Statistical Learning Course independent paper: Prediction of Bank Customer's Long-term Deposit Product Purchases

Luhan Tang

Luhantang2002@163.com

School of Statistics, Beijing Normal University

Abstract

Based on the Banking Dataset Classification dataset, this report predicts whether bank customers will subscribe (yes/no) to the bank's term deposit products, using 13 independent variables: age, job, marital status, education, default, housing, loan, contact, month, day of the week, duration, campaign, and outcome. It is a classification problem. Initially, the dataset was preprocessed and analyzed preliminarily. Then, four different models were used for predictive modeling: Logistic Regression, Linear Discriminant Analysis, KNN Classifier, and Decision Tree. The K value for the KNN model was selected using the CV criterion, and the Decision Tree was pruned. Finally, all models were compared and analyzed using metrics like accuracy, recall rate, specificity, ROC curve, and AUC value. The results showed that the AUC values for the Logistic Regression, Linear Discriminant Analysis, and Decision Tree models were 0.8547613, 0.8589598, and 0.8425568, respectively, indicating good predictive effectiveness. Moreover, in terms of accuracy and recall rate, these three models exceeded 91%, significantly outperforming the KNN Classifier model. By comparing and evaluating different models, this report enhances the precision of prediction results and provides diverse strategic insights for banks using models to solve customer purchasing issues.

Keywords: Classification; Logistic (Regression); LDA (Linear Discriminant Analysis); KNN (K-Nearest Neighbors); Decision Tree

1 Introduction

1.1 background

With the continuous improvement of the legal environment and policy conditions for diversified operations in the financial industry, banks domestically and internationally are expanding their diversified businesses, either by establishing holding subsidiaries or through internal innovation. They are advancing integrated business strategies to create new profit growth points. Most banks consider diversified operations as a necessary means to achieve strategic transformation and enhance overall competitiveness, actively exploring related business in securities, insurance, funds, trusts, and other financial fields. These business revenues are important means for bank profitability.

In September 2008, the bankruptcy protection filing by Lehman Brothers, after the

Federal Reserve refused financial assistance, marked the full-blown global outbreak of the financial crisis triggered by "subprime loans." Originating in the United States, this crisis caused a chain reaction, plunging global credit markets into chaos, greatly impacting and disrupting the international financial order, and leading to a significant credit crunch effect in financial markets (Wang Liangliang, 2016).

In this financial crisis, a Portuguese bank experienced a decline in revenue, and bank managers wanted to know what actions to take in response. They found that the root cause was insufficient long-term deposit investments by customers. Therefore, the bank aimed to identify existing customers who were more likely to purchase long-term deposits and focus marketing efforts on them. The data is related to the direct marketing activities of the Portuguese banking institution. The marketing campaigns were primarily phone-based, with bank staff needing to call customers to inquire whether they would like to subscribe to the bank's term deposits (“yes”) or not (“no”).

1.2 Research purpose and data introduction

The aim of this classification study is to predict whether customers of the bank will subscribe (yes/no) to a term deposit. The Banking Dataset Classification from the Kaggle data platform collected relevant data of the bank's customers from May 2008 to November 2010, including 15 independent variables (X) and one dependent variable (y). The variables and their descriptions are as follows in the table below.

Table 1 Variables and their descriptions

variable	type	description
x_1 age	figure	One's age
x_2 job	category	Type of job (Administrator, Blue Collar, Entrepreneur, Maid, Management, retired, self-employed, Service, student, Technician, unemployed, unknown)
x_3 marriage	category	Marital status (divorced, married, single, unknown)
x_4 education	category	Education (high school, illiterate, professional courses, university degree, unknown)
x_5 default	category	Is there a default credit? (" No ", "Yes", "unknown")
x_6 housing	category	Is there a housing loan? (" No ", "Yes", "unknown")
x_7 loan	category	Is there a personal loan? (" No ", "Yes", "unknown")
x_8 contact	category	Contact communication type (" Cellular ", "Telephone")
x_9 month	category	Last contact month of the year (January to December)
x_{10} day_of_week	figure	Last contact day of the week (Monday to Friday)
x_{11} duration	figure	Last contact time, in seconds
x_{12} campaign	figure	Number of contacts made during this campaign and for this client (including the last contact)
x_{13} pdays	figure	The number of days that have passed since the last time the campaign contacted the customer (999 means the customer has not been contacted before)
x_{14} previous	figure	The number of contacts made before this campaign and for this client

x_{15} poutcome	category	Results of previous marketing campaigns (" failed ", "did not exist", "succeeded")
y	category	Does the customer subscribe to a time deposit? (" Yes ", "no")

data sources: Banking Dataset Classification

2 Modeling and estimation methods

2.1 Data preprocessing

2.1.1 Correlation Coefficient

Spearman Correlation Coefficient:

$$\rho = 1 - \frac{6\sum \mathrm{d}_\mathrm{i}^2}{\mathrm{n}(\mathrm{n}^2 - 1)}$$

Here, d^irepresents the difference in ranks of (x_i, x_j), and n is the number of data points in the variables.

If $\rho > 0$, the two variables are positively correlated; if $\rho < 0$, they are negatively correlated. In a heatmap, the closer the absolute value of ρ (|ρ|) is to 1, indicated by a darker color, the stronger the positive/negative correlation. The closer |ρ| is to 0, indicated by a lighter color, the weaker the correlation between the two variables.

2.2 Model introduction

2.2.1 Logistic Regression Model:

Given that the dependent variable y, indicating whether a customer subscribes to a product, is a binary variable ("no" or "yes," denoted as "0" and "1") and not a continuous variable, nonlinear functions are required for analyzing such binary dependent variables (Yu Liyong, 2004).

To study the relationship between the binary response variable y and the 13-dimensional predictor variable X, a logistic regression model is established:

$$P(y = 1|X) = \frac{e^{\beta_0 + \beta^T X}}{1 + e^{\beta_0 + \beta^T X}} =: P(X)$$

Here,

y = 1 or 0, is the dependent variable

$X = (x_1, x_2, \dots, x_{13})^T$is a 13-dimensional independent variable

β_0 and $\beta^\mathrm{T} = (\beta_1,\dots,\beta_{13})$ are unknown regression coefficients

P(X) is the logistic function

The likelihood function derived from P(X) is:

$$l(\beta_0, \beta) = \prod_{i=1}^{n} P(x_i)^{y_i} \left(1 - P(x_i)\right)^{1-y_i}$$

The log-likelihood function is:

$$L(\beta_0, \beta) = \sum_{i=1}^{n} \{y_i(\beta_0 + \beta^T x_i) - \log\left(1 + e^{\beta_0 + \beta^T x_i}\right)\}$$

Maximizing the log-likelihood function with respect to β and β_0, the maximum likelihood estimate of the regression coefficients is obtained as:

$$(\widehat{\beta_0}, \hat{\beta}) = argmax_{\beta_0,\beta} L(\beta_0, \beta)$$

The Newton-Raphson iterative method is used for solving:

$$\theta^{(k+1)} = \theta^{(k)} - (\frac{\partial^2 l(\theta)}{\partial\theta\partial\theta^T}|_{\theta=\theta^{(k)}})^{-1} \frac{\partial l(\theta)}{\partial\theta}|_{\theta=\theta^{(k)}}$$

Where,

$\theta = (\beta_0, \beta)^T$

$\theta^{(k)}$ represents the estimate of θ at the k-th step

For a given∈(can be 10^(-5)), stop the iteration when $\left\|\theta^{(k+1)} - \theta^{(k)}\right\|^2 <\in$

2.2.2 Linear Discriminant Analysis Model (LDA)

Establish the P(y=k | X=x) model, which means building a conditional distribution model for the response variable y given the predictor variable X. Based on the normality of the data distribution, use Bayes' theorem to estimate P(y=k | X=x) for classification purposes.

$$f(x) = \frac{1}{(2\pi)^{p/2}|\sum|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \sum^{-1} (x-\mu)\right)$$

Where,

The function $f(x)$ is a multivariate normal density function.

