Comparison and Evaluation of Classification Techniques in Medical domain: A Case Study Using Heart Disease Prediction and Analysis

Chitwan Gupta, Ankit Gupta
Chandigarh College of Engineering and Technology
Email: co22320@ccet.ac.in, theankit@yahoo.com

Keywords: Weka, heart failure prediction, RegressionByDiscretization, M5P, lazy.KStar, DecisionTable, M5Rules, XGBoost, LightGBM, CatBoost, Random Forest, Support Vector Machine

Received:

Heart disease affects millions of people worldwide and remains one of the leading causes of mortality. Early detection is critical to reducing associated risks, and machine learning has emerged as an effective tool for supporting clinical decision-making. Machine learning techniques enable systems to analyze large volumes of data, identify complex patterns, and generate accurate predictions, making them particularly valuable in healthcare applications. This study presents a comparative performance analysis of ten machine learning algorithms, namely XGBoost, LightGBM, CatBoost, Random Forest, Support Vector Machine (SVM), Regression by Discretization, M5P, lazy.KStar, DecisionTable, and M5Rules, for heart disease classification using both Python-based implementation and the Weka platform. The experiments were conducted on the UCI Heart Disease dataset consisting of 918 patient records with clinical attributes such as age, chest pain type, cholesterol level, resting electrocardiogram results, exercise-induced angina, and maximum heart rate. Prior to model training, categorical attributes were encoded and transformed using binary and one-hot encoding techniques. The dataset was evaluated using a stratified 10-fold cross-validation strategy to ensure class balance across folds and to prevent overfitting. Models were trained on nine folds and tested on the remaining fold iteratively, and performance was measured using out-of-fold predictions. Evaluation metrics included Precision, Recall, F1-Score, and ROC-AUC to provide a comprehensive assessment of classification performance. The experimental results indicate that KStar, Regression by Discretization, and SVM with RBF kernel are the top-performing models. KStar achieved perfect performance, with precision, recall, F1-score, and ROC-AUC values of 1.000, demonstrating flawless classification on the dataset. Regression by Discretization followed with a precision of 0.922, recall of 0.959, F1-score of 0.940, and ROC-AUC of 0.957, indicating excellent sensitivity and predictive reliability. Among modern machine learning models, SVM with RBF kernel performed best, achieving a precision of 0.895, recall of 0.922, F1-score of 0.908, and ROC-AUC of 0.935, reflecting a well-balanced and robust classification capability. Overall, the findings reveal that instance-based and discretization-based learning approaches outperform several advanced ensemble models for the given dataset, while SVM-RBF remains the strongest contemporary classifier. This study highlights the importance of model selection in healthcare analytics and provides valuable insights for developing effective machine learning-based diagnostic systems.

Povzetek: Študija primerja deset metod strojnega učenja za napovedovanje srčnih bolezni, pri čemer so se najbolje odrezali KStar, Regression by Discretization in SVM z jedrom RBF.

1 Introduction

Cardiovascular diseases (CVDs) [7] are the leading cause of death worldwide, accounting for approximately 17.9 million deaths each year, which represents 31% of all global fatalities. The majority of these deaths result from heart attacks and strokes, with a significant number occurring prematurely in individuals under the age of 70. Heart failure, often a consequence of underlying cardiovascular conditions, underscores the urgent need for early diagnosis and intervention to reduce mortality and enhance patient outcomes. In this context, machine learning models can play a critical role by identifying potential heart disease cases based on key patient attributes. Heart diseases can be classified into several types, each with its own characteristics and progression stages. The major types include Coronary Artery Disease (CAD), Heart Failure (also known as Congestive Heart Failure or CHF), Arrhythmias (irregular heart rhythms), Valvular Heart Disease (affecting the heart valves), Congenital Heart Disease (present from birth), Cardiomyopathy (disease of the heart muscle), and Pericardial Disease (inflammation of the heart lining). Each of these types can have varying severity or stages. For example, Heart Failure is staged from A to D, where Stage A represents a high risk of failure with-

