SCIPEDIA

Study on Basic Machine Learning Techniques to Detect and Classify Cardiovascular Diseases

Yahia Z. Rawash[1], Hua-Nong Ting[1,*] and Kok Han Chee[2]

[1] Faculty of Engineering, Department of Biomedical Engineering, University of Malaya, Kuala Lumpur, Malaysia

[2] Faculty of Medicine, Department of Medicine, University of Malaya, Kuala Lumpur, Malaysia

INFORMATION

Keywords:
Machine learning
heart disease
cardiovascular disease
K-nearest neighbor
XGBoost
random forest
GaussianNB
accuracy

DOI: 10.23967/j.rimni.2026.10.74340

SCIPEDIA Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería

RIMNI Revista Internacional Métodos numéricos para cálculo y diseño en ingeniería

Study on Basic Machine Learning Techniques to Detect and Classify Cardiovascular Diseases

Yahia Z. Rawash[1], Hua-Nong Ting[1,*] and Kok Han Chee[2]

[1]Faculty of Engineering, Department of Biomedical Engineering, University of Malaya, Kuala Lumpur, Malaysia

[2]Faculty of Medicine, Department of Medicine, University of Malaya, Kuala Lumpur, Malaysia

ABSTRACT

The heart is an essential organ required to maintain the general health of individuals. Cardiovascular diseases (CVDs) have become the leading cause of death globally, replacing cancer and diabetes. Computer-based techniques have made it easier for physicians to diagnose various cardiac conditions, including heart failure. We are currently in the "information age," a period characterized by the generation of millions of bytes of data every day. By applying ML algorithms' techniques, such as Random Forest (RF), XGBoost, KNN, and GaussianNB, we can evaluate and compare the performance of machine learning classifiers for heart disease prediction and transform these data into information for the estimation of heart disease. The World Health Organization has estimated that in 2019, cardiac disease was responsible for 32% of all deaths worldwide. In this paper, we use a public data set (Indicators of Heart Disease) and hyperparameters to develop four classifiers—the Random Forest, XGBoost, KNN, and GaussianNB—and compare their performance. Based on the trial data, XGBoost was the best, with an accuracy of 91.63%, a precision of 94.40%, a recall of 88.50%, an F1-score of 91.36%, a specificity of 94.75%, and an AUC score of 97.39%. This study showcases the accuracy of machine learning systems in predicting cardiac conditions and can serve as a foundation for developing a decision-support tool aimed at detecting and preventing heart disease in its early stages.

OPEN ACCESS

Received: 09/10/2025

Accepted: 20/01/2026

DOI
10.23967/j.rimni.2026.10.74340

Keywords:
Machine learning
heart disease
cardiovascular disease
K-nearest neighbor
XGBoost
random forest
GaussianNB
accuracy

1 Introduction

For the past 15 years, cardiovascular diseases (CVD) have been the top reason of death. If one were to focus on country like India, the World Health Organization estimates that heart-related illnesses cost the nation $237 billion between 2005 and 2015 [1]. According to the European Public Health Alliance (EPHA 2010) [1], strokes, heart attacks, and other circulatory illnesses account for 41% of all fatalities. According to projections made by the Australian Institute of Health and Welfare (AIHW), cardiovascular disease (CVD) would account for 42% of all mortalities in Australia in 2018 [2].

Many symptoms can indicate heart disease, making a quicker and more precise diagnosis difficult. Using patient databases for people with heart illness is similar to working with a real-world scenario.

*Correspondence: Hua-Nong Ting (tinghn@um.edu.my).

More weight is assigned to the attribute that significantly affects the prediction of disease. Therefore, it would seem reasonable to think about using the expertise of different specialists that has been compiled in datasets to assist with the diagnosis procedure.

Additionally, it gives medical practitioners another information source to select from. The main organ in the human body is the heart. There are two types of risk factors for heart disease. It may or may not be possible to manage heart problems. According to clinical research, uncontrollable conditions raise a person's risk of heart disease (HD), or more precisely, cardiovascular disease (CVD). These risk factors include, but are not limited to, a household history of cardiovascular disease (CVD), high rates of LDL (bad cholesterol), low levels of HDL (good cholesterol), obesity, hypertension, and a high-fat diet [3].

Alcohol consumption, obesity, smoking, high blood pressure, and high cholesterol are among the heart disease risk factors that people may control. In order to determine a diagnosis, doctors typically consider a patient's current health status, past diagnoses of individuals with similar conditions, and other variables. Heart disease risk factors are shown in Fig. 1 [4].

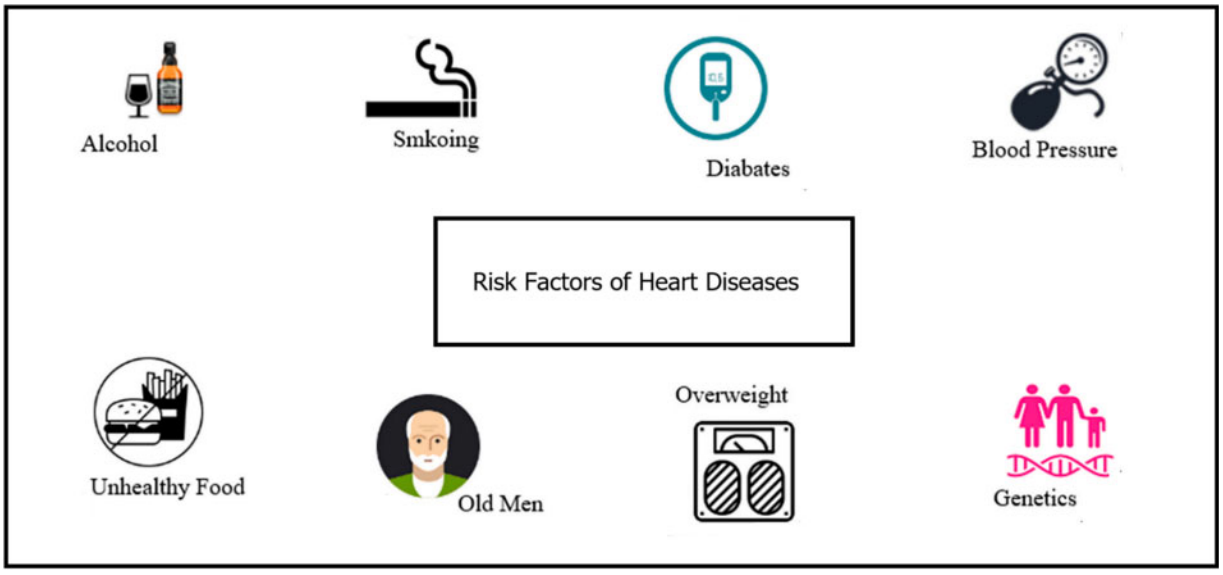


Figure 1: Risk issues for heart diseases

There are many different types of cardiovascular diseases; Reference [5] lists some of them. Coronary Heart Disease: Infection or destruction of the main blood vessels. Cardiomyopathy: An genetic or acquired condition that affects the heart muscles. Ischemic Heart Disease (IHD): cardiac artery narrowing that keeps the heart muscles from receiving enough blood and oxygen causes cardiac issues. A chronic condition known as heart failure occurs when the heart is unable to pump blood as quickly as it should. High blood pressure and cardiac issues are symptoms of hypertensive heart disease. Heart diseases or disorders caused by bacteria or viruses are referred to as inflammatory heart disease (IHD). Valvular heart disease is the result of a damaged or defective heart valve.

Related Work

Medical science is currently using a number of applications for data analysis and innovation that use various machine learning methods. Cases of the application of machine learning methods in healthcare research have been found in recent studies. These include the use of MRIs for tumor recognition [6,7], the recognition of COVID-19 using X-rays [8,9], the predictability of cardiac conditions [10,11], dengue viruses [12,13], stroke [14], and cancer illnesses [15].

Maini et al. [16] in their work utilize ANN methodology which needs to be utilized while developing a system for the prediction of cardiovascular disease. The 76 features of the Cleaveland data for cardiac disease will be collected. Dimensionality reduction is needed so that the data can be handled efficiently. The mining algorithms will be used once the data has been cleaned extensively. Accuracy, precision, recall, and f1 score metrics will be employed to evaluate the performances of the algorithms. The best-performing algorithms will also be combined further to maximize performance. Cross-validation will be performed using the k-fold cross-validation concept. Their research attained 90.74% accuracy.