$X = (x_1, x_2, \dots, x_{13})^T$ is a 13-dimensional variable, following a multivariate normal distribution with different means but the same covariance matrix.

$\mu = (\mu_1, \dots, \mu_p)^T$, and $\sum$ is the covariance matrix.

The Bayes classifier assigns X=x to the category that maximizes the following expression:

$$\delta_k(x) = x^T \widehat{\sum}^{-1} \widehat{\mu_k} - \frac{1}{2}\widehat{\mu_k}^T \widehat{\sum}^{-1} \widehat{\mu_k} + log\widehat{\pi_k}$$

In this model, there are two types of prediction results for y, so k is 0 or 1

2.2.3 KNN Classifier Model

The basic idea of the KNN (K-Nearest Neighbors) algorithm (Zhang Ning, 2005) is to use all samples of known categories as references. It considers the K nearest (most similar) known samples in the training set to an unknown sample. The category of the new sample is determined based on the categories of these K samples, following the majority-voting rule. The unknown sample is classified into the category that is more prevalent among these K nearest neighbors.

The KNN method uses the Bayesian criterion to classify the observation x_0 into the class with the highest probability. The conditional probability $P(Y = j|X = x_0)$, where the possible values of Y are 0 or 1, categorizes the observation as 1 if $P(Y = j|X = x_0) > 0.5$.

Given a value of K and a prediction point x_0, the KNN classifier first identifies the K nearest training observations to x_0, denoted as N_0. Then, for each category j, it estimates a score based on the points in N_0 as an estimate of the conditional probability (James, G. 2015). This score, representing the likelihood of belonging to category j, is calculated using the proportion of points in N_0that belong to category j:

$$P(Y = j|X = x_0) = \frac{1}{K}\sum_{i \in N_0} I(y_i = j)$$

For the choice of K value: taking the smaller K value provides the most flexible fitting, but will lead to smaller bias and larger variance; Taking a larger K value provides a smoother fit, but will result in larger bias and smaller variance. This report will try multiple K's to see the predictive effect.

2.2.4 Decision Tree

The target variable of this study is a qualitative variable, therefore a classification decision tree model is chosen (Jigen Lin, 2004). The process of building a classification decision tree model includes:

Using recursive binary splitting method, with the classification error rate as the criterion for choosing the split point. The expression for the classification error rate is:

$$E = 1 - \max_{k}(\hat{p}_{mk})$$

Where, $\hat{p}_{mk}$ represents the proportion of class k in the training set of the m region.

The deviation of classification tree can indicate the effect of tree fitting data, and a small deviation indicates a good fitting effect. The expression is as follows:

$$-2\sum_{m}\sum_{k} n_{mk} log\hat{p}_{mk}$$

Where n_{mk} is the number of observations belonging to class k at the m-th terminal node.

2.3 Evaluation index of classification model

2.3.1 Confusion Matrix (Zou Hongxia, 2009)

Usually the class of interest is positive, and the other classes are negative. There are four possible scenarios in which the classification algorithm's predictions on the test set are either correct or incorrect. It can be arranged in the so-called confusion matrix as follows:

Table 2 Confusion Matrix

		Predicted Value	
		Positive	Negative
True	Positive	TP (True Positive)	FN (False Positive)
Value	Negative	FP (False Positive)	TN (True Negative)

The first letter indicates whether the prediction is correct, T(True) indicates that the prediction is correct, F(False) indicates that the prediction is wrong; The second letter indicates the prediction result of the classification algorithm, P(Positive) indicates that the prediction is positive, N(Negative) indicates that the prediction is negative.

Based on the confusion matrix, there are several indicators:

2.3.2 Accuracy

Accuracy is the ratio of the number of correctly classified samples to the total number of samples in a classification algorithm. It is important to note that in cases of class imbalance, the category with the larger proportion often becomes the main factor affecting accuracy. In such scenarios, accuracy may not effectively reflect the overall performance of the model. Its definition is as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

2.3.3 Precision

Precision rate refers to the proportion of the positive samples predicted by the classification algorithm. The value range is [0,1]. The greater the value, the better the prediction ability of the model, which is defined as:

$$Precision = \frac{TP}{TP + FP}$$

2.3.4 Recall

Recall rate is the proportion of positive samples predicted correctly by the classification algorithm to all positive samples. The value range is 0,1. The larger the value, the better the model's prediction ability. It is defined as:

$$Recall = \frac{TP}{TP + FN}$$

2.3.5 ROC Curve

In classification tasks, the test phase often involves obtaining a probability that indicates the likelihood of the current sample being a positive instance. We typically use a threshold value, where probabilities higher than this threshold are classified as positive instances, and those lower are classified as negative instances. If we lower this threshold, more samples will be identified as positive, increasing the recognition rate of positive classes but decreasing the recognition rate of negative classes. To visually represent these changes, the ROC (Receiver Operating Characteristic) curve is used to evaluate the effectiveness of a classification algorithm.

The vertical axis of the ROC curve represents the True Positive Rate (TPR) at different threshold levels, which is the proportion of correctly predicted positive samples out of all positive samples:

$$TPR = \frac{TP}{TP + FN}$$

TPR is actually the Recall rate, also known as Sensitivity.

The horizontal axis of the ROC curve is the false positive rate at different threshold levels, that is, the proportion of false predictions made by the classifier in all negative samples:

$$FPR = \frac{FP}{FP + TN}$$

1-FPR is also known as Specificity.

2.3.6 AUC

To assess the classification effectiveness of different model algorithms, a reasonable criterion is to compare the area under the ROC curve, known as AUC (Area Under the ROC Curve). The larger the AUC value, the better the predictive performance of the model. In fact, if AUC $=1$, it indicates a perfect classification algorithm; if $0.5 <$ AUC < 1, the algorithm is better than random guessing, and an appropriate threshold selection can offer predictive value. If AUC $= 0.5$, the model is no more effective than random guessing and has no predictive value; if AUC < 0.5, it performs even worse than random guessing.

3 Numerical simulation and case analysis

3.1 Data processing

This report has been preliminarily prepared by $x_1, x_2, \dots, x_{15}$ is used to predict y. Since x_i contains some non-numerical variables, it is necessary to preprocess the data and conduct a simple analysis before establishing the model.

3.1.1 Missing value processing

An unknown field in the data set indicates that the data is missing. Due to the huge amount of data in the total data set, a total of 32,950 items, data containing "unknown" should be deleted.

3.1.2 Variable removal

Since x_{13} (pdays) and x_{14} (previous) mainly contain the same number, the variance of the two is small and technically not helpful for prediction, so these two variables are removed.

Rename the remaining variable as x_1,....,x_{13}, y, a total of 13 independent variables and 1 dependent variable.

3.1.3 Outlier processing

Drawing the histogram of the variables, it is found that the three variablesx_1 (age), x_{11} (duration) and x_{12} (campaign) contain outliers. Therefore, the outliers in the three variables are replaced by the variables' respective upper and lower boundaries.