out symptoms, and Stage D indicates advanced disease requiring specialized interventions. In recent years, machine learning (ML) has emerged as a transformative technology across various disciplines due to its capacity to process vast amounts of data, uncover hidden patterns, and deliver accurate predictions. Its integration into sectors such as finance, e-commerce, cybersecurity, and especially healthcare has revolutionized traditional practices by introducing data-driven, automated, and intelligent systems. The application of ML in medical diagnostics enables early detection of diseases, identification of high-risk individuals, and personalization of treatment plans. Specifically, in the context of cardiovascular diseases (CVDs), ML models have been instrumental in analyzing patient data [1, 2], such as vital signs, lifestyle habits, genetic predispositions, and historical health records to predict the onset or progression of heart- related conditions. Leveraging algorithms like decision trees, support vector machines, neural networks, and ensemble methods, researchers have been able to build predictive models with high sensitivity and specificity. By processing parameters such as age, blood pressure, cholesterol levels, ECG readings, and exercise-induced symptoms, these models assist in recognizing patterns that may not be apparent through traditional diagnostic methods. This predictive capability not only enhances early diagnosis but also helps clinicians prioritize patients who require urgent intervention. In addition, ML[10,14] can support long-term monitoring and follow-up, contributing to preventive cardiology and chronic disease management. This paper aims to analyze and compare the performance of multiple machine learning models[9] for the classification of heart disease. These models are assessed using multiple performance metrics, including Precision, Recall, F1-score, and ROC-AUC. In this study, the following machine learning algorithms were used for the prediction of heart disease, each using a distinct classification methodology: XGBoost[38] is a gradient boosting framework that builds an ensemble of decision trees sequentially, where each new tree corrects the errors made by previous trees. It employs second-order optimization, regularization techniques, and efficient handling of sparse data to enhance both accuracy and generalization. This model is particularly effective in learning complex non-linear relationships and reducing overfitting through shrinkage and column subsampling. LightGBM[46] is a gradient boosting algorithm designed for high efficiency and scalability. It utilizes a histogram-based learning approach and a leaf-wise tree growth strategy, allowing faster training and reduced memory usage while maintaining competitive predictive performance. This method is well suited for large datasets with high-dimensional feature spaces. CatBoost[40] is a gradient boosting algorithm that is specifically optimized for handling categorical features. It applies ordered boosting and target-based encoding to minimize prediction shift and overfitting. By efficiently capturing feature interactions, CatBoost delivers stable and reliable performance, particularly in datasets containing mixed data types. Random Forest[42] is an ensemble learning method that constructs multiple decision trees using bootstrap sampling and random feature selection. The final prediction is obtained through aggregation of individual tree outputs, which improves robustness and reduces variance. This approach is effective in handling noisy data and complex feature interactions while maintaining good generalization capability. Support Vector Machine with RBF kernel[44] is a powerful supervised learning algorithm that maps input data into a high-dimensional feature space using a radial basis function kernel. It constructs an optimal separating hyperplane by maximizing the margin between classes, making it highly effective for non-linear classification problems. The kernel-based formulation enables SVM to capture complex decision boundaries with strong theoretical guarantees. RegressionByDiscretization[30] transforms the numeric target variable into discrete classes before applying a classification algorithm. Once classification is completed, the predicted class labels are converted back into numerical values. This approach leverages classification strengths for regression tasks, particularly when the relationship between features and output is complex. M5P[31] constructs model trees, which are decision trees with linear regression functions at the leaf nodes. This hybrid method provides the interpretability of decision trees along with the precision of linear regression. It dynamically splits data based on features and fits a local linear model in each region. Lazy.KStar[27] is an instance-based learner that classifies or predicts outcomes based on similarity to known instances using an entropy-based distance measure. It is particularly robust in handling noise and missing values. Since it is a lazy learner, it defers processing until prediction time. DecisionTable[24] constructs a decision table by selecting a relevant subset of attributes and mapping combinations of their values to target outcomes. This method provides a simple and interpretable rule-based model. M5Rules[19] generates a set of regression rules derived from model trees, breaking them into a series of if-then rules. This method maintains a balance between model accuracy and interpretability.