A new machine learning method of predicting heart disease is introduced in the proposed work (Kavitha et al. [17]). The Cleveland heart disease data set was used in the projected work, and regression and classification data mining algorithms were used. Machine learning algorithms random forests and Decision trees are used. The machine learning model's new process is designed. Three machine learning algorithms—Random Forest, Decision Tree, and Hybrid model—are used during the execution. Random Forest and Decision Tree are used simultaneously. Based on experimental findings, the hybrid heart disease prediction model is 88.7% accurate.

Heart disease is discussed along with its hazards, and Katarya and Meena [18] provide an overview of machine learning (ML) techniques. They forecasted cardiovascular disease (CVD) and provided an assessment of the supervised learning reproductions used throughout the diagnosis test by using such machine learning techniques. Compared to other classifiers, the Logistic Regression classifier in this suggested model exhibits a superior accuracy of 93.40%.

Some features pertaining to heart disease and an algorithmic model of supervised learning based on Naïve Bayes, decision tree, K-nearest neighbors, and the random forest algorithm are shown here (Shah et al. [19]). This is based upon the Cleveland database, which is a UCI collection of patient data with cardiac disease. It has 76 features and 303 instances within the data set. Only 14 of these 76 features are chosen for testing, which is extremely important to verify the efficiency of different algorithms. The purpose of this research is to predict the risk of patients having heart disease. From the results, K-nearest neighbor has the highest accuracy rate of (90.79%) with lower fault rates.

In order to compare and analyze the findings of a Supervised Learning Record Coronary Heart Illness database, Bharti et al. [20] set up different machine learning (ML) models and deep learning methods. The 14 essential traits that could be applied in the study are included in this compilation. The overall average accuracy of the deep-learning method is 94.2%. Parashar et al. [21] produced quick and precise computerized coronary heart disease finding using the XGBoost boosting algorithms and SVM classification. They trained and tested using the random-forest technique. The study's findings demonstrated that the N2Genetic-nuSVM, that is composed of the extremely carefully curated Z-Alizadeh Sani database, has a 93.08% accuracy rate in predicting diagnoses of medical cardiovascular illness.

A model with two SVM was developed by Ali et al. [22] to accurately diagnose heart disease. The first SVM was used to remove features that weren't needed, and the second SVM was used to make

predictions. Compared to the conventional SVM model, the hybrid grid search algorithm (HGSA) produced 3.3% greater exactness.

A prototype containing the Random Forest System (RFS) and the Random Search Algorithm (RSA) was advocated by Javeed et al. [23] for testing and training datasets. The results exposed that the RSA-RF system outperformed a random forest model by 3.3%. The suggested approach improved training accuracy while achieving a 93.33% classification accuracy.

Decision trees and Nave Bayes machine learning algorithms were employed by Santhana Krishnan and Geetha [24] to forecast heart attacks. The Decision Tree Model's accuracy level was 87%, whereas Nave Bayes' precision rate was 91%. They assumed that a decision tree was the optimal approach for managing sets of data.

Rabbi et al. [25] in a bid to dramatically improve the accuracy of heart disease prediction, the research establishes an ensemble method that integrates logistic regression, decision trees, random forests, support vector machines, K-nearest neighbors, and Gaussian naive Bayes. For making early and accurate predictions of heart disease, we also use advanced ensemble learning methods such as stacking, bagging, voting, and boosting. Three data sets—Framingham Heart Disease, Indicators of Heart Disease Dataset (2020), and Cleveland Heart Disease—are applied in performance testing in order to support a good validation of our methodologies. From the results, the bagging ensemble machine learning algorithm had a 97% accuracy on Framingham Heart Disease and Indicators of Heart Disease (2020), and the voting ensemble machine learning algorithm (Bagging Decision Tree) had 92% accuracy on the CHD.

To forecast the risk of heart disease, in [26], ensemble machine learning model has been proposed based on quantum machine learning classifiers. The used method was a bagging ensemble model with a quantum support vector classifier as a base classifier. In addition, the SHapley Additive exPlanations (SHAP) model is employed to calculate and display each feature's weight in the prediction to make it interpretable. Comparative and contrast experimental work is performed with the traditional machine learning classifiers like Support Vector Machine (SVM) and Artificial Neural Network (ANN) with other quantum classifiers one by one, i.e., Quantum Support Vector Classifier (QSVC), Quantum Neural Network (QNN), and Variational Quantum Classifier (VQC). Experimental results of the Cleveland dataset show QSVC to be superior to the others and thus utilized in the bagging model proposed. Bagging-QSVC with an accuracy rate of 90.16% is significantly superior to all the above classifiers and is extremely competitive against most state-of-the-art models that utilize the same data.

Das et al. [27] tested diagnosing cardiac disease with different techniques. Some of the techniques that were used were extreme gradient boosting machine (XGBoost), bagging, random forest, decision tree, K-nearest neighbor, and naive bayes. They processed the Kaggle dataset "Key Indicators of Heart Disease" with 319,795 cases and over 300 features, of which 18 were selected for processing. Among the features, four were decimal features, five were strings, and nine were Boolean. There were also some data processing techniques that were employed, such as data cleaning, duplicates removed, and translation of categorical variables. 80%–20% ratio was employed for training and testing, respectively. Comparisons were made through several evaluation metrics such as area under the curve (AUC), F1-score, accuracy, sensitivity, and precision. With an accuracy rate of 91.30%, the Xgboost model demonstrated optimal performance outcomes.

Jahed et al. [28] employed Random Forest, Decision Tree, and Stochastic Gradient Descent (SGD) methodologies. Kaggle Indicators of Heart Disease data was employed in testing these methods. A few tests with respect to partitioning the data on race and sex indicated that dividing the data according to these factors enhanced accuracy. They obtained a 76% SGD accuracy.

Based on certain lifestyle data, we have developed a prediction method in this study that uses the Extreme Gradient Boost (XGBoost) Classifier in conjunction with ADASYN SMOTE to forecast the likelihood of developing heart disease [29]. The model's accuracy on the Personal Key Indicators of Heart Disease dataset, which is based on recent data gathered by the Centers for Disease Control and Prevention, was 94.7%. The dataset contains the data of around four lakh individuals who have disclosed certain fundamental health-related information, such as having had myocardial infarction (MI) or coronary heart disease (CHD). Using balanced data, the XGBoost classifier achieves a 94.7% accuracy rate.

The models [30] were created to lower the chance of acquiring serious illnesses by assisting medical professionals in the early diagnosis and prognosis of individuals with heart disease. The Jupyter Notebook Web application serves as the foundation for the developed heart disease risk assessment model. Unbiased metrics like true positive rate, true negative rate, accuracy, precision, misclassification rate, area under the ROC curve, and cross-validation are used to calculate the model's performance. The ensemble heart disease model works better than the other suggested and used models, according to the findings.

The goal of this study [31] is to evaluate the data on cardiovascular illnesses that is now available in order to anticipate heart disease early on and stop it from happening. The Indian state of Jammu and Kashmir provided the dataset of heart disease patients, which was then kept on the cloud. In order to forecast heart illnesses, stored data is first pre-processed and then further examined using machine learning algorithms. The dataset was analyzed using a variety of machine learning techniques, including Random Forest, Decision Tree, Naive based, K-nearest neighbors, and Support Vector Machine. The results showed the performance metrics (F1 Score, Precision, and Recall) for each technique, demonstrating that Naive Bayes is superior without parameter tuning while Random Forest algorithm proved to be the best technique with hyperparameter tuning.

In order to predict the vulnerability of cardiac problems, Gavhane et al. [32] focused on machine learning applications by utilizing several data including lifetime, gender, heart rate, etc.

Neural network supervised algorithms, such as the multilayer perceptron (MLP) that produces consistent results from user input, were utilized to train and evaluate the dataset. Neural network-based machine learning algorithms are the most dependable and accurate algorithms.

The following are the key aids of the research:

- The Centers for Disease Control (CDC)—Indicators of Heart Disease dataset statistics on heart disease were utilized in this study to estimate the efficiency of the model.
- To get the best accuracy, preprocessing techniques and feature selection are applied.
- The XGBoost classifier outperforms the other three categorization methods used to forecast cardiac risk conditions in terms of accuracy and performance.