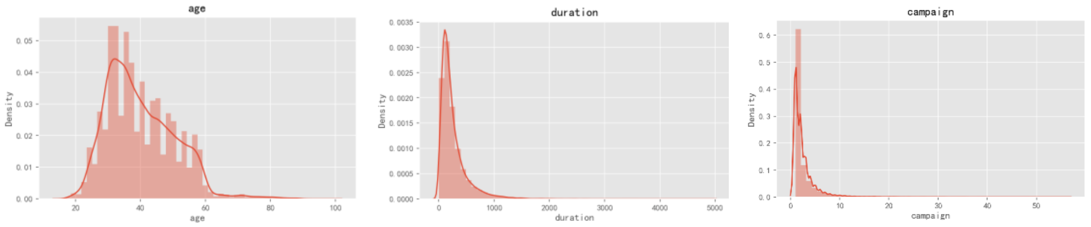


Figure 1 Histograms of age, duration and campaign

3.1.4 Coding classification features

According to the subsequent modeling needs, the classified variables need to be encoded and converted into numerical values. The conversion results are shown in Table 1 to Table 10 in the appendix. It is worth noting that the target variable y is encoded as "yes" =1 and "no" =0.

3.1.5 Divide test set and training set

The training set is divided into the test set (7:3). The training set data is used to fit the model, and the test set data is used to calculate the Accuracy, Precision and Recall of the model, draw the ROC curve and calculate the AUC.

3.2 Preliminary data analysis

3.2.1 Data balance

In the data set, about 88.73% of customers do not subscribe to time deposits, that is, y=0; 11.27% of customers subscribe to time deposits, i.e. y=1. The class distribution of the target variable y is about 89:11, indicating that the data set is unbalanced.

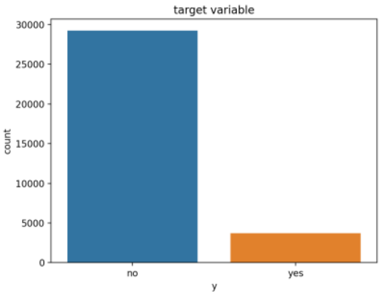


Figure 2 Target variable class distribution

Looking at the specific quantity of x_i, the following preliminary conclusions can be drawn: the top three customer occupations are administrative, blue-collar work and technical personnel; Among the customers who subscribe to time deposits, the majority are those who work in administration; A large number of clients are married; Most customers have no default credit; Many clients in the past have applied for home loans, but few have applied for personal loans; Mobile seems to be the preferred way to reach customers; The bank had already contacted many customers in May.

3.2.2 Correlation analysis

Firstly, the normality test is carried out, and the results are as follows: the significance of $x_{1\,,....,}x_{13}$ is lower than 5%, rejecting the null hypothesis and disobeying the normal distribution. Spearman correlation coefficient was chosen to explore the correlation.

Table 3 Normality test

	Kolmogorov-smirnoff (V)a Rielly's significance correction		
	Statistical	degree of freedom	significance
age	.095	32950	.000
job	.257	32950	.000
marital	.330	32950	.000
education	.203	32950	.000
default	.504	32950	.000
housing	.366	32950	.000
loan	.512	32950	.000
contact	.411	32950	.000
month	.242	32950	.000
day_of_week	.161	32950	.000
duration	.132	32950	.000
campaign	.255	32950	.000
poutcome	.473	32950	.000

The thermal map of the correlation coefficient of the derived variable x_i is drawn, and there is no highly correlated feature.

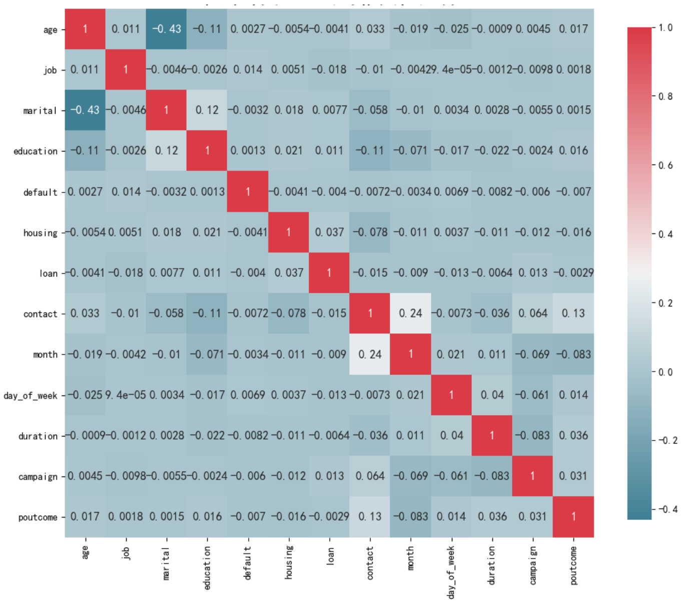


Figure 3 Correlation analysis

Based on the above data processing and analysis, four different models including Logistic regression model, linear discriminant analysis model, KNN classifier model and decision tree model are proposed for prediction research in order to predict the situation of customer ordering products (" yes "or" no "), and the prediction effect of each model is compared to select the optimal prediction model.

3.3 Modeling and Prediction

3.3.1 Logistic regression model

The Logistic regression model was established for the pre-processed data using R language, and the results were as follows:

Table 4 Logistic regression results

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	-2.550e-01	1.390e-02	-18.338	< 2e-16	***
age	1.441e-03	2.024e-04	7.117	1.13e-12	***
job	1.436e-03	5.332e-04	2.693	0.00708	**
marital	2.731e-02	3.408e-03	8.014	1.17e-15	***
education	8.080e-03	9.259e-04	8.727	< 2e-16	***
default	-3.501e-02	1.647e-01	-0.213	0.83168	
housing	4.520e-03	3.786e-03	1.194	0.23258	
loan	-2.328e-03	5.234e-03	-0.445	0.65652	
contact	-9.361e-02	4.149e-03	-22.560	< 2e-16	***
month	5.504e-03	8.512e-04	6.466	1.03e-10	***
day_of_week	-1.511e-03	1.344e-03	-1.125	0.26076	
duration	7.053e-04	1.066e-05	66.161	< 2e-16	***
campaign	-5.851e-03	1.231e-03	-4.753	2.01e-06	***
poutcome	1.120e-01	5.209e-03	21.503	< 2e-16	***

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

According to the above table, it can be seen that most of the variables are significant, but there are still four variables: x_5, x_6, x_7, x_10 are not significant. Below, the four non-significant variables are deleted, and the results of the two regression equations are compared and analyzed by re-fitting.

3.3.1.1 logistic regression model fitted with all explanatory variables $\mathbf{x_{1\,,\ldots,}x_{13}}$

Confusion Matrix:

Table 5 Confusion Matrix

	0	1
0	5923	542
1	172	178

Accuracy= 0.8953693

Precision= 0.9717801

Recall= 0.916164

Sensitivity= 0.916164

Specificity= 0.5208914

ROC curve is shown below:

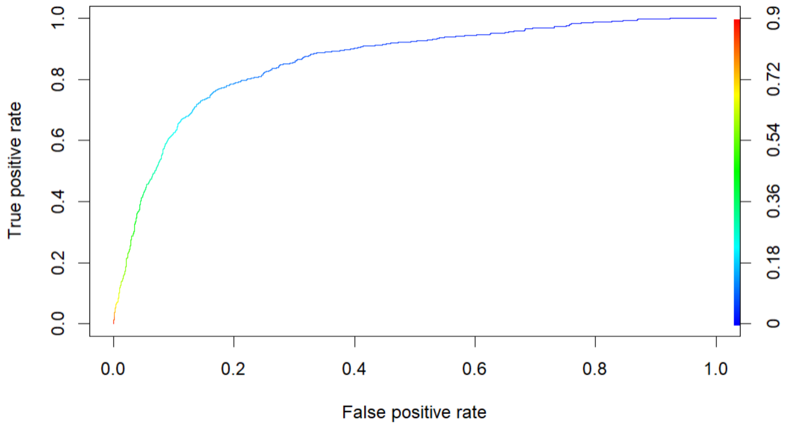


Figure 4 ROC curve

AUC = 0.8589598

3.3.1.2 Delete the logistic regression model fitted withx_5、x_6、x_7、x_{10}

Confusion Matrix:

Table 6 Confusion Matrix

	0	1
0	5933	548
1	162	181

Accuracy= 0.8959555

Precision= 0.9734208

Recall= 0.9154451

Sensitivity= 0.9154451

Specificity= 0.5276968

ROC curve is shown below:

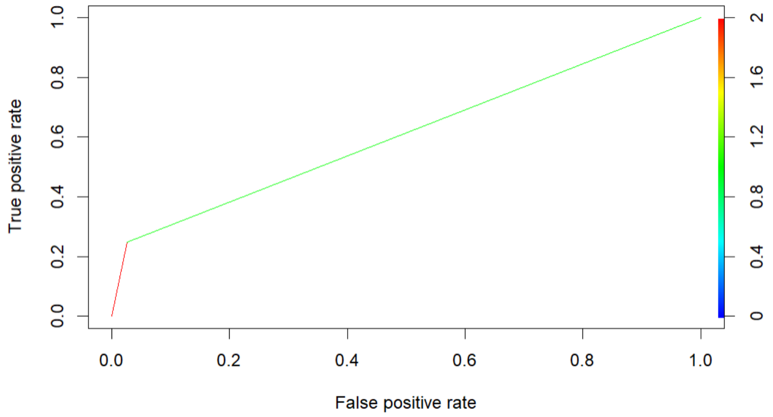


Figure 5 ROC curve

AUC = 0.6108531

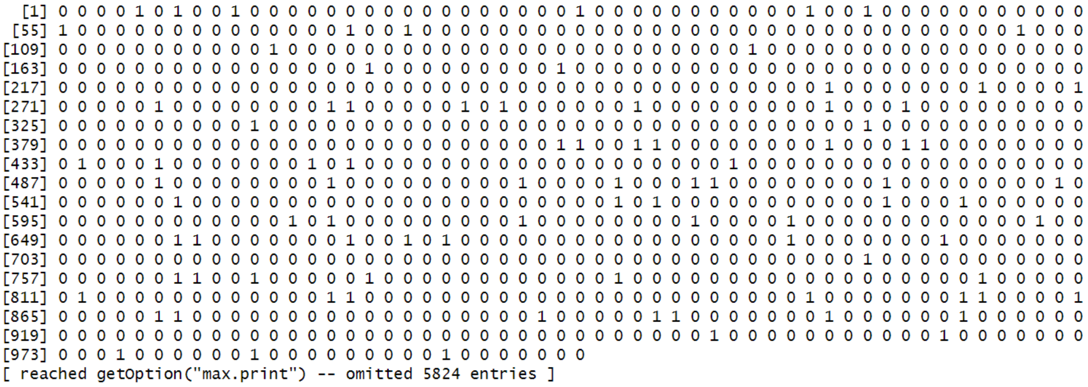
Due to the differences in AUC values, the model with larger AUC values and without deleting variables is selected. The Logistic regression model estimation equation is written as follows:

$$P(y=1|X)=\frac{e^{f(x)}}{1+e^{f(x)}}=:P(X)$$

$$f(x) = -0.2549578096 + 0.0014408498x_1 + \ 0.0014362939x_2 + 0.0273111423x_3 + 0.0080802643x_4 - 0.0350109019x_5 + 0.0045200607x_6 - 0.0023277120x_7 - 0.0936052216x_8 + 0.0055036132x_9 - 0.0015110232x_{10} + 0.0007052944x_{11} - 0.0058511946x_{12} + 0.1120176261x_{13}$$

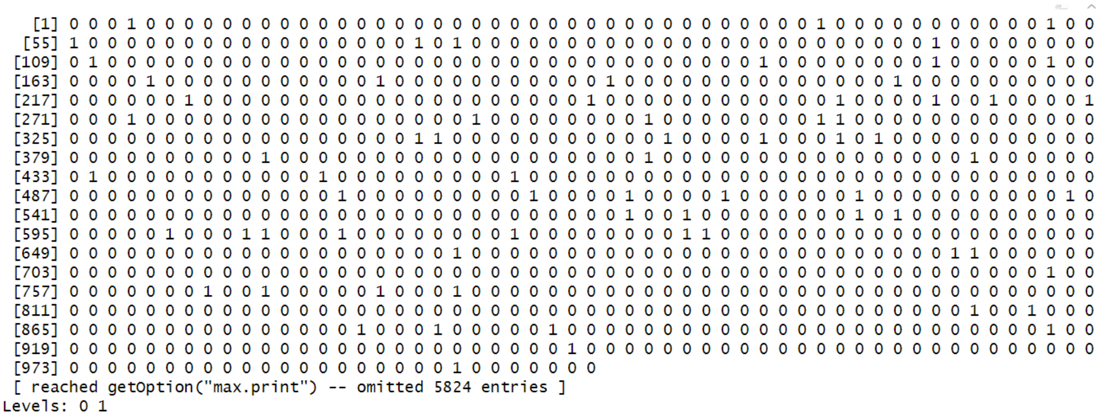
3.3.2 LDA

R was used to fit the LDA model to the training set, and the data in the test set was tested. The results were as follows:



```
  [1] 0 0 0 0 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0
 [55] 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
[109] 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[163] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[217] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 1
[271] 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 1 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0
[325] 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
[379] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0 0 0 0
[433] 0 1 0 0 0 1 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[487] 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0
[541] 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0
[595] 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
[649] 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 1 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
[703] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
[757] 0 0 0 0 0 0 1 1 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
[811] 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 1 0 0 0 0 1
[865] 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0
[919] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
[973] 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
[ reached getOption("max.print") -- omitted 5824 entries ]
```

Figure 6 Test set true values



```
   [1] 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0
  [55] 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
 [109] 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0
 [163] 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
 [217] 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 1 0 0 0 0 1
 [271] 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
 [325] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0
 [379] 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
 [433] 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [487] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0
 [541] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0
 [595] 0 0 0 0 0 1 0 0 0 1 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [649] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0
 [703] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
 [757] 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [811] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0
 [865] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
 [919] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [973] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
 [ reached getOption("max.print") -- omitted 5824 entries ]
Levels: 0 1
```

Figure 7 Test set predicted values

The confusion matrix is:

Table 7 Confusion Matrix

	0	1
0	5800	438
1	295	291

Accuracy= 0.892585

Precision= 0.9515997

Recall= 0.9297852

Sensitivity= 0.9297852

Specificity= 0.496587

ROC curve is shown below:

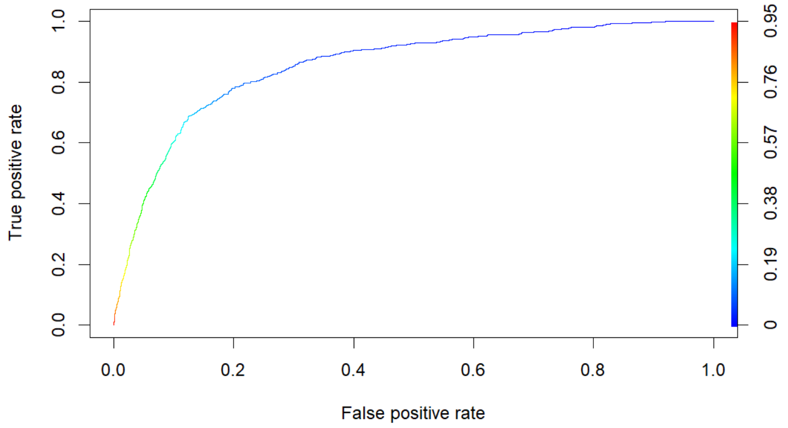


Figure 8 ROC curve

AUC = 0.8547613

3.3.3 KNN classifier model

KNN model is constructed with R, and the corresponding accuracy and AUC values are obtained by selecting different K.