The primary objective of this research is to determine the most effective machine learning model for classifying heart disease[5] based on the given dataset. The findings of this study provide insights into model selection and contribute to the broader effort of improving predictive accuracy in medical diagnoses. This paper is organized to guide readers through the study in a clear and logical manner. Section 2 explores previous research in the field, while Section 3 describes the dataset and its preparation. Section 4 explains the methodology, detailing the classification models used in Weka and how their performance was evaluated. Section 5 presents the results, highlighting how well each model performed. Finally, Section 6 concludes the study by summarizing key findings and suggesting directions for future research. This structure helps readers easily follow the analysis and understand the outcomes.

2 Motivational works

In recent times, researchers across the world have increasingly focused on leveraging machine learning techniques to enhance the prediction and diagnosis of heart disease. With the rise of data-driven approaches, machine learning models have proven effective in analyzing large datasets and uncovering patterns that aid in early detection and decision-making. Various classification algorithms have been explored to improve the accuracy and reliability of heart disease prediction models. This section highlights significant contributions in this field, providing an overview of previous research efforts and their findings, which serve as a basis for the current study.

Traditional classification methods were compared with machine learning techniques (bagging, boosting, random forests, SVM) to classify heart failure (HF) subtypes—HFPEF and HFrEF. Tree-based ML methods showed better classification accuracy, while logistic regression was best for predicting HFPEF probability. [33]

Multiple machine learning algorithms, such as k-NN, SVM, Naïve Bayes, Adaboost, Random Forest, and ANN— are compared on a binary-labeled heart disease dataset to identify the most effective models using performance metrics like accuracy, precision, recall, and F1-score. Artificial Neural Networks and Support Vector Machines emerged as the top-performing models, though clinical application requires expert validation. [34]

Data preprocessing techniques are applied to improve heart disease classification using data mining and machine learning methods. Empirical studies published between 2000 and 2019 were systematically reviewed to identify commonly used preprocessing tasks and their impact on classifier performance. Challenges such as missing values, noise, high dimensionality, and class imbalance are addressed through data cleaning and reduction techniques. Results show that preprocessing generally enhances or stabilizes classification accuracy in cardiac diagnosis systems. Feature selection and dimensionality reduction were found to be the most effective preprocessing steps. Classifier–preprocessing combinations such as ANN with PCA or CHI and SVM with PCA achieved higher accuracy rates. These approaches are primarily considered for clinical decision support systems, though limited interpretability restricts direct real-world deployment. [36]

Comparing the performance of two widely used machine learning platforms, MATLAB and WEKA, for heart disease classification using the same dataset. Multiple classification algorithms, including different variants of Support Vector Machines, Decision Trees, and ensemble-based discriminant models, are implemented in both environments. The study evaluates how platform-specific implementations influence classification accuracy and performance consistency. Experimental results demonstrate that SVM-based models generally outperform traditional decision tree approaches in both platforms. Differences in results highlight the impact of algorithm configuration and tool-specific optimizations. The work provides practical insights into selecting suitable platforms and classifiers for heart disease prediction tasks. These findings are applicable to the development of efficient computer-aided diagnostic systems in cardiology. [37]