2 Methodology

In this part of the study, we put into practice the general procedure shown in Fig. 2 for obtaining the CVD estimation using ML models. The four primary processes in this workflow are data preparation, training models, model evaluation, and data collecting.

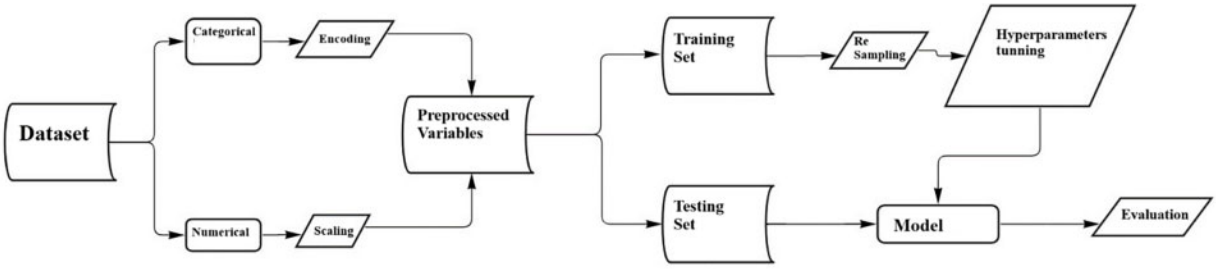


Figure 2: An outline of machine learning process cast off in this project

2.1 Dataset Description

The Kaggle repository provided the public dataset that was utilized in this investigation. The Behavioral Risk Factor Surveillance System [33] of the Centers for Disease Control and Prevention is where it first originated. This system involved telephone surveys about the health status of US individuals and had over 300 characteristics. It has been cleaned and preprocessed. In order to use the smaller data set with 18 features in this work. The dataset, which has 319,795 records without null values and is classified into 13 category variables, 4 numerical variables, and one categorical target ('HeartDisease') is highly unbalanced. And it is available on line on Kaggle repository. The common category for each category feature is shown in Table 1.

Table 1: Dataset attribute

No.	Attribute (A)	Value (V)	Type (T)
1	Heart Disease	1: Yes 2: No	Nominal
2	BMI	12–94.8	Numerical
3	Smoking	1: Yes 2: No	Nominal
4	Alcohol Drinking	1: Yes 2: No	Nominal
5	Stroke	1: Yes 2: No	Nominal
6	Physical Health	1–30	Numerical
7	Diff. Walking	1: Yes 2: No	Nominal
8	Sex	1: Male 2: Female	Nominal
9	Age	1: 18–24 2: 25–29 3: 30–34 4: 35–39 5: 40–44 6: 45–49 7: 50–54 8: 55–59 9: 60–64 10: 65–69 11: 70–74 12: 75–79 13: 80 or older	Numerical
10	Race	1: White 2: Hispanic 3: Black 4: Other 5: Asian 6: American Indian/Alaskan Native	Nominal
11	Diabetic	1: Yes 2: No	Nominal
12	Physical Activity	1: Yes 2: No	Nominal
13	Gen. Health	1: Excellent 2: Very Good 3: Good 4: Fair 5: Poor	Nominal
14	Sleep Time	1–24	Numerical
15	Asthma	1: Yes 2: No	Nominal
16	Kidney Disease	1: Yes 2: No	Nominal

(Continued)

Table 1 (continued)

No.	Attribute (A)	Value (V)	Type (T)
17	Mental Health	0–30	Numerical
18	Skin Cancer	1: Yes 2: No	Nominal

Fig. 3 shows some target connections with some variable like age, Race, stroke and alcohol drinking. We are aware that there have been 319,795 observations, of which 292,422 are in the negative class. It is crucial to balance the data as a result.

(a) heart disease with percentage of Alcohol Drinking

(b) heart disease with percentage of Diabetic value

(c) the heart disease count for alcohol drinking

(d) the heart disease count for stroke

(e) the heart disease count for age

(f) the heart disease count for Race

Figure 3: Target connection with other variables

2.2 Data Pre-Processing

Given that the dataset is unbalanced and we are working with both numerical and categorical variables, we need to do at least three pre-processing tasks: Apply One-Hot encoding to the variables that are categorical. Apply conventional scaling to variables with numerical values. To balance classes, we employ both oversampling and under-sampling.

We choose to carry out the first step of the feature preparation process using the Python OneHotEncoder and StandardScaler sklearn pre-processing package. Next, we utilized RandomOverSampler and RandomUnderSampler to resample in order to balance the classes. Lastly, we divided the data into testing and training sets.

Once we were aware of the variable data types, we built a pipeline. Using a conventional scaler, we first normalized all numerical features. Next, we employed One-Hot encoding for categorical values. Finally, we used a label encoder for the target feature. There was no need to impute null values because there were no empty values.

The database was then separated between 80% training data and 20% testing data. It is noteworthy to mention that we used data stratification to guarantee that all goal values were evenly distributed across the training and testing sets. Since the ratio of the negative and positive classes was initially 9 vs. 91, the third stage involved balancing the target feature. To obtain a relationship of 70–30, we randomly over-sampled, and to obtain a 50−50 ratio, we randomly under-sampled. The train set alone was in equilibrium.

Table 2 shows the percentage and the number of training and testing data before (Same-samples) and after sampling (Over-sampling and Under-sampling)

Table 2: Number of training and testing dataset for same, over and under sampling

	Training data	Testing data	Total
Percentage	80%	20%	100%
Number of data with Same data balance	255,836	63,959	319,795
Number of data with Over sampling data	467,875	116,969	584,814
Number of data with Under sampling data	43,796	10,950	54,746

2.3 Model Evaluation

We made the decision to evaluate four distinct model classifiers: XGBoost, KNN, GaussianNB, and Random Forest. We first used Python to run the hyper-parameter optimization process in order to control the optimal procedure. Next, based on the accuracy measure (1), the best classifier is chosen, with TP, TN, FP, and FN standing for true positive, false negative, false positive, and false negative, respectively:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

In order to automate hyper-parameter, search more effectively, we choose to use Python libraries like NumPy, Scikit-Learn, Seaborn, Pandas, and Matplotlib as well as GridSearchCV for hyper-parameter tuning. Python is an open-source framework for hyper-parameter optimization.

The fact that this program can automatically search for the best hyper-parameters utilizing Python conditionals, loops, and syntax is one of the main reasons we chose to utilize it. Additionally, it makes use of cutting-edge algorithms that facilitate quick findings by effectively searching wide areas and eliminating unsuccessful trials. In order to choose which hyper-parameters to assess in the genuine objective function, the basis use Python optimization to create a probability model of the wrapped objective function the optimal parameters are shown in Table 3.

Table 3: Optimized hyper-parameters for the four models

Model	Hyperparameters
XGBoost	max_depth = 7, n_estimators = 300,learning_rate = 0.2
KNN	n_neighbors = 3
GaussianNB	Not available
Random Forests	n_estimators = 10, random_state = 42

In the matrix, samples in a projected class are characterized by each column, whereas examples in a real class are characterized by each row. An examination of a classification model's confusion matrix is presented in Table 4. A description of some of the measurements is given below.

