Problem Chosen
A

2024
MCM/ICM
Summary Sheet

Team Control Number
JS06012024153

Deconstruction Risk Prediction of Cardiovascular Diseases Based on Deep Learning

Abstract

The accurate prediction of **cardiovascular diseases (CVD)** is of great significance for the prevention of CVD. This research paper presents a comprehensive approach to predicting the risk of CVD using **machine learning** algorithms, emphasizing the significance of meticulous analysis of clinical diagnostic data for early diagnosis and effective treatment strategies in light of the intricate etiology and prediction challenges associated with CVD.

The paper is structured around three key tasks: prioritizing and processing training data, applying machine learning classification methods to develop predictive models, and cross-validating these models for accuracy.

Based on physiological indicators, medical test results, and subjective information from patients, various data processing methods can be employed to incorporate deep learning techniques into corresponding models. The sigmoid function is utilized for fitting and simulating continuous variable data. To train large patient datasets with limited memory capacity, the L-BFGS-B algorithm is employed. Logistic regression analysis is initially chosen for classification purposes. By combining BMI, hypertension discrimination methods, etc., continuous data can be dichotomized, leading to the adoption of **Bayesian** models suitable for discrete data. Subsequently, **Netica** software is used to simulate training of parent-child nodes in order to establish their relationships with the common node 'Cardio'. On the other hand, the XGBoost classification model attempts to fit an optimal function by weighting a class of basic functions combined with CART decision trees. These three types of models differ in their approaches towards data processing; therefore it is crucial to compare their accuracies.

The paper proceeds to delve into exploratory data analysis, with a focus on innovative techniques such as dimensionality reduction and correlation analysis. Each model is trained and evaluated using various metrics including accuracy, confusion matrices, ROC curves, and learning curves.

Ultimately, the findings demonstrate that the Bayesian classification model outperforms other models with an accuracy of 0.793 while the XGBoost model achieves an accuracy of 0.728 and the logistic regression model has an accuracy of 0.723. This success can be attributed to its ability to handle high-dimensional data and learn incrementally which is crucial for adapting to new data. These insights contribute significantly towards ongoing efforts in precision medicine by facilitating more personalized diagnostic and treatment approaches for cardiovascular diseases.

Key Words: cardiovascular; machine learning; Bayesian; Netica

Contents

1 Introduction

1.1 Problem Background

Currently, doctors primarily rely on their experience and clinical test reports to assess whether patients are afflicted with cardiovascular and cerebrovascular diseases (CVD). The etiology of CVD is intricate, resulting in poor predictability. It is generally challenging for non-professionals to determine if they may be susceptible to such diseases. Individuals who are more concerned about their physical well-being often monitor their health based on routine physical examination indicators like blood pressure and blood lipids, while overlooking factors that could contribute to CVD, such as family medical history and pathological changes in other organs.[1]Disease prediction can significantly aid in early patient diagnosis, recommend effective treatment during the initial stages of disease progression, alleviate patient suffering, and reduce economic burden. Therefore, conducting comprehensive mining and analysis of patients' clinical diagnostic data holds immense importance. This includes identifying disease subtypes based on patient prognosis information and conducting research analyzing population differences among these subtypes to enhance the ability and level of individualized patient diagnosis and treatment. Considering the background information and restricted conditions identified in the problem statement, we need to solve the following problems:

Task 1 Prioritize the processing of the training data provided in the appendix. Based on this foundation, it is important to rely on cardiovascular disease-related studies to identify the inherent correlations between different variables and attempt to explore the connection between the data and the final output "cardio" through methods such as correlation analysis.

Task 2 Based on processed data, machine learning classification methods are employed to establish a mathematical model aiming to predict the likelihood of a specific patient developing cardiovascular diseases.

Task 3 Segmenting a portion of the dataset as the testing set, the mathematical model established using the training set is involved in cross-validation by assessing its accuracy. The conclusion is drawn based on selecting the mathematical model with a higher accuracy.

1.2 Literature Review

Numerous studies have been conducted on the risk factors associated with cardiovascular diseases. One editorial article specifically focused on the influence of gender on CVD and has demonstrated its impact.[2]Research from a perspective of developing countries has shown that basic patient information such as age and BMI (Body Mass Index) is also associated with CVD.[3] Meanwhile, a study conducted on the Spanish population has found that certain unhealthy habits among local residents, such as excessive alcohol consumption, smoking, and others, are also associated with this disease.[4] With the spread of COVID-19, Agewall suggests that medical indicators of patients, such as blood glucose levels, also have an impact on cardiovascular diseases.[5] Research on deep learning for cardiovascular disease prediction has been slow due to difficulties in feature extraction and other reasons.[6] Cuadrado-Godia and colleagues discovered a substantial rise in the utilization of neural networks, machine learning, and deep learning techniques in image processing to accurately assess the severity of cerebral small vessel disease;[7] Wang et al. demonstrated the capability of deep learning in automatically extracting pertinent features from diverse and complex datasets;[8] This has significant implications for various fields including precision medicine, disease diagnosis, epidemic response, and prevention. Similarly, Dan et al. developed deep CNN models specifically tailored for disease analysis, and their findings were comprehensively analyzed stroke.[9] The above studies demonstrate the feasibility of neural networks for predicting similar diseases in medicine, but the accuracy of neural network prediction is largely unspecified in the studies. Therefore, this paper utilizes various data processing methods to incorporate these related variables into different machine learning models for prediction. By comparing the accuracy of these machine learning models and considering their fitness to the attached data, the most suitable predictive model for the dataset is determined.