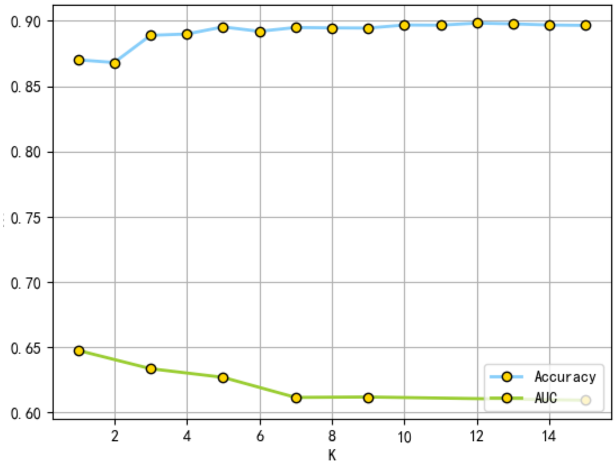


Figure 9: Accuracy and AUC corresponding to K

When K=1, the accuracy is about 0.87, which is not much different from the accuracy when K=3. However, the AUC value is the largest when K=1, so K=1 is chosen for model prediction.

The confusion matrix is:

Table 8 Confusion Matrix

	0	1
0	5678	417
1	466	263

Accuracy= 0.8706038

Precision= 0.9241536

Recall= 0.9315833

Sensitivity= 0.9315833

Specificity= 0.3607682

ROC curve is shown below:

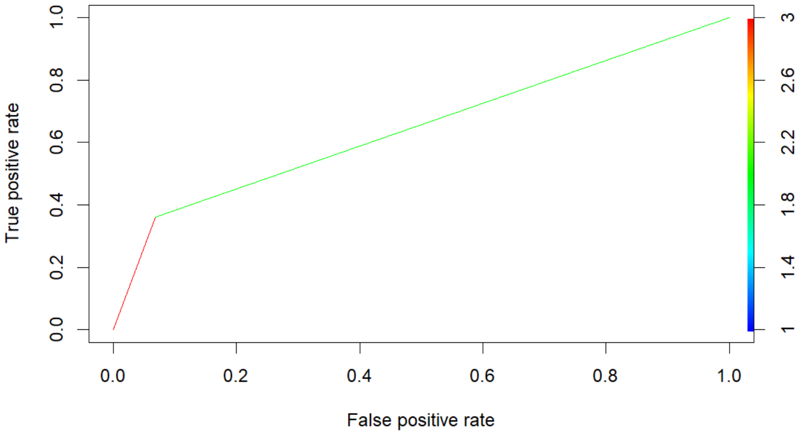


Figure 10 ROC curve

AUC = 0.6461757

3.3.4 Decision Tree

R is used to model the training set, and the classification process is as follows:

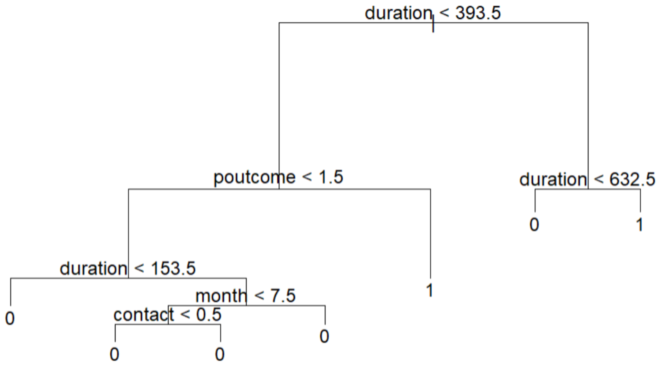


Figure 11 Decision tree model

The root node of this model is duration, the number of leaves is 7, the training error rate is 11%, and the average residual is 16238.

The confusion matrix is:

Table 9 Confusion Matrix

	0	1
0	5768	382
1	327	347

Accuracy= 0.896102

Precision= 0.9463495

Recall= 0.9378862

Sensitivity= 0.9378862

Specificity= 0.5148368

ROC curve is shown below:

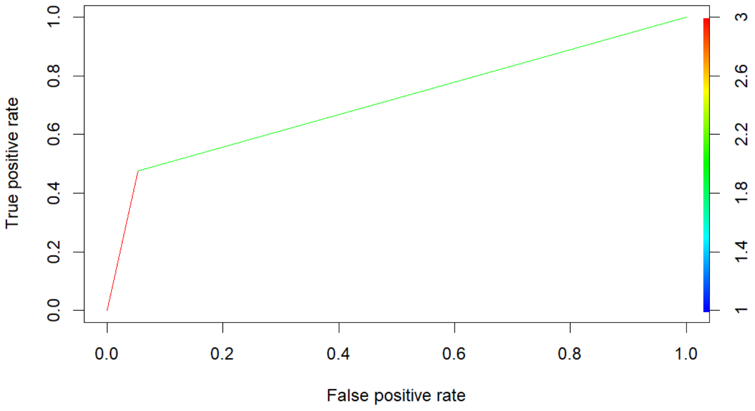


Figure 12 ROC curve

AUC = 0.711172

Because the decision tree is easy to cause the problem of overfitting, the classification tree is pruned below.

R was used to determine the optimal tree complexity by cross-validation CV. When the terminal junction number was 4, the cross-validation error rate was the lowest, with a total of 1817 cross-validation errors.

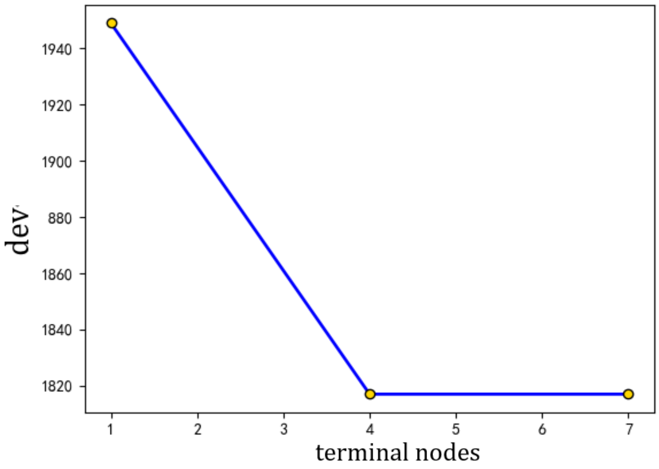


Figure 13 Relationship between terminal nodes and dev

The following selects the terminal knot number as 4, and draws the classification process after pruning:

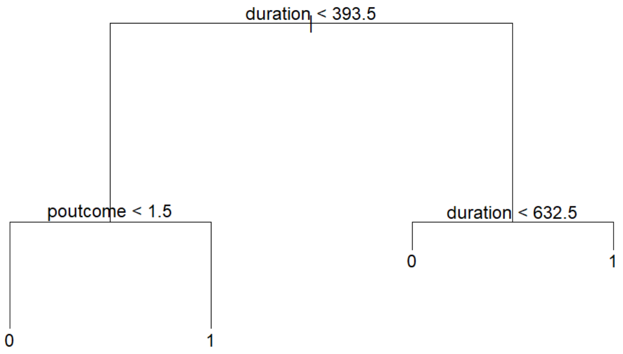


Figure 14 Decision tree model after pruning

Taking duration as the root node and combining with the leaf node, we can see the importance of the variable duration in this classification.