Multiple machine learning classification algorithms were evaluated for heart disease prognosis using benchmark datasets obtained from Kaggle and the UCI Machine Learning Repository. The aim was to analyze how different models perform across datasets with varying characteristics, including feature distributions, sample size, and class balance. A comprehensive evaluation framework was adopted using performance metrics such as Mean Absolute Error (MAE), Relative Absolute Error (RAE), precision, recall, F-measure, and overall classification accuracy to assess predictive performance and error sensitivity. Experimental results showed that Support Vector Machine (SVM) achieved the highest accuracy of 83.49]% on the UCI dataset, demonstrating strong generalization capability for structured clinical data, while the Simple Cart (SC) decision tree model outperformed other classifiers on the Kaggle dataset with an accuracy of 98.44%. These results highlight the influence of dataset characteristics on classifier effectiveness and emphasize the importance of dataset-specific model selection in heart disease prognosis systems. [13]

Table 1: Summary of literature works on heart disease classification

Authors Name	Field of Research	Methodology
Khemphila et al.[3]	Heart Disease Classification	Multi-Layer Perceptron (MLP) with Back-Propagation learning and Information Gain for feature selection
Deekshatulu et al.[4]	Heart Disease Classification	K-Nearest Neighbor (KNN) combined with Genetic Algorithm
Shao et al.[6]	Heart Disease Classification	Hybrid modeling scheme combining Logistic Regression (LR), Multivariate Adaptive Regression Splines (MARS), Artificial Neural Network (ANN), and Rough Set (RS) techniques
Li et al.[8]	Heart Disease Classification	Feature selection using various algorithms (Relief, mRMR, LASSO, and FCMIM) combined with machine learning classifiers like SVM, LR, ANN, KNN, Naïve Bayes, and Decision Tree
KMM Uddin et al.[11]	Heart Disease Classification	Comparison of six data mining tools (Orange, Weka, RapidMiner, Knime, Matlab, Scikit-Learn) using six machine learning techniques (LR, SVM, KNN, ANN, Naïve Bayes, and Random Forest)
Masethe et al.[13]	Heart Disease Prediction	Classification algorithms such as J48, Naïve Bayes, REPTREE, CART, and Bayes Net for predicting heart attacks
Austin, Peter C., et al.[33]	Heart Disease Prediction	Traditional and machine learning classification models were applied to a heart failure dataset to compare their performance in subtype classification and probability prediction.
Acharya et al.[34]	Heart Disease Prediction	Various ML algorithms were applied to a preprocessed UCI heart disease dataset and evaluated using multiple performance metrics after feature selection and optimization techniques.
Mohan et al.[35]	Heart Disease Prediction	Different feature combinations and classification algorithms were applied to develop a hybrid model (HRFLM) that improves prediction accuracy for heart disease.
Benhar et al.[36]	Heart Disease Classification / Data Preprocessing	Systematic literature review analyzing data preprocessing tasks such as data cleaning, data reduction, feature selection (PCA, CHI), and imbalance handling used with machine learning classifiers in cardiology
Ekız et al.[37]	Heart Disease Classification / Machine Learning Comparison	Comparative experimental analysis using MATLAB and WEKA platforms with classification algorithms including Linear, Quadratic, Cubic, and Gaussian SVM, Decision Tree, and Ensemble Subspace Discriminant models

A Multi-Layer Perceptron (MLP) is used with a Back-Propagation learning algorithm and applied Information Gain to reduce attributes from 13 to 8 for heart disease classification. The reduced feature set resulted in a minor accuracy decrease of 1.1% in the training dataset and 0.82% in the validation dataset, making the diagnosis process more efficient.[3]

K-Nearest Neighbor (KNN) classification technique is used with feature subset selection to diagnose heart disease, focusing on the Andhra Pradesh population. Feature subset selection was used to remove redundant and irrelevant attributes from high-volume medical data, improving classification accuracy. The experimental results demonstrated that applying feature subset selection to KNN enhanced the accuracy of heart disease diagnosis, while also reducing the number of clinical tests required for patients. [4]