1.3 Our Work

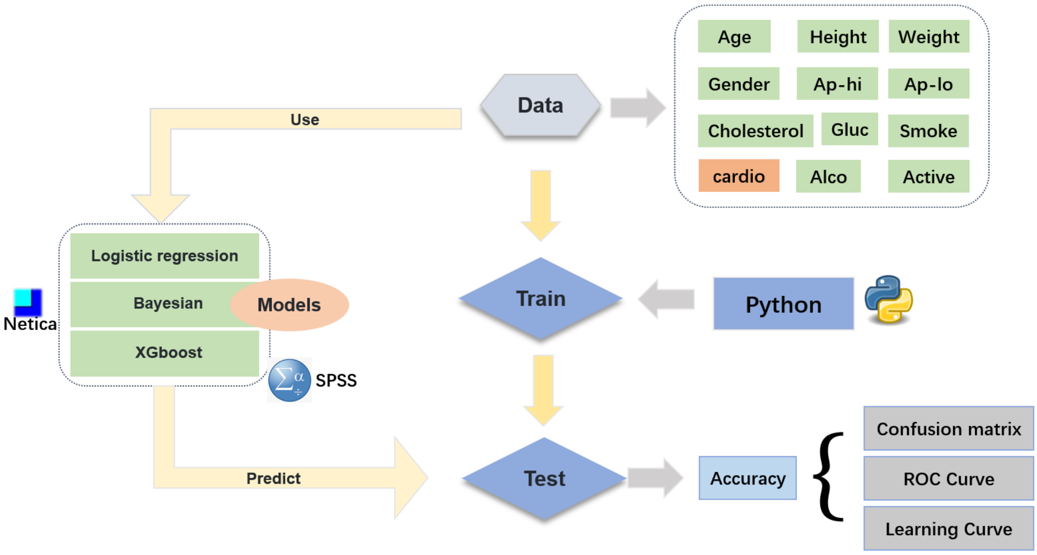


Figure 1: flow chart

2 Assumptions and Data Preprocessing

2.1 Assumptions

Assumption 1: Data with diastolic blood pressure greater than systolic blood pressure are assumed to be outliers.

⇒ Justification:The rationale behind assuming that data with diastolic blood pressure greater than systolic blood pressure are outliers is that it contradicts the physiological norm. In a healthy individual, systolic blood pressure should be higher than diastolic blood pressure.

Assumption 2: The physiological indicators of height and weight are both limited to a fixed range, where the BMI index for both should be above 18.5, while severe obesity (BMI > 32) is allowed.

⇒ Justification: The rationality behind this assumption is that the BMI index, which is calculated based on height and weight, has established ranges that correspond to different categories of weight. A BMI below 18.5 indicates underweight, while a BMI above 32 denotes severe obesity.

Assumption 3: The patient's medical test data are obtained from accurate medical instruments without significant errors. The patients selected are representative in terms of geographical location and can reflect general characteristics based on sample data.

⇒ Justification: The assumption of the reliability of the sample data ensures that the subsequent data in the machine learning process is trustworthy.

2.2 Data Preprocessing

Since this research focuses on the causes of cardiovascular diseases in humans, it is not reasonable to simply rely on the cold and simplistic numerical values presented in the table. This necessitates that the data be closely associated with the human body and ensure that these data are logically sound.

For model one (Logistic regression classification):

The logistic regression model, proficient in handling both discrete and continuous data, allows for the retention of the dataset's fundamental structure. Initial data processing involves the following steps:

1. Outlier Detection and Processing: Statistical methods are employed to identify outliers in physiological and medical testing indicators. Outliers are subsequently removed to mitigate any adverse impact on the model.

2. Feature standardization: Numerical features, such as physiological and medical testing indicators, undergo standardization to ensure consistent scaling. Z-score normalization is implemented to enhance the stability of model training.

3. Categorical feature coding: Categorical features (e.g.,gender, cholesterol) are subjected to One-Hot Encoding, transforming them into a numerical format understandable by the model.

4. Data Segmentation: The dataset is partitioned into training and test sets, maintaining a 70-30 ratio. This division ensures effective model validation and testing post-training.

For model two (Bayesian classification):

Utilizing a Bayesian network model necessitates discrete nodes in the network. For several continuous data types, processing is conducted using the BMI scale, high blood pressure ratings, etc. Following continuous data processing, discrete data is normalized to facilitate integration into the model.

For model three (XGboost classification):

XGBoost, exhibiting flexibility with no specific requirements on the nature of input data, can seamlessly handle mixed feature types. For XGBoost model data, we apply the dataset operations from Model One, including outlier processing and One-Hot Encoding.

3 Exploratory Data Analysis

3.1 Innovative analysis of data

For the innovative analysis part of the data, it is clear from analyzing the data that the data provides an analysis of whether or not the participant suffers from cardiovascular disease from three dimensions and through 11 indicators. From the perspectives of the number of indicators and the relationship between the indicators, it was decided to use the method of data dimensionality reduction and standardization and correlation analysis for the innovative analysis of the data.

3.1.1 Data downscaling and normalization

After reviewing the relevant literature, the downscaling and standardization of five indicators were performed, namely: height, weight, diastolic blood pressure, systolic blood pressure, and age.

Height and weight were replaced with BMI (Body Mass Index), which is calculated as weight divided by height squared. The following ranges were used to categorize BMI: 18.5<=BMI<=23.9 for normal weight, recorded as type 0; 24.0<=BMI<=27.9 for overweight, recorded as type 1; 28.0<=BMI<=32.0 for obesity, recorded as type 2; BMI>=32.9 for severe obesity, also recorded as type 3.

As for diastolic and systolic blood pressure, values were considered as hypertension if diastolic blood pressure was greater than or equal to 90mmHg and systolic blood pressure was greater than or equal to 140mmHg. Hypertension was recorded as 1, otherwise it was recorded as 0.

For age categorization, the following groups were established: 30-35 recorded as 0, 36-40 recorded as 1, 41-45 recorded as 2, 46-50 recorded as 3, 51-55 recorded as 4, 56-60 recorded as 5, and 60-65 recorded as 6. Thus, the data was divided into 7 groups.

3.1.2 Correlation analysis

The data provided three dimensions: physiological indicators, medical indicators, and subjective information, and we analyzed three sets of correlations for each of the 12 indicators within the three dimensions.

I. A significant correlation was obtained between sex and height with a correlation coefficient of 0.533 and a low correlation between height and weight with a correlation coefficient of 0.314.

Correlation analysis of systolic blood pressure, diastolic blood pressure, cholesterol level, blood glucose concentration of medical indicators

The results are as follows:

	age	gender	height	weight
age	1 (0.000***)	-0.021 (0.000***)	-0.082 (0.000***)	0.063 (0.000***)
gender	-0.021 (0.000***)	1 (0.000***)	0.533 (0.000***)	0.173 (0.000***)
height	-0.082 (0.000***)	0.533 (0.000***)	1 (0.000***)	0.314 (0.000***)
weight	0.063 (0.000***)	0.173 (0.000***)	0.314 (0.000***)	1 (0.000***)

Note: ***, **, * represent 1%, 5%, and 10% significance levels, respectively.

II. A significant correlation between systolic and diastolic blood pressure was obtained, with a correlation coefficient of 0.742; cholesterol levels showed a low correlation with blood glucose concentration, with a correlation coefficient of 0.407.