- Accuracy is considered by dividing the number of occurrences that the model correctly identified or assessed as equal by the total sample size.
- Recall shows the percentage of real-time, legitimate scenarios that the model correctly identified or predicted as positive.
- Precision is the percentage of patients who are anticipated to be diagnosed but don't mainly happened.
- The F1-Score needs to be defined as the accuracy and recall of such a model's harmonic average.
- Particulars: The ratio of newly classified healthy individuals to all healthy individuals is known as specificity. It indicates that the individual is healthy and the forecast is incorrect.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = Sensitivity = \frac{TP}{TP + FN} \tag{3}$$

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{4}$$

$$Specificity = \frac{TN}{TN + FP} \tag{5}$$

- AUC–ROC curve: Plotting True Positive Rate (TPR) and False Positive Rate (FPR) on the Y and X axes, respectively, yields the receiver operating characteristic curve (ROC). Plotting the ROC curve involves calculating the TPR and FPR at numerous verges and displaying the results on a graph to determine the optimal threshold for the model.

$$TPR = TP/(TP + FN) \tag{6}$$

$$FPR = FP/(FP + TN) \tag{7}$$

The area under the receiver operating characteristic curve, or the AUC-ROC curve, is a statistical plot that describes a range of values between 0 and 1, hence giving a measure of the model's ability to discriminate between different levels of the threshold. Typically, it is observed that a higher value of AUC is indicative of a higher performance of the model under consideration, with an indication close to 1 for an exemplary discrimination ability across different classes and indicating an extremely poor discrimination power across the respective classes for values close to 0.5. These are referred to as TP (True Positive) wherein the predictive model correctly identifies and detects the occurrence of a particular cardiac condition, thereby proving to be effective in clinical diagnosis. On the other hand, TN (True Negative) refers to those cases when the model correctly identifies an taget as normal, thereby proving to be effective in ruling out cardiac abnormality. In addition, FP, or False Positive, are the times when the model incorrectly diagnoses the presence of a particular cardiac condition when it is not present, pinpointing a critical area of concern regarding diagnostic accuracy. FN, or False Negative, are the times when the model does not detect an existing cardiac condition even when it does, thus pinpointing the possible dangers in cases of non-detection.

Table 4: The confusion matrix elements

	Negative (0)	Positive (1)	
Negative (0)	TN—True Negative	FP—False Positive	Projected
Positive (1)	FN—False Negative	TP—True Positive	

Note: True Positive (TP): The patient has the disease, according to the positive test results. False Positive (FP): The test findings are positive even though the patient is in good health. True Negative (TN): The test results are negative and the patient is well. False Negative (FN): Although the test result was negative, the patient still has the disease.

Binary splits are used by Random Forests, a group of regression and classification trees, to make predictions [34]. A machine learning system that mixes several decision trees is known as random forest. Random features and a randomly chosen dataset are used to train individually tree in a random forest. The last estimate is then derived by combining the guesses of the various trees. As a result, Random Forest (RF) is regarded as one of the most effective algorithms [35]. The algorithm's capacity to manage huge datasets is another advantage [36–38].

2.4 Implementation

Since certain implementation details were covered in the methodology section before. Python is the preferred language for this project. Python code is being executed within the Anaconda Navigator's Jupyter notebook. Jupyter notebook significantly outperforms Python IDE solutions like PyCharm and Microsoft Visual when it comes to designing machine learning algorithms.

One advantage of the Jupyter notebook is that it makes data examination and graph creation—like heatmaps and column figs of related matrices—simple when script code. The phases of putting them into practice consist of:

a. Gathering data.
b. Imported resources: Matplotlib, Warnings, Seaborn, OS, and Scikit-Learn public library were utilized.
c. Understanding study data in order to learn more about the information.

d. Data cleaning and processing: A examination for null and junk values is done using the Is Null() and isna() methods. Throughout the preparation phase, feature engineering was used to the dataset. The Pandas library's get-get dummies function was utilized to transform grouping data into numerical variable quantity. There are a rare classes variable quantity in every database.

e. Features grading: This method normalizes experimental data by using the Standard Scalar and fit transform functions from the scikit-learn compendium.

f. A model decision: People go forward by subtracting x from y. In experimental datasets, the features or involvement elements are characterized by x values, while the reliant on or objective variables—that are essential for illness estimate—are characterized by y values. Create train and test divisions for the x and y data using the "train-test split" technique from the sklearn compendium. The model collection function from the sklearn compendium was imported in order to accomplish this. Information testing and training organizations are being established. created a learning time, accuracy, and confusion matrix for every machine learning algorithm.

g. Using the model with maximum accuracy in its deployment.

3 Results

This part presents the data analysis, analysis of model, comparative analysis with previous efforts, and system application. Because the dataset is in tabular format, it is difficult to notice and understand the dataset in this or any other formula. Consequently, the data is being displayed. The accompanying Fig. 3a–f show a graphical illustration of the amount of entities that are predicted and disconnected by arithmetical features that are explicitly targeted, feature selections, and feature categories. Knowing the trend of the data is helpful. The data is realistically characterized in this approach of data picturing. Pie charts and bar charts are used to illustrate the cleaned and preprocessed data that was acquired throughout this inquiry. The data attributes' activities are illustrated. Through graphical representation, the intricate relationship between the qualities is made easy to understand. A variety of dataset attributes are presented by elements. A frame containing numerous histograms is produced via the [dataset.hist ()] function.

Table 5 shows the accuracy, Sensitivity and Specificity for different tested model, the results show the superior of XGBoost classifier on other type of models like gaussianNB, KNN, and Random Forest. The accuracy for XGBoost is 91.63%, followed by KNN with accuracy of 88.27%, then followed by Random Forest classifier with accuracy of 76.95% finally the gaussianNB with least accuracy of 72.44%.

Table 5: Accuracy value for different tested models

Model	Random forests	GaussianNB	KNN	XGBoost
Accuracy	0.7695	0.7244	0.8827	0.9163
Sensitivity/recall	0.7304	0.8446	0.9404	0.8850
Specificity	0.8085	0.6043	0.8250	0.9475

XGBoost performed better than the other classifiers primarily because it is based on gradient boosted decision trees with powerful regularization techniques, which makes it capable of detecting complex and non-linear relationships between features and target variables without being prone to overfitting problems. The robust capabilities of XGBoost to deal with missing values, give higher

importance to difficult-to-predict examples, and adjust most of its hyperparameters may result in its superior predictability power for most types of data. Nevertheless, XGBoost may also have some drawbacks because it may need careful hyperparameter adjustment and may become computationally inefficient for large datasets.

Table 6 shows the metric parameters for best model with high accuracy the XGBoost classifier the accuracy, precision, recall, AUC and F1-score all have value with 88.5% and more. Which shows the efficiency of XGBoost model for detecting heart disease. Data traffic is being monitored. The data distribution plays a critical role in prediction and problem classification. The data showed that a large percentage of cases had cardiac disease, compared to a low percentage of cases that did not. Thus, in order to avoid overfitting, a balanced dataset is crucial. makes it possible for the algorithm to differentiate between dataset patterns that cause heart disease and those that don't.

Table 6: Evaluation values for classifiers for oversampling

Model	Accuracy	Precision	Sensitivity/Recall	Specificity	F1-score	AUC scores
XGBoost	0.9163	0.9440	0.8850	0.9475	0.9136	0.9739
KNN	0.8827	0.8431	0.9404	0.8250	0.8891	0.9306
Random Forest	0.7695	0.7923	0.7304	0.8085	0.7601	0.8524
Gaussian NB	0.7244	0.6809	0.8446	0.6043	0.7540	0.8046

Tables 6–8 show the metrics value like accuracy, precision, sensitivity, specificity, F1-score and AUC scores for data after Oversampling, same-sampling and under-sampling respectively, it can be seen that over sampling has best evaluation metrics than same and under sampling for four different ML models.

Table 7: Evaluation values for model classifiers same sampling data

Model	Accuracy	Precision	Sensitivity/Recall	Specificity	F1-score	AUC scores
XGBoost	0.9122	0.4914	0.1033	0.9897	0.1708	0.8300
KNN	0.8961	0.3208	0.1682	0.9658	0.2207	0.6600
Random Forest	0.9128	0.6530	0.0057	0.9997	0.0113	0.8100
Gaussian NB	0.7331	0.2102	0.7448	0.7319	0.3279	0.8100

Table 8: Evaluation values for model classifiers under sampling data

Model	Accuracy	Precision	Sensitivity/Recall	Specificity	F1-score	AUC scores
XGBoost	0.7505	0.7412	0.7755	0.7252	0.7580	0.8200
KNN	0.7240	0.7263	0.7253	0.7226	0.7258	0.7700
Random Forest	0.7379	0.7603	0.7005	0.7758	0.7291	0.8200
Gaussian NB	0.7315	0.6960	0.8292	0.6324	0.7568	0.8100

Fig. 4 represents the heat map correlation for different factors with heart disease, it shows less association with genhealth, alcohol drinking and physical activity and show moderate association

with sleep time, mental health, Race, asthma, BMI, sex, skin cancer, and have high association with smoking, kidney disease, diabetic, physical health, stroke, difficulties walking and age.