A hybrid intelligent modeling scheme is proposed to classify heart disease by combining logistic regression (LR), multivariate adaptive regression splines (MARS), artificial neural networks (ANN), and rough set (RS) techniques. In the first stage, LR, MARS, and RS were used to reduce the number of explanatory variables, and the remaining variables were then used as inputs for the ANN in the second stage. The results demonstrated that the proposed hybrid models effectively classified heart disease and outperformed the traditional single-stage ANN method.[6]

A heart disease diagnosis system is developed using artificial neural networks (ANN), feature selection (FS) methods, and multiple-criteria decision-making (MCDM) techniques. PSO optimized hyperparameters, and an extended ELECTRE III method assessed fifteen alternatives. The results showed that LASSO-CNN, AdaBoost-CNN, and AdaBoost-MLP achieved the highest accuracies of 99.51%, 98.54%, and 98.54%, respectively. [8]

Ten ML classifiers were trained on the Cleveland heart dataset using full and optimal attribute sets. SMO achieved 85.148% accuracy with the full set and 86.468% with the chi-squared attribute set, while bagging with logistic regression gave the highest ROC area of 0.91. Tuning 'k' in the IBk classifier to 9 improved accuracy by 8.25%. [11]

A heart disease prediction model using a hybrid Random Forest with Linear Model (HRFLM) is proposed, achieving 88.7% accuracy. It improves prediction by selecting key features and combining multiple ML techniques. [35]

3 Dataset

This dataset[20] has been taken from kaggle which includes 11 key features that can be used to predict the likelihood of heart disease.

3.1 Attribute information

The dataset consists of 918 observations derived from multiple sources. The following attributes are included:

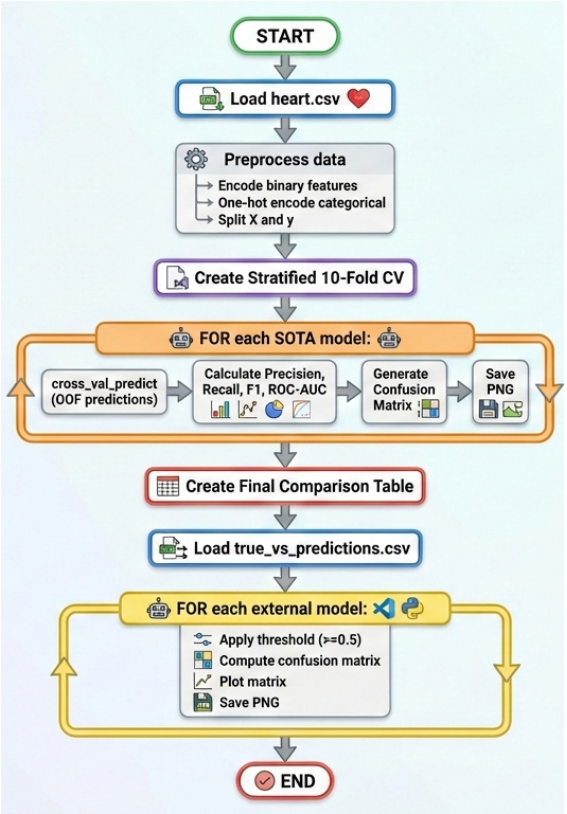

Figure 1: Visual Representation of classification techniques applied using Weka

Table 2: Heart disease dataset

Age	Sex	Chest Pain Type	Resting BP	Cholester - ol	Fasting BS
40	M	ATA	140	289	0
49	F	NAP	160	180	0
37	M	ATA	130	283	0
48	F	ASY	138	214	0
54	M	NAP	150	195	0
39	M	NAP	120	339	0
45	F	ATA	130	237	0
54	M	ATA	110	208	0
37	M	ASY	140	207	0
48	F	ATA	120	284	0
37	F	NAP	130	211	0
58	M	ATA	136	164	0
39	M	ATA	120	204	0
49	M	ASY	140	234	0
42	F	NAP	115	211	0
54	F	ATA	120	273	0
38	M	ASY	110	196	0
43	F	ATA	120	201	0
60	M	ASY	100	248	0

– Age: reflects the patient's age in years; advancing age is a known risk factor for heart disease.