Correlation analysis of subjective information on whether or not they smoke, drink alcohol, exercise, and have cardiovascular disease

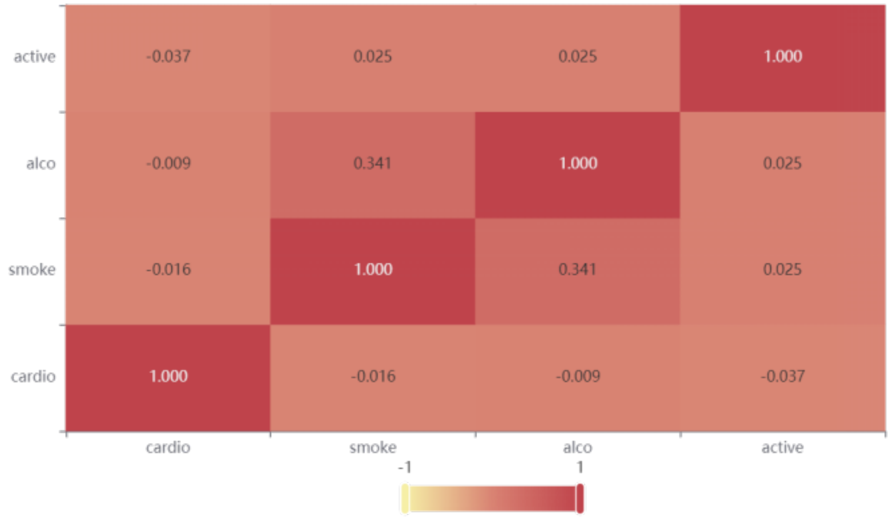


Figure 2: Medical Indicator Correlation Heatmap

A low correlation was obtained between whether a patient smoked or not and whether they drank alcohol or not, with a correlation coefficient of 0.341.

4 Machine Learning Methods for Classification

4.1 Logistic regression classification

In etiological research on diseases, it is often necessary to analyze the quantitative relationship between the occurrence of the disease and various risk factors. For example, studying the relationship between cardiovascular disease occurrence and risk factors such as smoking, drinking, and poor dietary habits. However, when using multiple linear regression analysis, if the dependent variable y is a binary variable (usually taking values 0 or 1), it does not meet the assumptions of normal distribution and equal variance. If linear regression analysis is applied forcefully, the predicted values may exhibit significant bias and not be interpretable.[10] Indeed, when the output variable is a binary variable, logistic regression is a well-suited choice.

4.1.1 The Establishment of Function

Hypothesis Function

The hypothesis function of logistic regression uses the logistic function (also known as the sigmoid function) to map the linear combination of input features to a probability value between 0 and 1. The expression of the hypothesis function is as follows:

$$h_\theta(x) = \frac{1}{1 + e^{-(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n)}} \tag{1}$$

Scatter plots illustrating the score proportions of certain continuous variables are shown below(The remaining discrete variables and similar data on the cause are included in the appendix):

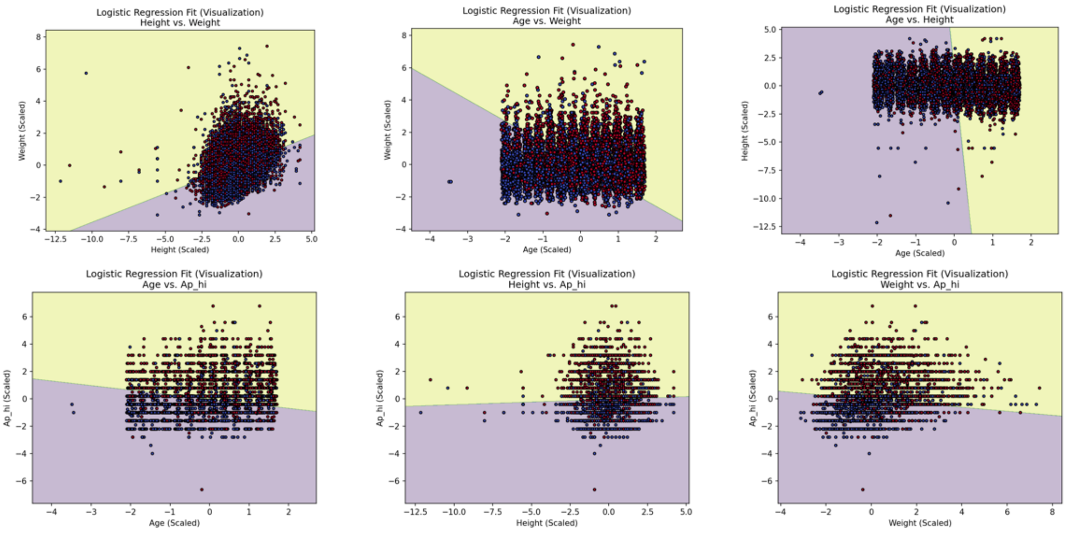


Figure 3: Visualization of the Logical Analysis of Continuous Data Relationships

Cost Function

Logistic regression uses the log loss function to measure the performance of the model. For a single sample, the expression of the loss function is as follows:

$$J(\theta) = -ylog(h_\theta(x)) - (1-y)log(1-h_\theta(x)) \tag{2}$$

Where J(θ)represents the loss function, y is the actual class label of the sample (0 or 1), and hθ(x) is the predicted probability obtained through the hypothesis function.

Limited-memory Broyden-Fletcher-Goldfarb-Shanno with bounds

The L-BFGS-B algorithm is a type of optimization algorithm based on the BFGS method. It is designed specifically for solving unconstrained optimization problems by incorporating boundary constraints. In the case of analyzing a dataset containing nearly 70,000 patient records, the L-BFGS-B algorithm employs a "limited-memory" approach that eliminates the need to store the complete Hessian matrix during computations. This method efficiently optimizes the problem while considering the boundary constraints and addresses the memory limitations associated with storing and manipulating large Hessian matrices.The specific steps of the algorithm are as follows:

Step 1 Initialization of Parameters: At the initial stage, a set of parameters is selected. The choice of can be a zero vector or any other suitable initial values depending on the problem at hand.

Step2 Computing Gradient: For a given set of parameters , the gradient vector J() is computed by calculating the partial derivatives of the loss function J() with respect to each parameter. The gradient represents the direction of change of the loss function at the current parameter values.

Step3 Determining the Step Size via Line Search: The appropriate step size is determined through line search, which involves using the Armijo line search with a decay factor multiplied by the step size. The process is iterated until certain convergence conditions are met.