The confusion matrix is:

Table10 Confusion Matrix

	0	1
0	5768	382
1	327	347

Accuracy= 0.896102

Precision= 0.9463495

Recall= 0.9378862

Sensitivity= 0.9378862

Specificity= 0.5148368

ROC curve is shown below:

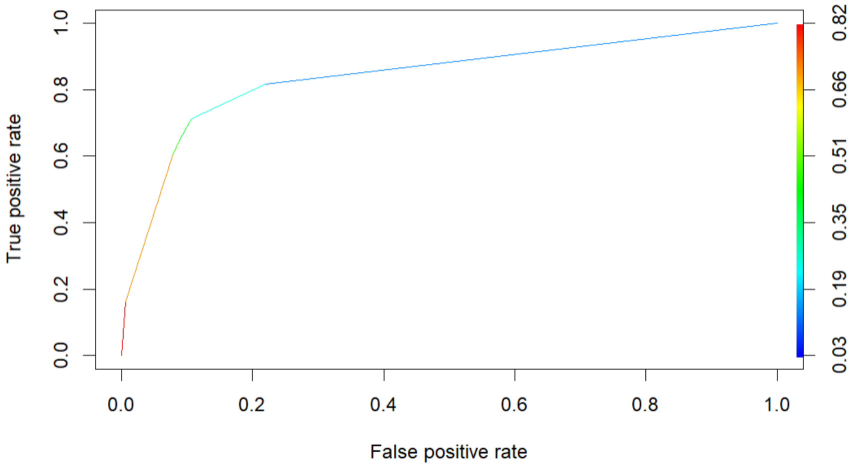


Figure 15 ROC curve

AUC = 0.8425568

Combining the results before and after pruning, it can be seen that the accuracy rate, accuracy rate and other indicators did not change, but the AUC value increased from 0.711172 to 0.8425568, and the effect of model prediction has been better improved.

4 Research results and discussion

Table 11 Comparison of indicators of the four models

	Model name			
name of index	Logistic	LDA	KNN	Decision Tree
Accuracy	0.8953693	0.892585	0.8706038	0.896102
Precision	0.9717801	0.9515997	0.9241536	0.9463495
Recall/Sensitivity	0.916164	0.9297852	0.9315833	0.9378862
Specificity	0.5208914	0.496587	0.3607682	0.5148368
AUC	0.8589598	0.8547613	0.6461757	0.8425568

Among them, K=1 is selected in KNN model; The pruned model is used in the decision tree model.

4.1 Index analysis

4.1.1 Accuracy:

Among all models, the decision tree model has the highest accuracy, but it is not much different from the Logistic model and the LDA model, while the KNN model has a relatively lower accuracy.

4.1.2 Precision

The model with the highest accuracy rate is Logistic model, LDA model and decision tree model have the same accuracy rate, and KNN is significantly lower.

4.1.3 Recall rate/sensitivity:

The highest recall rate/sensitivity is the decision tree, followed by KNN, LDA and Logistic model.

4.1.4 Specificity:

The specificity of the four models is relatively low, among which the Logistic model is the highest, followed by the decision tree model, LDA model and KNN model.

4.1.5 AUC：

Among the four models, the KNN model with the smallest AUC, only 0.64, has the worst prediction effect. The AUC of Logistic model and LDA model is around 0.85, and the AUC of decision tree is slightly lower than both. Except KNN model, the other three models have better prediction effect.

4.2 Comprehensive analysis

In this study, the bank aims to identify existing customers with a higher likelihood of purchasing long-term deposits, i.e., predicting those customers where y=1. The bank plans to focus its marketing efforts on these identified customers. For predicting y, a high recall rate is desired, as the bank wants to identify every customer likely to purchase long-term deposits and initiate subsequent marketing efforts. Targeting these customers, the bank has a high probability of selling its long-term deposit products. While prioritizing recall rate significantly increases the identification of actual buyers, it also raises the chance of falsely identifying non-buyers.

Therefore, the bank must also ensure the model's precision. If precision is too low, the bank will waste resources marketing to customers unlikely to purchase, incurring losses due to the costs of marketing efforts. A higher specificity means a lower probability of wrongly classifying non-buyers as buyers, beneficial for the bank's marketing strategy. Given the imbalance in the dataset, with the 0-1 ratio of the target variable being about 89:11, accuracy is not suitable for measuring classification effectiveness in this analysis.

While comparing these metrics, the AUC value can also be referenced to assess the model's predictive effectiveness. The study finds that the variable duration significantly impacts the model's prediction, suggesting that the timing of the last contact with a customer nearly determines their final decision. Hence, the bank should expand its contact personnel team and ensure a high frequency of customer contact, which is more likely to increase the purchase rate.

Comparing the four models, the following conclusions can be drawn:

The Logistic model has high precision, specificity, and AUC, which are beneficial for the next step of marketing arrangements. However, its recall rate is somewhat lower compared to other models, which might lead to missing out on some potential buyers who do not purchase due to lack of marketing, thus causing income loss for the bank. But, with a recall rate of 91.61%, it's still considered high, making it a good model.

The LDA model also has high precision and AUC, with its recall rate and specificity ranking third. However, both its precision and recall rates are quite high, making it a reasonably good model.

The KNN model has the lowest precision and specificity and a significant gap compared to the other three models. This may be due to the influence of the dimension p=13, as KNN tends to perform poorly with a larger number of dimensions. However, KNN has a recall rate of 93.16%, indicating a tendency to predict customers will purchase the product. This method could be considered if the bank is not concerned about marketing costs. Generally, with an AUC of only around 0.64, it is not considered a good predictive model.

The Decision Tree model has the highest recall rate, along with high specificity and AUC, although its precision is slightly lower. Still, it is also a good model given its high numeric value.

In summary, bank decision-makers can consider their costs and plans and choose between the Logistic, LDA, and Decision Tree models.

References

Wang Liangliang. The Impact of Financial Crisis, Financing Constraints, and Corporate Tax Avoidance [J]. Nankai Business Review, 2016, 19(01): 155-168.

Yu Liyong, Zhan Jiehui. Research on Default Probability Prediction Based on Logistic Regression Analysis [J]. Finance and Economics Research, 2004(09): 15-23. DOI: 10.16538/j.cnki.jfe.2004.09.002.

Zhang Ning, Jia Ziyan, Shi Zhongzhi. Text Classification Using the KNN Algorithm [J]. Computer Engineering, 2005(08): 171-172+185.

Introduction to Statistical Learning - Based on R Applications / (US) James, G., et al.; translated by Wang Xing, et al. Beijing: Mechanical Industry Publishing House, 2015.6

Luan Lihua, Ji Genlin. Research on Decision Tree Classification Technology [J]. Computer Engineering, 2004(09): 94-96+105.

Zou Hongxia, Qin Feng, Cheng Zekai, Wang Xiaoyu. ROC Curve Generation Algorithm for Binary Classifiers [J]. Computer Technology and Development, 2009, 19(06): 109-112.