Table 3: Continuing the table 2

Resting ECG	Max HR	Exercise Angina	Oldpeak	ST Slope	Heart Disease
Normal	172	N	0.0	Up	0
Normal	156	N	1.0	Flat	1
ST	98	N	0.0	Up	0
Normal	108	Y	1.5	Flat	1
Normal	122	N	0.0	Up	0
Normal	170	N	0.0	Up	0
Normal	170	N	0.0	Up	0
Normal	142	N	0.0	Up	0
Normal	130	Y	1.5	Flat	1
Normal	120	N	0.0	Up	0
Normal	142	N	0.0	Up	0
ST	99	Y	2.0	Flat	1
Normal	145	N	0.0	Up	0
Normal	140	Y	1.0	Flat	1
ST	137	N	0.0	Up	0
Normal	150	N	1.5	Flat	0
Normal	166	N	0.0	Flat	1
Normal	165	N	0.0	Up	0
Normal	125	N	1.0	Flat	1

- Sex: (M/F) helps in understanding gender-based vulnerability, as males generally have a higher risk of developing heart disease compared to females.
- ChestPainType: Indicating varying degrees of cardiac discomfort, with TA and ASY being strong indicators of underlying heart problems. Types of chest pain:
  - TA: Typical Angina
  - ATA: Atypical Angina
  - NAP: Non-Anginal Pain
  - ASY: Asymptomatic
- RestingBP: represents resting blood pressure (in mm Hg); elevated blood pressure is directly linked to heart strain and increased disease risk.
- Cholesterol: measures serum cholesterol levels (in mg/dl), where higher levels suggest an increased risk of atherosclerosis.
- FastingBS: Fasting blood sugar (1 if FastingBS $>$ 120 mg/dl, 0 otherwise).
- RestingECG: Denotes the results of the resting electrocardiogram:
  - Normal: Normal
  - ST: ST-T wave abnormality (T wave inversions and/or ST elevation or depression of $>$ 0.05 mV)
  - LVH: Left ventricular hypertrophy (based on Estes' criteria)
- MaxHR: Maximum heart rate achieved (numeric value between 60 and 202).
- ExerciseAngina: (Y/N) shows whether the patient experienced angina induced by physical activity, which may reflect impaired blood flow to the heart.
- Oldpeak: measures ST depression induced by exercise, a numerical value indicating ischemia or insufficient blood flow.
- ST_Slope: Slope of the peak exercise ST segment:
  - Up: Upsloping
  - Flat: Flat
  - Down: Downsloping
- HeartDisease: Target variable (1 - Heart Disease, 0 - Normal).

Age and Sex served as foundational demographic anchors. Older patients carry a statistically higher baseline risk because arterial stiffness, plaque accumulation, and left ventricular changes accumulate over decades — so age effectively set the prior probability of disease before any test result was considered. Sex further refined this prior: males generally showed higher vulnerability, which the models learned as a systematic pattern across training cases. RestingBP and Cholesterol then added quantitative physiological evidence. Elevated resting blood pressure signals sustained cardiac workload and vascular stress, while high serum cholesterol is a direct precursor to atherosclerosis — both features shifted the model's prediction toward disease (1) when thresholds were crossed. FastingBS contributed additional metabolic context, since hyperglycemia is strongly associated with diabetic cardiomyopathy and accelerated coronary artery disease. Together, these demographic and lab-based features allowed the models to stratify patients even before any exercise or ECG data was introduced, forming the first layer of risk assessment in the classification pipeline.