Step4 Parameter Update: Using the L-BFGS-B algorithm, the parameters are updated iteratively based on the inverse of the approximate Hessian matrix. According to the update rule of L-BFGS-B, the parameter update formula can be represented as follows:

$$\theta_{new} = \theta_{old} + \alpha p \tag{3}$$

θnew represents the updated parameter vector, θold represents the current parameter vector. is the step size determined through line search, and p is the update direction which is calculated using the following formula derived from the L-BFGS-B algorithm:

$$p = -B_k \nabla J(\theta_k) \tag{4}$$

In the context of the L-BFGS-B algorithm, Bk represents the inverse of the approximate Hessian matrix at the k-th iteration, while
$nablaJ(\theta_k)$denotes the gradient of the loss function J with respect to the parameter θ.

Following the principles of quasi-Newton methods, the L-BFGS-B algorithm iteratively updates the parameters using the inverse of the approximated Hessian matrix. By leveraging limited historical information, it estimates the current point's gradient and inverse Hessian matrix. This approach avoids the direct computation and storage of the complete Hessian matrix, thereby reducing computational and storage costs.

4.1.2 Model Training

Step 1 Data Standardization

To expedite the convergence speed of the model, eliminate data bias, and enhance the model's generalization capability, we computed the mean and standard deviation of each patient's data. This process aids in removing bias from the data, thereby mitigating the risk of overfitting the training data. Consequently, the model's ability to generalize and perform well on unseen data is improved.

For each data point, Z-score standardization was employed to standardize the values:

$$x_i' = \frac{X_i - \overline{X}}{\sigma} \tag{5}$$

Step2 Data Splitting

In order to mitigate overfitting and assess the generalization ability of a model, it is essential to initially partition the dataset into training and testing sets. By observing the performance on the testing set, it becomes possible to select the optimal model or parameter configuration based on metrics such as accuracy, precision, recall, or other relevant indicators. This approach serves as a crucial step in guaranteeing reliable evaluation of the model's capability to generalize beyond the training data.

To ensure a fair sampling and to minimize bias, a random partition was performed to split the dataset into training and testing sets. The data from the larger group were selected for machine learning training, using a ratio of 7:3, while the remaining 30% were set aside to serve as the testing set for evaluating the accuracy of the model.

Step3 Modal Prediction

To train a logistic regression model using 70% of the given data and setting the random state to 42, follow these steps and obtain the fitted model:

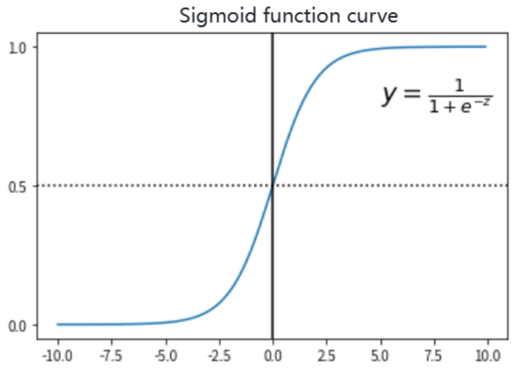


Figure 4: sigmoid function curve

After analyzing the predictions, we get the following final probability function:

$$P(cardio = 1) = \frac{1}{1 + e^{-(0.06577022 - 0.0219786721 * Age + 0.359025922 * Height + \cdots - 0.0735299358 * Bmi)}} \tag{6}$$

Step4 Modal Elevation

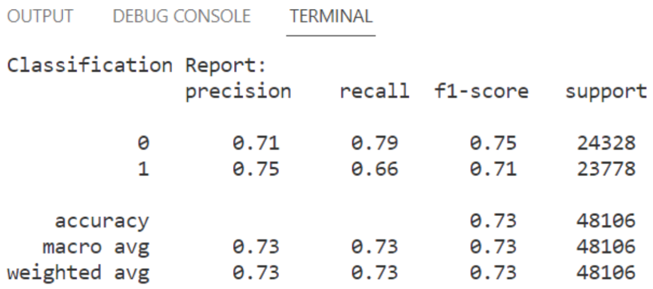
Take the last 30% of the data and input it into the fitted model. Call the predict method to calculate the predicted values for the disease condition. Evaluate the models prediction performance using metrics such as accuracy, confusion matrix, recall, and F1 score.

Accuracy for the Logistic classification model: 0.7236261337730999.

Confusion Matrix :

$$\begin{vmatrix} 8149 & 2239 \\ 3459 & 6770 \end{vmatrix} \tag{7}$$

And also the Classification Report:



```
Classification Report:
              precision    recall  f1-score   support

           0       0.71      0.79      0.75     24328
           1       0.75      0.66      0.71     23778

    accuracy                           0.73     48106
   macro avg       0.73      0.73      0.73     48106
weighted avg       0.73      0.73      0.73     48106
```

Figure 5: Logistic Function Prediction Results

4.2 Bayesian classification

The adoption of a discrete Bayesian model has been selected for the subsequent analysis of the dataset. This Bayesian model imparts a flexible framework for modeling by introducing prior distributions, enabling the comprehensive incorporation of prior knowledge, and facilitating the continuous refinement of estimates for unknown parameters based on observed data [11].

In the Bayesian paradigm, Bayesian networks elucidate the dependency relationships among variables through the interconnected nodes and edges. The initial selection of appropriate prior distributions and the utilization of Bayesian networks are integral to the construction of the model. This method not only provides probability estimates for parameters but also adeptly manages uncertainty.

The extant data was amalgamated and processed as the parent node within the Bayesian network. Three principal sub-nodes, denoted as physiological indicators, medical indicators, and subjective information, were defined. The interrelationships among these nodes were subsequently analyzed leveraging maximum likelihood estimation (MLE).

4.2.1 The Establishment of funcion

Likelihood function:

For each node, write a parameterized likelihood function that represents the probability of observing the data given the network structure:

$$(P(X_i|parents(X_i);\theta_i))$$

In this case, (θ_i) is the parameter of node (X_i), and ($parents(X_i)$) is the parent node of node(X_i). Taking the case of hypertension as an example, the likelihood function can be expressed as:

$$P(Hypertension|Age, BMI, Cholesterol; \theta Hypertension)$$

Log-likelihood function:

To streamline computations, we employ the logarithm of the likelihood function, denoted as $\log(L(\theta|D))$, where θ represents the parameter set for all nodes, and D stands for the observed data. This logarithmic likelihood function will be utilized in the subsequent maximization process for a more efficient estimation of model parameters:

$$log(L(\theta|D)) = \sum ilogP(X_i|parents(X_i); \theta_i) \tag{8}$$

Parameter solving:

The log-likelihood function is maximized by maximizing the log-likelihood function, i.e., solving for :

$$\frac{\partial L(\theta|D)}{\partial \theta} = 0 \tag{9}$$

Finding the set of parameters θ such that the log-likelihood function achieves its maximum value. **Testing, Interpretation, and Prediction:**

Test whether the estimated parameter values are within reasonable ranges and interpret what these parameters mean in the context of the network. Ensure that the fit of the model meets expectations. Finally, test the test set data.