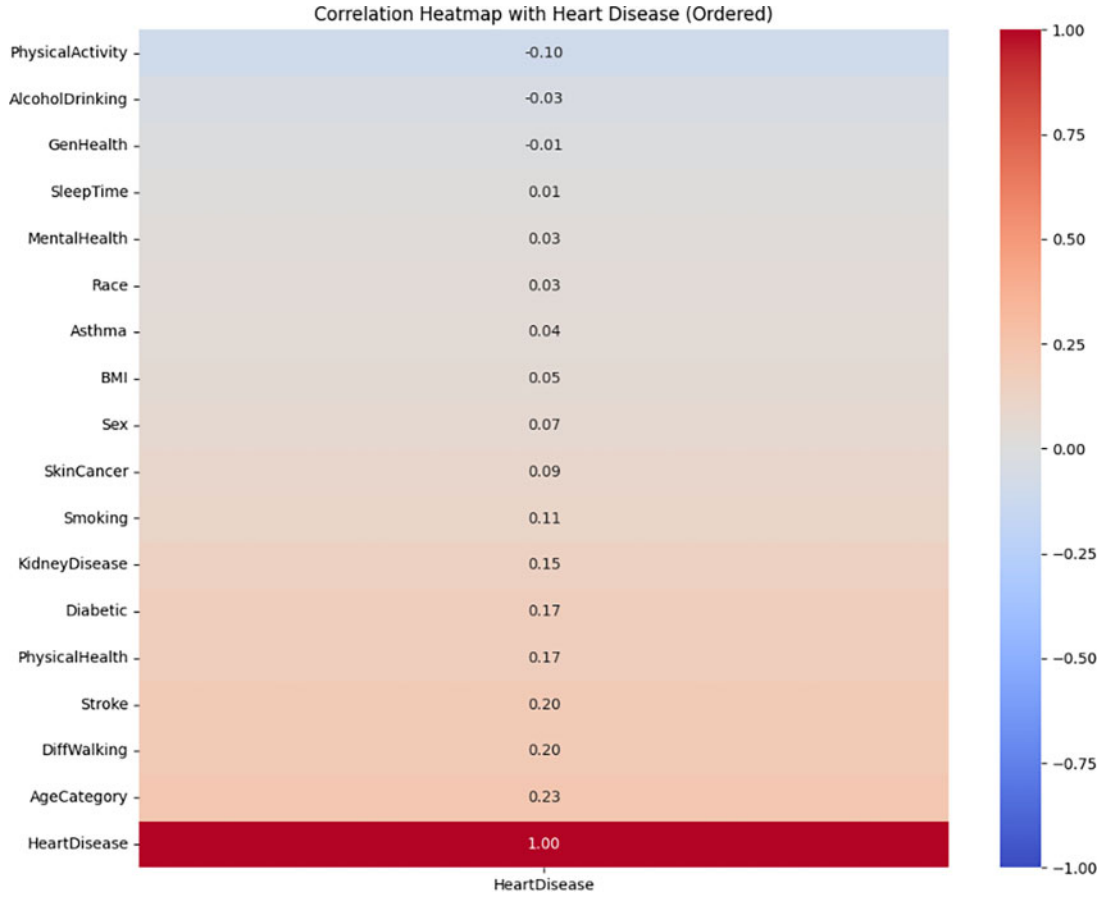


Figure 4: Correlation coefficient as heat map with heart disease

One kind of data visualization that shows the size of a phenomenon in two dimensions is a heat map. The student is better able to comprehend how the spectacles are distributed or change over intergalactic because of the hue variability. Although they are commonly working for analytics, academics also regularly use them and their research to illuminate the impetuses underlying social principles.

The amounts of the minor component are displayed as units, whereas the variables are displayed as cells. The dataset's correlation matrix between variable quantity is shown as a heatmap in Fig. 4. Fig. 4 indicates that whereas warm colors suggest higher values, cool colors suggest lower values. The connection among the row quality call and the column call is displayed, with a value of 1, denoting the maximum correlation coefficient. The worm brown tone implies that the visit and the wait are closely related. The scientists found that the dependent variable, or target variable, had a positive association with age and sleep duration, a negative correlation with alcohol use and overall health, and a positive correlation with diffwalking and age and diabetes.

After gathering the datasets, it was split into training sets and testing sets and then randomly combined. Certain data training techniques were only used during the testing stage before being useful

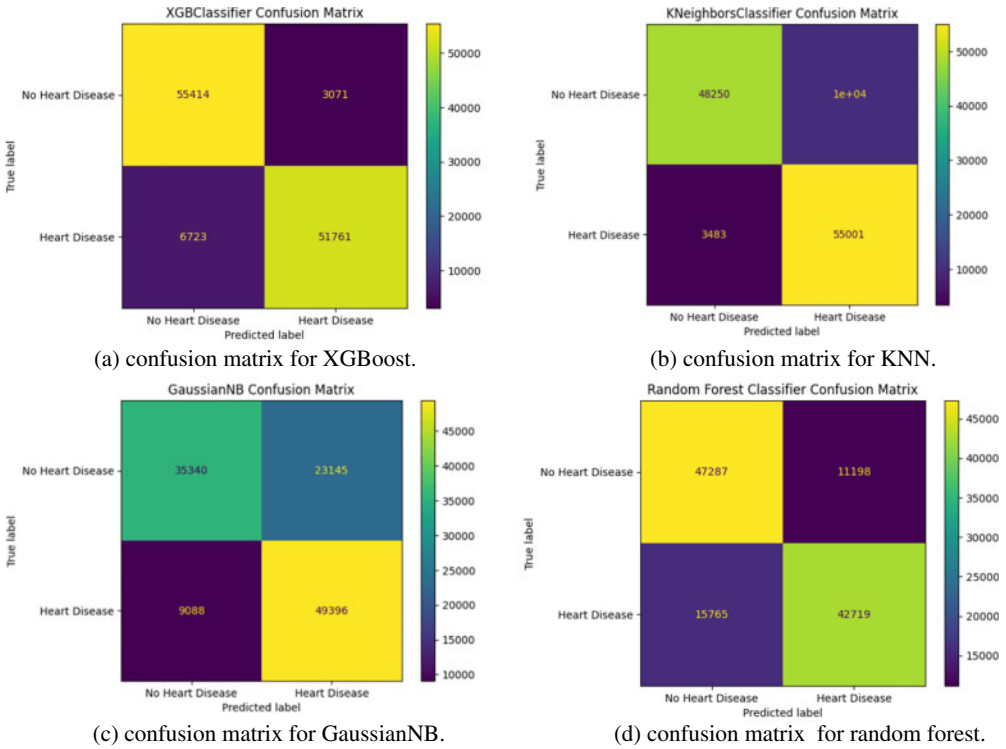
to the testing groups in order to stop data escapes and over-fitting. For the investigation to yield the desired outcomes, a specific setting was required.

(11th Gen Intel(R) Core(TM) i7-1185G7 @ 3.00 GHz 3.00 GHz with 16 GB RAM) is the configuration needed for the experiment's environment. All operations were carried out locally using a Jupyter notebook using Python libraries or modules like NumPy, Scikit-Learn, Seaborn, Pandas, and Matplotlib on the portable computer.

Pearson correlation is limited in scenarios involving categorical predictors or nonlinear relationships, because it assumes continuous variables and linear associations; As a result, it may fail to capture meaningful patterns or dependencies present in more complex or non-linear models.

The outcomes of using a random forest, Xgradient boosting, KNN, and gaussianNB classifier are presented in this section and are shown. Eqs. (2)–(5) display a number of metrics, such as accuracy rate, precision (P), recall (R), and F1-score, that are used to assess how well these algorithms work. Eq. (2) presents the precision metric as a precise indicator of positive analysis. Eq. (3) describes the recall, which defines the measure of genuine positives that are correct. Eq. (4) provides an F-measure for evaluating precision. A specific type of table association known as a confusion matrix makes it possible to evaluate the effectiveness of a monitored teaching strategy.

Fig. 5a shows the confusion matrix for XGBoost classifier with 92% accuracy, Fig. 5b shows the confusion matrix for KNN classifier with 88% accuracy, Fig. 5c shows the confusion matrix for GaussianNB classifier with 72% accuracy and Fig. 5d shows the confusion matrix for Random Forest classifier with 77% accuracy. Also the ROC Curves for four models the XGBoost, KNN, GaussianNB and Random Forest are shown in Fig. 6 below.