Appendix

1. Variable coding results

Table 1

Catagory of job
0 blue-collar
1 entrepreneur
2 retired
3 admin.
4 student
5 services
6 technician
7 self-employed
8 management
9 unemployed
10 unknown
11 housemaid

Table 2

Category of marital
0 married
1 divorced
2 single
3 unknown

Table 3

Catagory of education
0 basic.9y
1 university.degree
2 basic.4y
3 high.school
4 professional.course
5 unknown
6 basic.6y
7 illiterate

Table 4

Catagory of def ault
0 unknown
1 no
2 yes

Table 5

Catagory of housing
0 no
1 yes
2 unknown

Table 6

Catagory of loan
0 no
1 yes
2 unknown

Table 7

Catagory of contact
0 cellular
1 telephone

Table 8

Catagory of month
0 nov
1 jul
2 may
3 jun
4 aug
5 mar
6 oct
7 apr
8 sep
9 dec

Table 9

Catagory of day_of_week
0 wed
1 mon
2 tue
3 fri
4 thu

Table 10

Catagory of poutcome

0 nonexistent
1 failure
2 success

Table 11

Catagory of y
0 no
1 yes

2. Code display

1. Data preprocessing: Python code

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import os
for dirname, _, filenames in os.walk('/kaggle/input'):
        for filename in filenames:
                print(os.path.join(dirname, filename))
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
train_data = pd.read_csv('D://大三上//统计学习//期末报告//数据//银行数据//new_train.csv')
df = pd.DataFrame(train_data)
df.head(5)
%matplotlib qt5
# percentage of class present in target variable(y)
print("percentage of NO and YES\n",df["y"].value_counts()/len(df)*100)
#查看缺失
data.isnull().sum()
#去除变量
data.drop(columns=["pdays", "previous"], axis=1, inplace=True)
#处理离群值
data.describe()
# compute interquantile range to calculate the boundaries
lower_boundries= []
upper_boundries= []
for i in ["age", "duration", "campaign"]:
        IQR= data[i].quantile(0.75) - data[i].quantile(0.25)
        lower_bound= data[i].quantile(0.25) - (1.5*IQR)
        upper_bound= data[i].quantile(0.75) + (1.5*IQR)
        print(i, ":", lower_bound, ",",    upper_bound)
        lower_boundries.append(lower_bound)
        upper_boundries.append(upper_bound)
# replace the all the outliers which is greater then upper boundary by upper boundary
j = 0
for i in ["age", "duration", "campaign"]:
        data.loc[data[i] > upper_boundries[j], i] = int(upper_boundries[j])
        j = j + 1
#编码
dftObjcat = dftObject.columns
for i in dftObjcat:
        print(f'Catagory of {i}')
        catlist = dftObject[i].unique()
        for j, val in enumerate(catlist):
                dftobjfinal = dftObject[i].replace({val:j+1},inplace=True)
                #print(dftobjfinal)
                print(j,val)
```

2. Correlation coefficient heat map: Python code

```
 import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns    # 可视化 w
inputdata = pd.read_excel("D://大三上//统计学习//期末报告//数据//银行数据//trainnoy.xls")
plt.rcParams['font.sans-serif'] = ['SimHei']    # 用来正常显示中文标签
plt.rcParams['axes.unicode_minus'] = False    # 用来正常显示负号
df = inputdata.copy()
_, ax = plt.subplots(figsize=(12, 10))    # 分辨率 1200×1000
corr = df.corr(method='spearman') # 斯皮尔曼秩相关系数
cmap = sns.diverging_palette(220, 10, as_cmap=True)
_ = sns.heatmap(
        corr,   # 使用 Pandas DataFrame 数据，索引/列信息用于标记列和行
        cmap=cmap,   # 数据值到颜色空间的映射
        square=True,   # 每个单元格都是正方形
        cbar_kws={'shrink': .9},   # `fig.colorbar`的关键字参数
        ax=ax,   # 绘制图的轴
        annot=True,   # 在单元格中标注数据值
        annot_kws={'fontsize': 12})   # 热图，将矩形数据绘制为颜色编码矩阵
plt.title("各解释变量之间的相关性强弱", fontsize=20)
plt.show()
```

3. Logistic regression: R code

```
obj=glm(y~age+job+marital+education+default+housing+loan+contact+month+day_of_week+duration+campaign+poutcome,data = train)
```

```
summary(obj)
fit.reduced <- glm(y ~ age + job + marital + education +
contact+month+duration+campaign+poutcome ,family=binomial(),data=train)
summary(fit.reduced)
coef(obj)
mydata = train
mydata$y[mydata$y == '1'] = 1
mydata$y[mydata$y == '0'] = 0
mydata$y = as.numeric(mydata$y)
index = sample(x = 1:2,size = nrow(mydata), replace = TRUE, prob = c(0.7,0.3))
train = mydata[index == 1, ]
test = mydata[index == 2, ]
logistic.model = glm(y~., data = train, family = binomial(link = 'logit'))
train_predict0 = predict(logistic.model, train, type='response')
train_predict = ifelse(train_predict0>0.5, 1, 0)
test_predict0 = predict(logistic.model, test, type='response')
test_predict = ifelse(test_predict0>0.5, 1, 0)
predict_value=test_predict0
true_value=test[,14]
```

4. LDA：R code

```
library(MASS)
lda.fit=lda(y~age+job+marital+education+default+housing+loan+contact+month+da
y_of_week+duration+campaign+poutcome,data=train)
lda.pred=predict(lda.fit,test)
names(lda.pred)
lda.pred
lda.class=lda.pred$class
#plot(lda.fit)
#lda.fit
lda.class#预测
test$y#实际
t=table(lda.class,test$y)
```

5. KNN：R code

```
library(class)
set.seed(1)   #设置一个随机种子
knn.pred = knn(train, test,train$y, k = i)
t=table(test$y,knn.pred)
correct = rep(0,15)
for(i in 1:15){
  fit_pre = knn(train, test,train$y,k=i)
  correct[i] = mean(fit_pre ==    test$y)
}
```

6. Decision Tree： R code