4.2.2 Model Training

Step 1: Load the dataset and split it into training and testing sets.

Use the preprocessed data to analyze the Bayesian network model. Since all the data is discrete and has been cleaned, there is no need to perform data standardization or normalization again.

Step 2: Define the topology of nodes and edges

Create node:

Using discretized feature values as parent nodes and target values as child nodes, extracting nodes as shown in the table below:

Creating Bayesian networks:

The relationship between the common node Cardio and the three indicators is mapped to determine the final output result of whether a person has cardiovascular disease. This relationship is connected to the input variables (i.e., parent nodes).

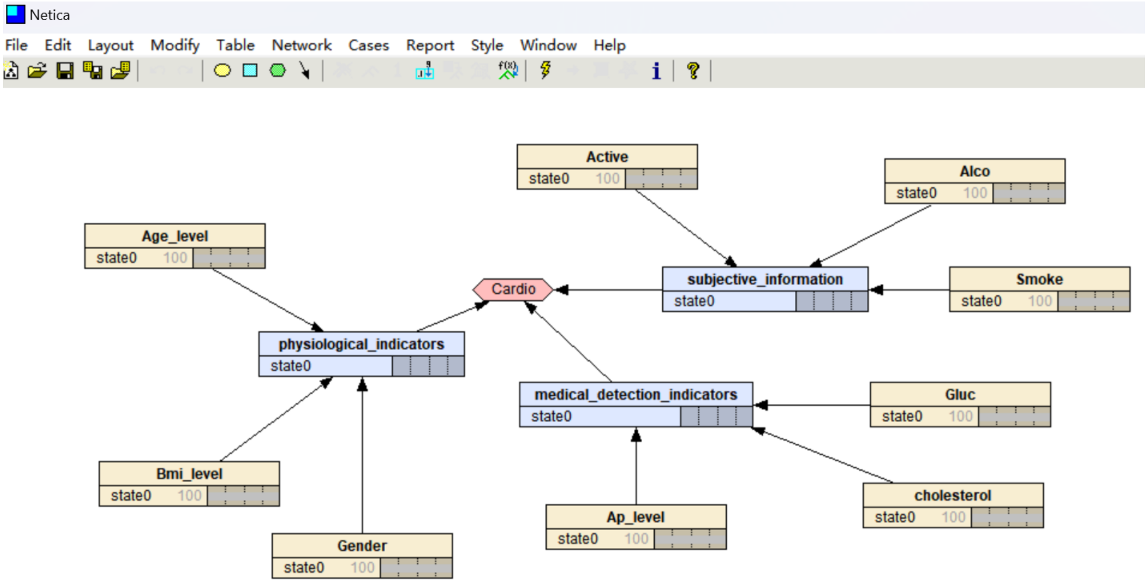


Figure 6: Bayesian Network Structure Diagram

Step3:Calculate CPD (Conditional Probability Distribution)

Applying the Maximum Likelihood Estimation (MLE) algorithm, the conditional probability distribution for each node in the Bayesian network model is calculated, leading to the subsequent table of probability distributions:

Table shows the proportions of cases in each category, broadly reflecting the standard structure of the survey sample and its health. In particular, we found a high percentage of subjects with hypertension. But at the same time most of them are still doing well in terms of smoking, alcohol consumption and exercise.

Step4 Make predictions on a test set and analyze the results

Bringing in the fitted model, the corresponding predicted value is found for each data in the prediction set and compared with the true value to find the predicted probability as follows:

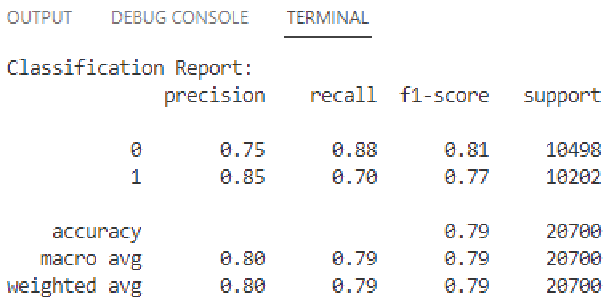
Accuracy for the Bayesian classification model: 0.79304347826086695

Also we get the confusion matrix as follows:

Confusion Matrix :

$$\begin{vmatrix} 9285 & 1213 \\ 3071 & 7131 \end{vmatrix} \tag{10}$$

And also the Classification Report:



```
Classification Report:
              precision    recall  f1-score   support

           0       0.75      0.88      0.81     10498
           1       0.85      0.70      0.77     10202

    accuracy                           0.79     20700
   macro avg       0.80      0.79      0.79     20700
weighted avg       0.80      0.79      0.79     20700
```

Figure 7: Bayesian Network Model Prediction Results

4.3 XGboost Classification

4.3.1 The Establishment of funcion

XGBoost classification model is the abbreviation of "Extreme Gradient Boosting" (Extreme Gradient Boosting), XGBoost algorithm is a class of base functions and weights are combined to form a synthetic algorithm that fits the data well.

For a dataset containing n bars of m dimensions, the XGBoost model can be expressed as:

$$\hat{y}_i = \sum_{i=1}^{k} f_k(x_i), f_k \in (i = 1,2,3\ldots,n) \tag{11}$$

$$F = f(x) = w_{q(x)}(q : R^m \to \{1,2,\ldots T\}, w \in R^T) \tag{12}$$

F is the set of CART decision tree structures, q is the tree structure of samples mapped to leaf nodes, T is the number of leaf nodes, and w is the real fraction of leaf nodes. When constructing the XGBoost model, it is necessary to find the optimal parameters according to the principle of minimizing the objective function in order to build the optimal model.The objective function of the XGBoost model can be divided into the error function term L and the model complexity function term 2.The objective function can be written as follows.

$$Ob_j = L + \omega \tag{13}$$

$$L = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \equiv \omega = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{14}$$

WhereγT represents the L1 regularization term,and$\frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$.

When training the model optimally using the training data, it is necessary to keep the original model unchanged and add - a new function f to the model so that the objective function is reduced as much as possible, the process is as follows:

$$\hat{y}_i^{(0)} = 0 \tag{15}$$

$$\hat{y}_i^{(1)} = \hat{y}_i^{(1)} + f_2(x_i) \tag{16}$$

$$\hat{y}_i^{(2)} = \hat{y}_i^{(0)} + f_1(x_i) \tag{17}$$

......