(a) confusion matrix for XGBoost.

(b) confusion matrix for KNN.

(c) confusion matrix for GaussianNB.

(d) confusion matrix for random forest.

Figure 5: Plot of confusion matrix for different classifiers

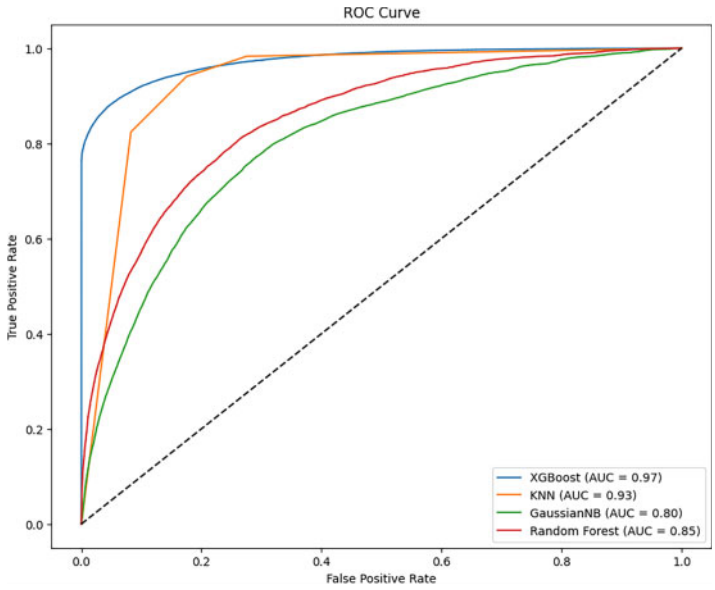


Figure 6: ROC Curves for four models the XGBoost, KNN, GaussianNB and Random Forest

Comparison with Existing Work

Nonetheless, for many years, a number of researchers have experimented with different machine-learning approaches in relation to cardiac disease, with varying grades of achievement. It is vital to discover heart abnormalities as soon as probable so that individuals can be transformed, as this is growing more widespread on the globe. Individuals are therefore motivated to work with the illness and its limitations rather than against them. The comparative study in Table 9 backs up each other's claims that their approach is the most accurate of all the techniques. Different accuracy results may appear due to different dataset types, so for CDC dataset with XGBoost we achieve 91.63% accuracy.

Table 9: Comparison with previous ML-based heart disease prediction works

Authors	Data	Applied method	Accuracy (ACC)
Maini et al. [16]	The Cleveland heart disease (CHD) data	ANN	90.74%
Kavitha et al. [17]	The CHD data	Hybrid model	88.70%
Abdulsalam et al. [26]	The CHD data	Ensemble-quantum machine model	90.16%
Shah et al. [19]	The CHD data	KNN	90.79%
Alotaibi et al. [39]	The CHD data	NB, SVM, KNN, C4.5, RF, AdaBoost, and LR algorithms, as well as machine learning and data mining techniques	87.91%

(Continued)

Table 9 (continued)

Authors	Data	Applied method	Accuracy (ACC)
Das et al. [27]	Key indicators of heart disease	Decision Tree	86.32%
Jahed et al. [28]	Key indicators of heart disease	SGD	76%
Rabbi et al. [25]	Framingham heart disease, and indicators of heart disease dataset (2020)	Bagging ensemble machine learning algorithm	97%
Rabbi et al. [25]	Cleveland heart disease	Voting ensemble machine learning algorithm	92%
Sharma and Singhal [29]	Key indicators of heart disease dataset	XGBoost with ADASYN SMOTE	94.7%
Proposed System	CDC-Kaggle Data (Indicators of Heart Disease)	XGBoost	91.63%

4 Discussion

In this study we introduce four different classifiers the random forest, the KNN, the XGBoost and the gaussianNB, every one of those classifiers give good accuracy and the best is XGBoost and the worst is the gaussianNB.

Random forests, sometimes referred to as random decision forests, are cooperative learning approaches that produce a huge quantity of decision trees during training for sorting, regression, and other issues. For classification issues, the random forest produces the class that most of the trees select. For regression problems, the outcome is the average of the trees' predictions [40,41]. Random forests counteract decision trees' propensity to overfit to their training set [40].

Ho [41] created the first algorithm for random decision forests in 1995 using the random subspace method, which was devised as a way to apply Eugene Kleinberg's "stochastic discrimination" approach to classification [42–45].

After developing an extension of the method, Breiman [46] and Cutler and Liaw [47] registered the trademark "Random Forests" in 2006; Minitab, Inc. presently owns this property [41,48]. The extension combines random feature selection with Breiman's "bagging" approach, first introduced by Ho [41] and later separately by Amit and Geman [49], to produce a assembly of decision trees with measured variance.

Decision trees are a widespread method for a range of machine learning issues. Tree learning is essentially "an off-the-shelf procedure for data mining" because it produces inspectable models, is resistant to the insertion of superfluous features, and remains invariant despite scaling and other feature value changes, claim Hastie et al. Seldom are they right.

Trees that are grownup very deep, in particular, have a tendency to change very erratic patterns; they overfit their training sets, meaning they have low bias but high variance. Random forests are a method for averaging multiple deep decision trees that were competent on different parts of the same

training set in order to decrease the variation. This frequently results in an important improvement in the final model's concert, but at the expense of a small upsurge in bias and a loss of interpretability.

Evelyn Fix and Joseph Hodges first developed the k-nearest neighbors algorithm (k-NN), a non-parametric supervised learning method, in 1951 [50], which Thomas Cover subsequently improved [51]. It can be used for both classification and regression. In all cases, the input consists of the k closest training samples from a data collection. The result depends on whether k-NN is used for classification or regression:

- The outcome of the k-NN classification procedure is a class membership. Based on a plurality division of its neighbors, an item is allocated to the class that is most frequent among its k nearest neighbors (k is a positive number, typically small). If $k = 1$, the object is simply located in the class of that one nearest neighbor.
- The outcome of k-NN regression is the object's assets value. This number is the mean of the values of the k nearest neighbors. If $k = 1$, the value of the single nearest neighbor is simply set to the output.

With k-NN classification, the function is just locally approached and no calculation is performed until the function is assessed. Since this method classifies data based on distance, feature-wise normalization of the training data can greatly recover the accuracy of the algorithm if the features arrive in widely different scales or represent different physical units.

For both regression and classification, it can be useful to provide weights to neighbor contributions, so that the earlier neighbors add more to the average than the beyond neighbors. One popular increment technique, for instance, is to assign a weight of 1/d to each neighbor, where d is the neighbor's distance.

The neighbors are selected from a group of items for whom the class or object attribute value is known when utilizing k-NN classification or k-NN regression. This can be believed of as the algorithm's training set, even though an obvious training phase is not required.

The k-NN algorithm's sensitivity to the local structure of the data is one of its peculiarities and occasionally even a drawback.

As a research project for the Distributed Deep Machine Learning Community (DMLC) group, Chen [52] originally created XGBoost. Initially, it was a terminal application that could be constructed using a libsvm configuration file. When it was used in the engaging solution for the Higgs Machine Learning Challenge, it became well-known in the ML struggle community. Package implementations for XGBoost are currently available in Java, Scala, Julia, Perl, and other languages, and the Python and R packages were built soon after. This broadened the library's user base and improved its standing in the Kaggle community, where it has been used in many competitions [53].

In the areas where it was utilized, it was immediately combined with a quantity of other packages, making it cooler to use. It has now been combined with scikit-learn for Python users and the caret package for R users. It may also be comprised into Data Flow frameworks like Apache Spark, Apache Hadoop, and Apache Flink using the abstracted Rabit and XGBoost4J. XGBoost is also available on OpenCL for FPGAs. Guestrin and Chen have demonstrated a scalable and efficient XGBoost application.

The inherent interpretability of decision trees is lost when using the XGBoost model, even if it frequently achieves advanced accuracy than a single decision tree. It is easy to track the decision-making path of a decision tree, for instance, but considerably more difficult to follow the pathways of hundreds or thousands of trees.