```
 #install.packages("tree")
library(tree)
library(rpart)
#install.packages("rpart.plot")
library(rpart.plot)
#install.packages("rattle")
library(rattle)
chose=ifelse(train$y>0.5,"1", "0")
bank=data.frame(train,chose)
tree.bank=tree(chose~.-y,bank)
summary(tree.bank)
plot(tree.bank)
text(tree.bank,pretty=0)
tree.bank=tree(chose~.-y,bank)
tree.pred=predict(tree.bank,test,type='class')
t=table(tree.pred,test$y)
set.seed(3)#剪枝
cv.bank=cv.tree(tree.bank,FUN = prune.misclass)
names(cv.bank)
cv.bank
prune.bank=prune.misclass(tree.bank,best=4)
plot(prune.bank)
text(prune.bank,pretty=0)
tree.pred=predict(prune.bank,test,type="class")
t=table(tree.pred,test$y)
```

7. Calculate each indicator + visual ROC: R code

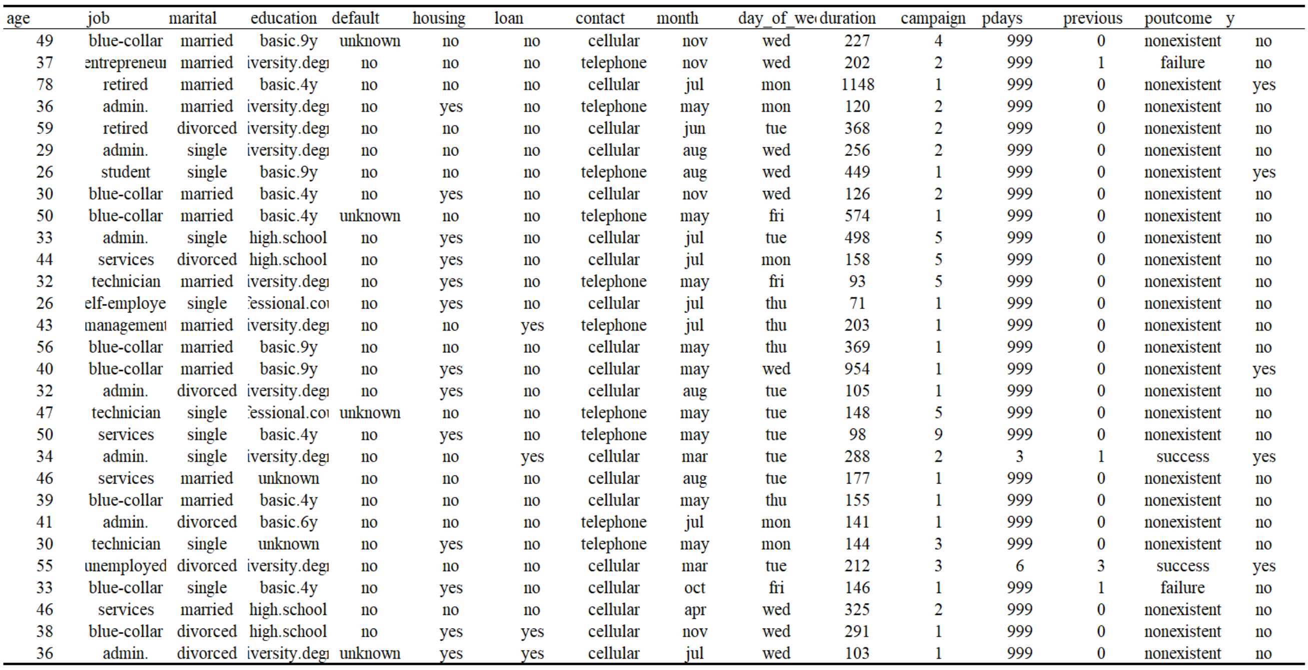
```
#混淆矩阵，显示结果依次为 TP、FN、FP、TN
t=table(test_predict,test$y)
t
tp <- t[1, 1]
tn <- t[2, 2]
fp <- t[2, 1]
fn <- t[1, 2]
print(accuracy <- (tp + tn)/(tp + tn + fp + fn))
print(precision <- tp/(tp + fp))
print(recall <- tp/(tp + fn))
print(sensitivity <- tp/(tp + fn))
print(specificity <- tn/(tn + fp))
#install.packages("caret", dependencies = c("Depends", "Suggests"))
library(pROC)
library(ggplot2)
library(magrittr)
#install.packages("ROCR")
library(ROCR)
pred <- prediction(predict_value,true_value)      #预测值(0.5 二分类之前的预测值)和真实值
performance(pred,'auc')@y.values            #AUC 值
perf <- performance(pred,'tpr','fpr')    #y 轴为 tpr(true positive rate),x 轴为 fpr(false positive rate)
plot(perf,colorize=TRUE)
```

8. Normality test: spss code

```
NEW FILE.
DATASET NAME 数据集1 WINDOW=FRONT.
GET DATA
  /TYPE=XLS
  /FILE='C:\Users\tlh\Desktop\train.xls'
  /SHEET=name 'Sheet1'
  /CELLRANGE=FULL
  /READNAMES=ON
  /DATATYPEMIN PERCENTAGE=95.0.
EXECUTE.
DATASET NAME 数据集2 WINDOW=FRONT.
DATASET CLOSE 数据集1.
EXAMINE VARIABLES=age job marital education default housing loan contact month day_of_week duration
    campaign poutcome
  /PLOT BOXPLOT STEMLEAF NPPLOT
  /COMPARE GROUPS
  /STATISTICS DESCRIPTIVES
  /CINTERVAL 95
  /MISSING LISTWISE
  /NOTOTAL.
```

2. Data display

Table 12



age	job	marital	education	default	housing	loan	contact	month	day_of_we	duration	campaign	pdays	previous	poutcome	y
49	blue-collar	married	basic.9y	unknown	no	no	cellular	nov	wed	227	4	999	0	nonexistent	no
37	entrepreneu	married	iversity.deg	no	no	no	telephone	nov	wed	202	2	999	1	failure	no
78	retired	married	basic.4y	no	no	no	cellular	jul	mon	1148	1	999	0	nonexistent	yes
36	admin.	married	iversity.deg	no	yes	no	telephone	may	mon	120	2	999	0	nonexistent	no
59	retired	divorced	iversity.deg	no	no	no	cellular	jun	tue	368	2	999	0	nonexistent	no
29	admin.	single	iversity.deg	no	no	no	cellular	aug	wed	256	2	999	0	nonexistent	no
26	student	single	basic.9y	no	no	no	telephone	aug	wed	449	1	999	0	nonexistent	yes
30	blue-collar	married	basic.4y	no	yes	no	cellular	nov	wed	126	2	999	0	nonexistent	no
50	blue-collar	married	basic.4y	unknown	no	no	telephone	may	fri	574	1	999	0	nonexistent	no
33	admin.	single	high.school	no	yes	no	cellular	jul	tue	498	5	999	0	nonexistent	no
44	services	divorced	high.school	no	yes	no	cellular	jul	mon	158	5	999	0	nonexistent	no
32	technician	married	iversity.deg	no	yes	no	telephone	may	fri	93	5	999	0	nonexistent	no
26	elf-employe	single	essional.co	no	yes	no	cellular	jul	thu	71	1	999	0	nonexistent	no
43	managemen	married	iversity.deg	no	no	yes	telephone	jul	thu	203	1	999	0	nonexistent	no
56	blue-collar	married	basic.9y	no	no	no	cellular	may	thu	369	1	999	0	nonexistent	no
40	blue-collar	married	basic.9y	no	yes	no	cellular	may	wed	954	1	999	0	nonexistent	yes
32	admin.	divorced	iversity.deg	no	yes	no	cellular	aug	tue	105	1	999	0	nonexistent	no
47	technician	single	essional.co	unknown	no	no	telephone	may	tue	148	5	999	0	nonexistent	no
50	services	single	basic.4y	no	yes	no	telephone	may	tue	98	9	999	0	nonexistent	no
34	admin.	single	iversity.deg	no	no	yes	cellular	mar	tue	288	2	3	1	success	yes
46	services	married	unknown	no	no	no	cellular	aug	tue	177	1	999	0	nonexistent	no
39	blue-collar	married	basic.4y	no	no	no	cellular	may	thu	155	1	999	0	nonexistent	no
41	admin.	divorced	basic.6y	no	no	no	telephone	jul	mon	141	1	999	0	nonexistent	no
30	technician	single	unknown	no	yes	no	telephone	may	mon	144	3	999	0	nonexistent	no
55	unemployed	divorced	iversity.deg	no	no	no	cellular	mar	tue	212	3	6	3	success	yes
33	blue-collar	single	basic.4y	no	yes	no	cellular	oct	fri	146	1	999	1	failure	no
46	services	married	high.school	no	no	no	cellular	apr	wed	325	2	999	0	nonexistent	no
38	blue-collar	divorced	high.school	no	yes	yes	cellular	nov	wed	291	1	999	0	nonexistent	no
36	admin.	divorced	iversity.deg	unknown	yes	yes	cellular	jul	wed	103	1	999	0	nonexistent	no

Note: There are 32952 rows in this dataset, only the first 36 rows are shown here.