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i) \tag{18}$$

where $\hat{y}_i^{(t)}$ is the predicted value of the model at the t-th time and $f_t(x_i)$is the new function added at the t-th time. At this time the objective function is expressed as:

$$Obj^{(t)} = \sum_{i=1}^{n} \left(y_i - \left(\hat{y}_i^{(t-1)} + f_i\left(x_i\right)\right)\right)^2 + \omega \tag{19}$$

In the XGBoost algorithm, in order to quickly find the parameters that minimize the objective function, a second-order Taylor expansion of the objective function is performed to obtain an approximate objective function:

$$Obj^{(t)} \approx \sum_{i=1}^{n} \left[\left(y_i - \hat{y}^{(t-1)}\right)^2 + 2\left(y_i - \hat{y}_i^{(t-1)}\right) f_t\left(x_i\right) - h_i f_t^2\left(x_i\right)\right] + \omega \tag{20}$$

When the constant term is removed, it can be seen that: the objective function is only related to the first and second order derivatives of the error function. At this point, the objective function is expressed as:

$$Obj^{(t)} \approx \sum_{i=1}^{n} \left[g_i w_{q(x_i)} + \frac{1}{2} h_i w_{q(x_i)}^2\right] + \gamma T + \frac{1}{2} \sum_{j=1}^{T} w_j^2 \tag{21}$$

$$= \sum_{j=1}^{T} \left[\left(\sum_{i \in I_j} g_i\right) w_j + \frac{1}{2}\left(\sum_{i \in I_j} h_i + \lambda\right) w_j^2\right] + \gamma T \tag{22}$$

If the structural part q of the tree is known, the objective function can be used to find the optimal Wj,and obtain the optimal objective function value. Its essence can be categorized as a quadratic function of the minimum value of the solution problem. Solution:

$$w_j^* = \frac{-\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} \tag{23}$$

$$Obj = -\frac{1}{2} \sum_{j=1}^{T} \frac{\left(\sum_{i \in I_j} g_i\right)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T \tag{24}$$

Obj can be used as a scoring function to evaluate the model, the smaller the value of Obj, the better the model. By recursively calling the above tree building method, a large number of regression tree structures can be obtained, and the optimal tree structure can be searched by using Obj and put into the existing model, so as to build the optimal XGBoost model.

4.3.2 Model Training

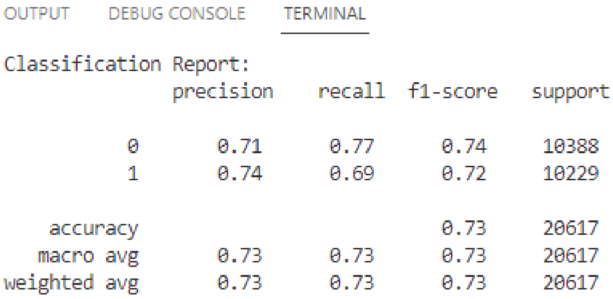
Accuracy for the Bayesian classification model: 0.7283794926516952

Also we get the confusion matrix as follows:

Confusion Matrix :

$$\begin{vmatrix} 7971 & 2417 \\ 3183 & 7046 \end{vmatrix} \tag{25}$$

And also the Classification Report:



```
Classification Report:
              precision    recall  f1-score   support

           0       0.71      0.77      0.74     10388
           1       0.74      0.69      0.72     10229

    accuracy                           0.73     20617
   macro avg       0.73      0.73      0.73     20617
weighted avg       0.73      0.73      0.73     20617
```

Figure 8: XGBoost algorithm Prediction Results

5 Model Evaluation and Further Discussion

To compare the predictive performance of different classifiers, a comprehensive analysis was conducted using accuracy, confusion matrices, ROC curves, and learning curves across three models.

5.1 The contrast of confusion matrix

The Confusion Matrix is a table used to evaluate the performance of a classification model. It presents the relationship between the model's predictions on the test set and the actual labels in matrix form, offering a detailed analysis of the model's performance. By conducting analysis and computations, we obtained the Accuracy and Confusion Matrices for the three models and visualized them in a heatmap as shown below:

- Logistic Regression model accuracy: 0.7236261337730999
- Bayesian Network model accuracy: 0.7930434782608695
- XGBoost model accuracy: 0.7283794926516952

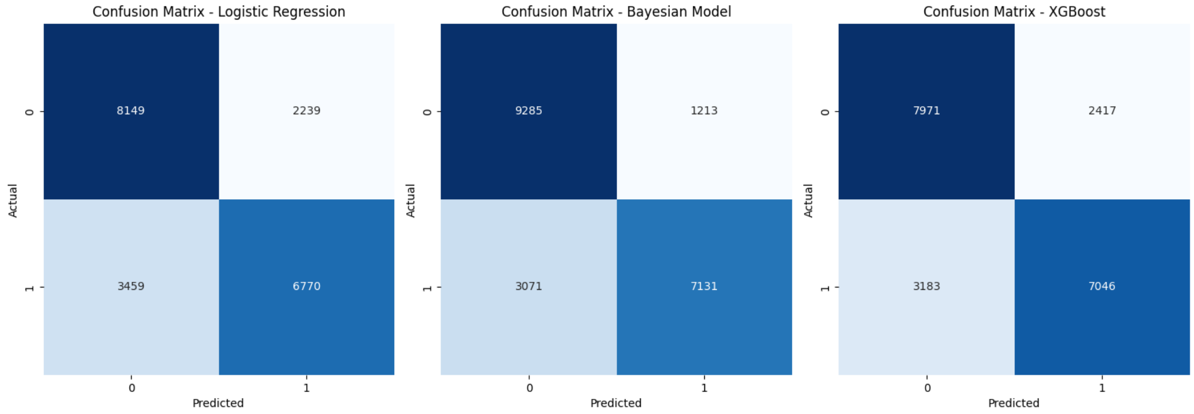


Figure 9: Confusion Matrix Comparison Chart

In continuing the computation of recall and F1 scores for each model, the following conclusions are drawn overall:

-The Bayesian Network model performs the best in terms of accuracy (0.7930), and its confusion matrix and classification report indicate robust predictions for both classes.

-The Logistic Regression model exhibits lower accuracy (0.7236), with a relatively lower recall for Class 1.

-The XGBoost model demonstrates average performance in terms of accuracy (0.7284), with relatively lower precision and recall for Class 1.