A simple approach to creating classifiers is GaussianNB, which creates models that, given issue examples expressed as vectors of feature values, classify those instances according to a limited number of class labels. Rather than using a single method, a diversity of procedures based on the same principle exists: all gaussianNB classifiers assume that the rate of a given feature is independent of the value of any other feature given the class capricious. For example, if a fruit is round, red, and about 10 cm in diameter, it may be classified as an apple.

A gaussianNB classifier considers each of the roundness, color, and distance features as independently contributing to the likelihood that this fruit is an apple, regardless of any correlations that may exist between them.

It is feasible to work with a gaussianNB model without adopting Bayesian probability or employing any Bayesian techniques; in many real-world situations, the maximum likelihood method is used to estimate parameters for gaussianNB models.

Despite their imprecise construction and ostensibly straightforward assumptions, gaussianNB classifiers have demonstrated impressive performance in a range of difficult real-world settings. Theoretical explanations for the seemingly unlikely efficacy of gaussianNB classifiers were found in analyses of the Bayesian classification problem in 2004. GaussianNB classification is, however, less effective than other approaches like as boosted trees or random forests, according to a comprehensive review conducted in 2006 vs. alternative classification algorithms. GaussianNB has the advantage of using little training data to forecast the constraints needed for classification.

The prevalence of cardiovascular disease (CVD) is rising because of many reasons such as aging population, lack of physical activity, poor dietary habits, steadily increasing cases of obesity and diabetes, and protracted exposure to chronic stress stimuli, all of which tend to raise cardiovascular risk. But advanced prediction and diagnosis of early signs of CVD using machine learning algorithms could certainly go a long way to mitigate cases of this prevalent cardiovascular condition by picking out hidden signs of risk not easily discernable through contemporary approaches being followed reduce healthcare costs.

It becomes clear from this investigation of various machine learning approaches for the classification and detection of cardiovascular illness that classical silo approaches are used in all research papers pertaining to disease prediction and detection. The majority of machine learning approaches have only used small datasets with certain parameters, and their disease prediction accuracy has not yet reached 100%. So as an addendum to this review new algorithms, feature engineering techniques or data pretreatment methods might be studied to improve the accuracy of ML models.

In addition, it is suggested that large-scale heterogeneous datasets with information from various demographics, geographic areas, and populations be employed to accomplish the aforementioned objective. Therefore, the current paper's future research could concentrate on applying and verifying ML models in actual clinical situations.

5 Conclusion

Heart disease (HD) is a serious condition that results in many deaths each year. If individuals ignore the warning signs of heart disease, they could face severe consequences very quickly. The

primary objective is to outline various data mining techniques that can be employed to predict cardiac disease. Our goal is to use fewer features and tests to get more accurate and useful predictions than before. In this work, several preprocessing methods and machine learning processes are active to conduct in-depth analysis and produce the desired outcomes.

The developed system is trained and tested using datasets from the CDC. If the gathered data were used, the XGBoost classifier outperformed the ML technique for each of the 18 attributes in the dataset. The research team classified heart abnormalities using a range of supervised learning classifications; XGBoost produced an accuracy of 91.63%, precision of 94.40%, recall of 88.50%, F1 score of 91.36%, specificity of 94.75%, and AUC score of 97.39% in the diagnosis process. If characteristics are handled effectively, achievement within the ordering of heart disease judgment should be apparent. Cardiovascular disease classification can be effective if features are kept up to date. This study can be extended to create the correctness, equality, transparency, and results of the model for the discovery of heart disease by the use of an explainable machine learning technique. This study can also be integrated into real-world clinical settings for practical use and expanded with the use of a self-created dataset.

Acknowledgement: Not applicable.

Funding Statement: The authors declare that this study received no financial support.

Author Contributions: Yahia Z. Rawash, as the first author, analyzed, organized, and processed the indicator of heart disease dataset samples and completed the original draft manuscript; Hua-Nong Ting designed, reviewed, and revised the manuscript; Kok Han Chee administrated the project. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: This manuscript did not produce new data; all data were described in the Methodology section and were sourced from indicator of heart disease datasets.

Ethics Approval: This article does not involve any research or experimental data or other information related to human or animal subjects.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Mancia G, Oparil S, Whelton PK, McKee M, Dominiczak A, Luft FC, et al. The technical report on sodium intake and cardiovascular disease in low- and middle-income countries by the joint working group of the World Heart Federation, the European Society of Hypertension and the European Public Health Association. Eur Heart J. 2017;38(10):712–9. doi:10.1093/eurheartj/ehw549.
2. Coronary A. Heart disease. 2020 [cited 2026 Jan 1]. Available from: https://www.aihw.gov.au/reports/heart-stroke-vascular-diseases/hsvd-facts/contents/all-heart-stroke-and-vascular-disease/coronary-heart-disease.
3. D'Souza A. Heart disease prediction using data mining techniques. Int J Res Eng Sci. 2015;3(3):74–7. doi:10.56726/irjmets38413.
4. Loesche WJ. Periodontal disease as a risk factor for heart disease. Compendium. 1994;15(8):976–8. doi:10.1515/9781937585921-009.
5. Sen SK. Predicting and diagnosing of heart disease using machine learning algorithms. Int J Eng Comput Sci. 2017;6(6):21623–31. doi:10.18535/ijecs/v6i6.14.

6. Hertel R, Benlamri R. A deep learning segmentation-classification pipeline for X-ray-based COVID-19 diagnosis. Biomed Eng Adv. 2022;3:100041. doi:10.1016/j.bea.2022.100041.
7. Chattopadhyay A, Maitra M. MRI-based brain tumour image detection using CNN based deep learning method. Neurosci Inform. 2022;2(4):100060. doi:10.1016/j.neuri.2022.100060.
8. Mamatha SK, Krishnappa HK, Shalini N. Graph theory based segmentation of magnetic resonance images for brain tumor detection. Pattern Recognit Image Anal. 2022;32(1):153–61. doi:10.1134/S1054661821040167.
9. Ahmad GN, Fatima H, Ullah S, Salah Saidi A, Imdadullah. Efficient medical diagnosis of human heart diseases using machine learning techniques with and without GridSearchCV. IEEE Access. 2022;10:80151–73. doi:10.1109/ACCESS.2022.3165792.
10. Rahman MMM, Rana MMR, Nur-A-Alam M, Islam S, Mohi KKM, Uddin. A web-based heart disease prediction system using machine learning algorithms. Netw Biol. 2022;12(2):64–81.
11. Dey SK, Rahman MM, Howlader A, Siddiqi UR, Uddin KMM, Borhan R, et al. Prediction of dengue incidents using hospitalized patients, metrological and socio-economic data in Bangladesh: a machine learning approach. PLoS One. 2022;17(7):e0270933. doi:10.1371/journal.pone.0270933.
12. Oliveira DVB, da Silva JF, de Sousa Araújo TA, Albuquerque UP. Influence of religiosity and spirituality on the adoption of behaviors of epidemiological relevance in emerging and re-emerging diseases: the case of dengue fever. J Relig Health. 2022;61(1):564–85. doi:10.1007/s10943-021-01436-x.
13. Biswas N, Uddin KMM, Rikta ST, Dey SK. A comparative analysis of machine learning classifiers for stroke prediction: a predictive analytics approach. Healthc Anal. 2022;2:100116. doi:10.1016/j.health.2022.100116.
14. Ghorbian M, Ghobaei-Arani M. A comprehensive review of LLM applications for lung cancer diagnosis and treatment: classification, challenges, and future directions. J Big Data. 2025;12:249. doi:10.1186/s40537-025-01304-5.
15. Akhtar J. Non-small cell lung cancer classification from histopathological images using feature fusion and deep CNN. Int J Eng Adv Technol. 2020;9(5):1013–8. doi:10.1101/197574.
16. Maini E, Venkateswarlu B, Gupta A. Applying machine learning algorithms to develop a universal cardiovascular disease prediction system. In: International Conference on Intelligent Data Communication Technologies and Internet of Things (ICICI). Berlin/Heidelberg, Germany: Springer; 2019. p. 627–32. doi:10.1007/978-3-030-03146-6_69.
17. Kavitha M, Gnaneswar G, Dinesh R, Sai Y, Suraj R. Heart disease prediction using hybrid machine learning model. In: Proceedings of the 2021 IEEE 6th International Conference on Inventive Computation Technologies (ICICT); 2021 Jan 20–22; Coimbatore, India. p. 1329–33.
18. Katarya R, Meena SK. Machine learning techniques for heart disease prediction: a comparative study and analysis. Health Technol. 2021;11(1):87–97. doi:10.1007/s12553-020-00505-7.
19. Shah D, Patel S, Bharti SK. Heart disease prediction using machine learning techniques. SN Comput Sci. 2020;1(6):345. doi:10.1007/s42979-020-00365-y.
20. Bharti R, Khamparia A, Shabaz M, Dhiman G, Pande S, Singh P. Prediction of heart disease using a combination of machine learning and deep learning. Comput Intell Neurosci. 2021;2021:8387680. doi:10.1155/2021/8387680.
21. Parashar AK, Jamliya A, Nasrat S, Soni R. XGBoost for heart disease prediction achieving high accuracy with robust machine learning techniques. Int J Innov Sci Eng Manage. 2025;4(3):185–91. doi:10.69968/ijisem.2025v4i3185-191.
22. Ali L, Niamat A, Ali Khan J, Golilarz NA, Xiong X, Noor A, et al. An optimized stacked support vector machines based expert system for the effective prediction of heart failure. IEEE Access. 2019;7:54007–14. doi:10.1109/access.2019.2909969.

