solve the open set problem to effectively distinguish the "unknown categories". Better results can be obtained if the optimisation algorithms mentioned in the open set are used. In supervised learning, the accuracy of the optimised K-approach and logistic regression is 0.62 and 0.67 respectively. however, due to the small and uneven distribution of the data, optimisation of the algorithm is necessary so that the weight loss problem is addressed. The logistic regression results were significantly improved through multiple binary classification optimisation algorithms and can be used in subsequent studies.

Unsupervised learning was achieved using the K-means algorithm to select the maximum contour coefficient among the four classes of classification, when the model was optimal.

Reference:

[1] J. Yang, D. Zhang, A. F. Frangi, and J. Yang, "Two-dimensional PCA: a new approach to appearance-based face representation and recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 1, pp. 131–137, Jan. 2004, doi: 10.1109/TPAMI.2004.1261097.

[2] A. Pascual-Montano, J. M. Carazo, K. Kochi, D. Lehmann, and R. D. Pascual-Marqui, "Nonsmooth nonnegative matrix factorization (nsNMF)," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 3, pp. 403–415, Mar. 2006, doi: 10.1109/tpami.2006.60.

[3] M. Balasubramanian and E. L. Schwartz, "The Isomap Algorithm and Topological Stability," Science, vol. 295, no. 5552, pp. 7–7, Jan. 2002, doi: 10.1126/science.295.5552.7a.

[4] L. McInnes, J. Healy, N. Saul, and L. Großberger, "UMAP: Uniform Manifold Approximation and Projection," Journal of Open Source Software, vol. 3, no. 29, p. 861, Sep. 2018, doi: 10.21105/joss.00861.

[5] M. Wattenberg, F. Viégas, and I. Johnson, "How to Use t-SNE Effectively," Distill, vol. 1, no. 10, Oct. 2016, doi: 10.23915/distill.00002.

[6] W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult, "Toward Open Set Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 7, pp. 1757–1772, Jul. 2013, doi: 10.1109/tpami.2012.256.

[7] H. Zhang, "Research on information popularity prediction of multimedia network based on fast K proximity algorithm," International Journal of Autonomous and Adaptive Communications Systems, vol. 13, no. 2, p. 103, 2020, doi: 10.1504/ijaacs.2020.109808.

[8] T. Vanicek, "Classification in major depressive disorder using randomForest and various cortical and subcortical gray matter measures," Intrinsic Activity, vol. 7, no. Suppl. 1, p. A3.49, Sep. 2019, doi: 10.25006/ia.7.s1-a3.49.

[9] *S. W. Menard, Applied logistic regression analysis. Thousand Oaks, Calif.: Sage Publications, 2002.*