5.2 ROC Curve

The ROC curve is a graphical tool used to evaluate the performance of a binary classification model.

It illustrates the relationship between the True Positive Rate (Sensitivity) and False Positive Rate at different thresholds.

When the Area Under the Curve (AUC) is equal to 0.5, it indicates that the model's classification performance is equivalent to random guessing. AUC greater than 0.5 and close to 1 suggests better model performance. The closer AUC is to 1, the better the model performs at different thresholds, and the ROC curve approaches the upper-left corner.

Through analysis and computations, we obtained the ROC curves as depicted in the following graph:

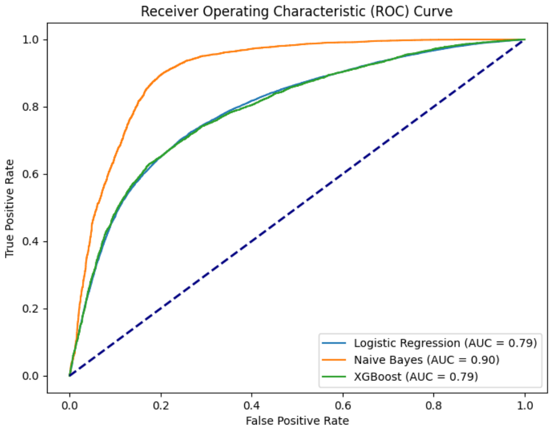


Figure 10: ROC Curve Comparison Chart

From the graph, we can observe that the Bayesian algorithm has the largest area under the curve (AUC), indicating the highest AUC. Therefore, based on the ROC curve, we can conclude that the Bayesian network model exhibits the best fitting performance among the three models.

5.3 Comparative observation of learning curve

The learning curve is a graphical tool used to assess the performance of a model under different amounts of training data. We have integrated the learning curves of the three models and plotted them in the figure below:

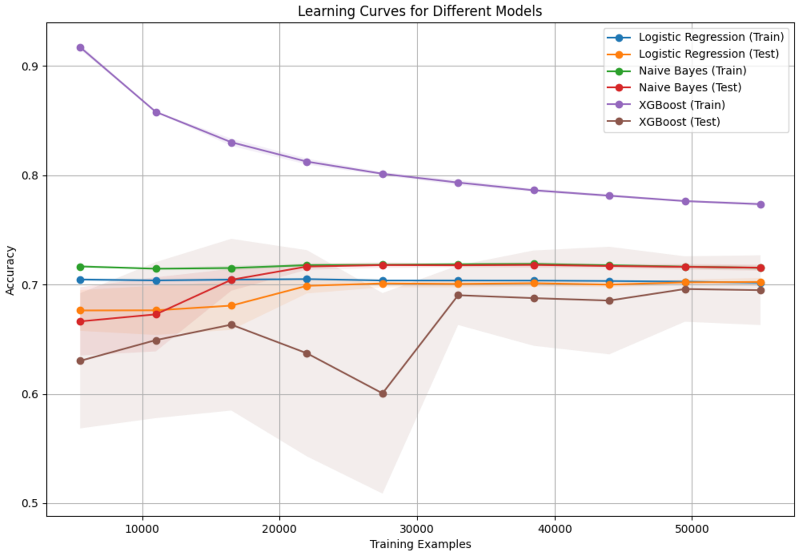


Figure 11: learning Curve Comparison Chart

In the figure, it is observed that the training accuracy of the XGBoost model is the highest, reaching around 80%. However, the predictive accuracy of the XGBoost model is relatively lower. On the other hand, the Bayesian Network model exhibits relatively high and stable accuracy in both training and testing, reinforcing the predictive performance of the Bayesian model.

6 Conclusion

For the establishment of classification models, we established three models, namely, logistic regression model, Bayesian classification model, and XGboost classification model, to analyze the patient's disease condition, and the results obtained are as follows:

- The Bayesian classification model has the best fitting effect, with an accuracy of 0.793, and its model excels at dealing with high-dimensional data and can be subjected to incremental learning, which means that the model can be gradually updated and quickly adapted to the new training data. Its model is good at handling high-dimensional data and can learn incrementally, so that the model can be updated gradually and quickly adapt to new training data. However, the model has some limitations: it needs to estimate the a priori probability, which may require a large amount of training data to obtain accurate estimation results. At the same time, the model has high computational complexity for large-scale datasets.

- The logistic regression model has the lowest fitting effect, the accuracy can reach 0.723, the model output results with probabilistic interpretation: logistic regression can output the probability that the sample belongs to a certain category, which is easy to understand and interpret the results of the model and has stability, and is insensitive to small noise. At the same time, the model has the problem of multicollinearity: if there is a high correlation between the features, the effect of the logistic regression model may be affected by the problem of multicollinearity.

- The fitting effect of XGboost classification model is next to that of the XGboost classification model, and the accuracy can reach 0.728, which is a very strong robust model, and the XGBoost can automatically deal with the missing values and is relatively insensitive to the outliers, and at the same time the parameter tuning is more complex but at the same time the performance of XGBoost is highly dependent on the choice of parameters and requires careful parameter tuning, which may take more time and computational resources.

References

[1] Xu, Y., & Meng, L. (2022). Deconstruction of Risk Prediction of Ischemic Cardiovascular and Cerebrovascular Diseases Based on Deep Learning. Contrast Media & Molecular Imaging, 2022, 8478835. https://doi.org/10.1155/2022/8478835

[2] Zhou, Q.,& Bei, Y. (2020). Editorial: Gender Differences in Cardiovascular Diseases. Journal of Cardiovascular Translational Research, 13(1),1-2. https://doi.org/ 10.1007/s12265-020-09956-9

[3] Teo, K. K.,& Rafiq, T. (2021). Cardiovascular Risk Factors and Prevention: A Perspective From Developing Countries. The Canadian Journal of Cardiology,37(5),733-743. https://doi.org/10.1016j.cjca.2021.02.009

[4] Sacramento-Pacheco, J., Sánchez-Gómez, M. B., Gómez-Salgado, J., Novo-Muñoz, M. M.,& Duarte-Clíments, G. (2023). Prevalence of Cardiovascular Risk Factors in Spain: A Systematic Review. Journal of Clinical Medicine,12(21),6944.https://doi.org/10.3390/jcm12216944

[5] Agewall, S.(2024). Cardiovascular prevention and risk factors. European Heart Journal. Cardiovascular Pharmacotherapy, 10(1),1-2.https://doi.org/10.1093/ehjcvp/pvad087

[6] Kee, O. T., Harun, H., Mustafa, N., Abdul Murad, N. A., Chin, S. F., Jaafar, R., & Abdullah, N. (2023). Cardiovascular complications in a diabetes prediction model using machine learning: A systematic review. Cardiovascular Diabetology, 22(1), 13. https://doi.org/10.1186/s12933-023-01741-7

[7] E.Cuadrado-Godia,P.Dwivedi,S.Sharma et al.,Cerebral small vessel disease: a review focusing on pathophysiology, biomarkers,and machine learning strategies,Journal of Stroke,vol.20,no.3,pp.302320, 2018.