23. Javeed A, Zhou S, Liao Y, Qasim I, Noor A, Nour R. An intelligent learning system based on random search algorithm and optimized random forest model for improved heart disease detection. IEEE Access. 2019;7:180235–43. doi:10.1109/ACCESS.2019.2952107.
24. Santhana Krishnan J, Geetha S. Prediction of heart disease using machine learning algorithms. In: Proceedings of the 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT); 2019 Apr 25–26; Chennai, India. p. 1–5. doi:10.1109/ICIICT1.2019.8741465.
25. Rabbi MSH, Bari MM, Debnath T, Rahman A, Das AK, Hossain MP, et al. Performance evaluation of optimal ensemble learning approaches with PCA and LDA-based feature extraction for heart disease prediction. Biomed Signal Process Control. 2025;101:107138. doi:10.1016/j.bspc.2024.107138.
26. Abdulsalam G, Meshoul S, Shaiba H. Explainable heart disease prediction using ensemble-quantum machine learning approach. Intell Autom Soft Comput. 2023;36(1):761–79. doi:10.32604/iasc.2023.032262.
27. Das RC, Das MC, Hossain MA, Rahman MA, Hossen MH, Hasan R. Heart disease detection using ML. In: Proceedings of the 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC); 2023 Mar 8–11; Las Vegas, NV, USA. p. 983–7. doi:10.1109/CCWC57344.2023.10099294.
28. Jahed R, Aseer O, Al-Mousa A. Using personal key indicators and machine learning-based classifiers for the prediction of heart disease. In: Proceedings of the 2023 International Conference on Smart Computing and Application (ICSCA); 2023 Feb 5–6; Hail, Saudi Arabia. p. 1–6. doi:10.1109/ICSCA57840.2023.10087430.
29. Sharma S, Singhal A. A novel heart disease prediction system using XGBoost classifier coupled with ADASYN SMOTE. In: Proceedings of the 2023 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS 2023); 2023 Nov 3–4; Greater Noida, India.
30. Ahamed J, Mir RN, Chishti MA. Industry 4.0 oriented predictive analytics of cardiovascular diseases using machine learning, hyperparameter tuning and ensemble techniques. Ind Robot Int J Robot Res Appl. 2022;49(3):544–54. doi:10.1108/ir-10-2021-0240.
31. Ahamed J, Manan Koli A, Ahmad K, Alam Jama M, Gupta BB. CDPS-IoT: cardiovascular disease prediction system based on IoT using machine learning. Int J Interact Multimed Artif Intell. 2022;7(4):78–86. doi:10.9781/ijimai.2021.09.002.
32. Gavhane A, Kokkula G, Pandya I, Devadkar K. Prediction of heart disease using machine learning. In: Proceedings of the 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA); 2018 Mar 29–31; Coimbatore, India. p. 1275–8. doi:10.1109/ICECA.2018.8474922.
33. CDC. Centers for Disease Control and Prevention. 2015 [cited 2026 Jan 1]. Available from: https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease.
34. Breiman L, Friedman JH, Olshen RA, Stone CJ. Classification and regression trees. London, UK: Routledge; 2017. doi:10.1201/9781315139470.
35. Schonlau M, Zou RY. The random forest algorithm for statistical learning. Stata J Promot Commun Stat Stata. 2020;20(1):3–29. doi:10.1177/1536867x20909688.
36. Speiser JL, Miller ME, Tooze J, Ip E. A comparison of random forest variable selection methods for classification prediction modeling. Expert Syst Appl. 2019;134:93–101. doi:10.1016/j.eswa.2019.05.028.
37. Staffini A, Svensson T, Chung UI, Svensson AK. Heart rate modeling and prediction using autoregressive models and deep learning. Sensors. 2021;22(1):34. doi:10.3390/s22010034.
38. Sangam S, Sharma P, Kushwaha D, Rana R, Singh H. Disease prediction using machine learning. In: Proceedings of the International Conference on Innovative Computing & Communication (ICICC 2024); 2024 Aug 21–24; Chennai, India. doi:10.2139/ssrn.5241511.
39. Alotaibi N, Alzahrani M. Comparative analysis of machine learning algorithms and data mining techniques for predicting the existence of heart disease. Int J Adv Comput Sci Appl. 2022;13(7):810–8. doi:10.14569/ijacsa.2022.0130794.
40. Hastie T, Tibshirani R, Friedman J. The elements of statistical learning. 2nd ed. Berlin/Heidelberg, Germany: Springer; 2008.

41. Ho TK. The random subspace method for constructing decision forests. IEEE Trans Pattern Anal Machine Intell. 1998;20(8):832–44. doi:10.1109/34.709601.
42. Panhalkar AR, Doye DD. A novel approach to build accurate and diverse decision tree forest. Evol Intell. 2022;15(1):439–53. doi:10.1007/s12065-020-00519-0.
43. Kleinberg E. Stochastic discrimination. Ann Math Artif Intell. 1990;1(1–4):207–39. doi:10.1007/bf01531079.
44. Kleinberg EM. An overtraining-resistant stochastic modeling method for pattern recognition. Ann Statist. 1996;24(6):2319–49. doi:10.1214/aos/1032181157.
45. Kleinberg EM. On the algorithmic implementation of stochastic discrimination. IEEE Trans Pattern Anal Machine Intell. 2000;22(5):473–90. doi:10.1109/34.857004.
46. Breiman L. Random forests. Mach Learn. 2001;45(1):5–32. doi:10.1023/a:1010933404324.
47. Cutler A, Liaw A. Documentation for R Package 'randomForest'. [cited 2026 Jan 1]. Available from: https://cran.r-project.org/web/packages/randomForest/randomForest.pdf.
48. Random forests—Trademark Details. Minitab, LLC. Registration Number 3185828. Serial Number 78642027. Justia Trademarks. [cited 2020 Jan 18]. Available from: http://trademarks.justia.com/786/42/random-78642027.html.
49. Amit Y, Geman D. Shape quantization and recognition with randomized trees. Neural Comput. 1997;9(7):1545–88. doi:10.1162/neco.1997.9.7.1545.
50. Fix E, Hodges JL. Discriminatory analysis. In: Nonparametric discrimination: consistency properties. Randolph Field, TX, USA: USAF School of Aviation Medicine; 1951.
51. Cover T, Hart P. Nearest neighbor pattern classification. IEEE Trans Inform Theory. 1967;13(1):21–7. doi:10.1109/tit.1967.1053964.
52. Story and Lessons behind the evolution of XGBoost. [cited 2026 Jan 1]. Available from: https://tqchen.com/old_post/2016-03-10-story-and-lessons-behind-the-evolution-of-xgboost.
53. XGBoost—ML Winning Solutions. GitHub. [cited 2026 Jan 1]. Available from: https://github.com/datacamp/Machine-Learning-With-XGboost-live-training/blob/master/notebooks/Machine-Learning-with-XGBoost-solution.ipynb.