[8] H. Wang,E.Pujos-Guillot,B.Comte et al., Deep learning in systems medicine,Briefings in Bioinformatics,vol.22,no.2,pp.15431559,2020.

[9] D.Scarafoni, B.A.Telfer, D.O.Ricke, J.R..ornton,and J.Comolli," Predicting influenza A tropism with end-to-end learning of deep networks,"Health Security,vol.17,no.6,pp.468476,2019.

[10] Regularization linear regression generating method, involves using verification sub-sets to evaluate regularized linear regression models to generate evaluation score of regularized linear regression models-All Databases.(n.d.).

[11] Ordovas, J. M., Rios-Insua, D., Santos-Lozano, A., Lucia, A., Torres, A., Kosgodagan, A., & Camacho, J. M. (2023). A Bayesian network model for predicting cardiovascular risk. Computer Methods and Programs in Biomedicine, 231, 107405. https://doi.org/10.1016/j.cmpb.2023.107405

Appendices

Appendix A: remaining figure

Here are remained figure we used in our model as follow.

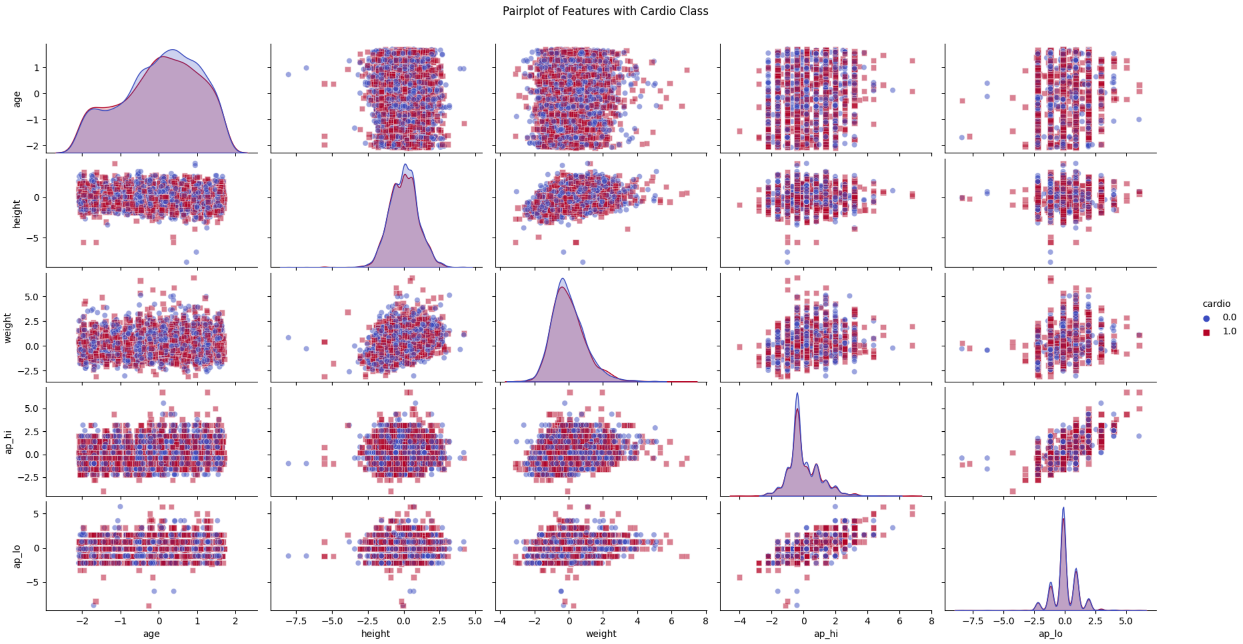


Figure 12: learning Curve Comparison Chart

Appendix B: Programmes Codes

Here are simulation programmes we used in our model as follow.

Input python source:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

data = pd.read_csv("C:/Users/92579/Desktop/MATH/2024-MCM/Problem/Problem/cardio_train.cs

features = [’age’, ’height’, ’weight’, ’ap_hi’, ’ap_lo’, ’gender’, ’cholesterol’, ’gluc’

data_encoded = pd.get_dummies(data[features])

X = pd.concat([data_encoded, data.drop(features, axis=1)], axis=1)
y = data[’cardio’]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42

nb_model = GaussianNB()
nb_model.fit(X_train, y_train)
```

```
predictions = nb_model.predict(X_test)
```

Input python source:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
from tabulate import tabulate


data = pd.read_csv("C:/Desktop/MATH/2024-MCM/Problem/Problem/output.csv")

X = data.drop('cardio', axis=1)
y = data['cardio']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42

scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

model = LogisticRegression(random_state=42)
model.fit(X_train_scaled, y_train)
```

Input python source:

```
import pandas as pd
import xgboost as xgb
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

data = pd.read_csv("C:/Desktop/MATH/2024-MCM/Problem/Problem/output.csv")

X = data.drop('cardio', axis=1)
y = data['cardio']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42

model = xgb.XGBClassifier(objective="binary:logistic", random_state=42)

model.fit(X_train, y_train)

y_pred = model.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
```

Report on Use of AI

1.OpenAI ChatGPT (Jan 8, 2024 version, ChatGPT-4,)
Query1: <Try to explain to me the plausibility of the following assumptions:Data with diastolic blood pressure greater than systolic blood pressure are assumed to be outliers.>
Output1: <The rationale behind assuming that data with diastolic blood pres- sure greater than systolic blood pressure are outliers is that it contradicts the physio- logical norm. In a healthy individual, systolic blood pressure should be higher than diastolic blood pressure>
Query2: <he physiological indicators of height and weight are both limited to a fixed range, where the BMI index for both should be above 18.5, while severe obesity (BMI > 32) is allowed>
Output2: <he rationality behind this assumption is that the BMI index, which is calculated based on height and weight, has established ranges that corre- spond to different categories of weight. A BMI below 18.5 indicates underweight, while a BMI above 32 denotes severe obesity>