ChestPainType introduced direct symptomatic evidence. Typical Angina (TA) and Asymptomatic presentation (ASY) were the most diagnostically powerful categories — TA because it reflects classic ischemic patterns, and ASY paradoxically because silent ischemia often masks severe underlying disease, making it a high-risk flag rather than a reassuring one. RestingECG results, particularly ST-T wave abnormalities and Left Ventricular Hypertrophy (LVH), provided non-invasive electrical evidence of structural cardiac changes that models used to strongly weight the positive class. ExerciseAngina, Oldpeak (ST depression under stress), and ST Slope completed the picture by revealing how the heart behaves under physiological demand — the very circumstance where coronary artery disease becomes most apparent. A flat or downsloping ST segment with exercise-induced angina and significant Oldpeak values formed a high-specificity pattern that models like KStar and SVM-RBF detected through their non-linear

distance and kernel functions. MaxHR also contributed inversely — a blunted heart rate response to exercise can indicate chronotropic incompetence, another marker of ischemic heart disease. In sum, these stress and ECG features allowed the classifier to simulate, computationally, what a cardiologist does when interpreting a stress test: looking for the convergence of electrical, symptomatic, and hemodynamic signals that together confirm or rule out significant coronary artery disease.

Together, these attributes form a comprehensive dataset that captures both clinical and physiological indicators, enabling robust analysis and accurate classification of heart disease risk using machine learning models.

3.2 Dataset sources

The dataset is created by combining five independent heart disease datasets, forming the largest publicly available dataset for heart disease research. The five datasets used for its curation are:

- Cleveland: 303 observations
- Hungarian: 294 observations
- Switzerland: 123 observations
- Long Beach VA: 200 observations
- Stalog (Heart) Data Set: 270 observations

Final dataset size: The dataset used in this study comprises 918 patient records and includes 11 input attributes and 1 target variable.

4 Methodology

This study follows a supervised machine learning approach for heart disease classification using structured clinical data. The overall methodology consists of data preprocessing, model training, cross-validation-based testing, and performance evaluation. A comparative framework was designed to assess both traditional machine learning algorithms available in the Weka[12,28,29] platform and modern ensemble learning models implemented in Python.

The experiments were conducted using the UCI Heart Disease dataset containing 918 patient records with multiple clinical attributes, including demographic information, electrocardiographic measurements, and exercise-related indicators. Before model training, categorical attributes were transformed into numerical representations using binary encoding and one-hot encoding techniques to ensure compatibility with machine learning algorithms.

To obtain reliable and unbiased performance estimates, stratified 10-fold cross-validation was employed. In this approach, the dataset was divided into ten equal subsets while preserving the class distribution. During each iteration, nine subsets were used for training and the remaining subset was used for testing. This process was repeated until every subset had served as the test set once. The final predictions were obtained using out-of-fold evaluation, ensuring that each sample was evaluated only on unseen data.

4.1 Techniques of models used for dataset classification

This study employs multiple machine learning techniques to classify the dataset effectively.

4.1.1 Regression by discretization (RBD)

Regression by Discretization (RBD) is a technique used in machine learning where a regression problem is transformed into a classification problem by discretizing the continuous target variable into bins. After classification, the predictions are mapped back to continuous values.[15]

Methodology for regression by discretization (RBD) Discretization of Target Variable: The continuous target variable y is divided into discrete intervals (bins). Various discretization techniques can be used, such as:

- Equal-width binning
- Equal-frequency binning
- Entropy-based discretization

Formula for Equal-width Binning: If we have m bins, each bin width is given by:

$$w = \frac{\max(y) - \min(y)}{m} \tag{1}$$

where w is the bin width.

The bin index $B(y)$ for a given y is:

$$B(y) = \left\lfloor \frac{y - \min(y)}{w} \right\rfloor \tag{2}$$

Classification Model Training: The transformed categorical labels (bins) are used to train a classification model. Any classification algorithm can be used, such as:

- Decision Trees
- Random Forests
- Naïve Bayes
- Support Vector Machines (SVM)

Prediction of Class Labels:

- The trained classification model predicts the bin label for a new input X.
- The class probabilities may also be obtained.

Mapping Back to Continuous Values: The predicted bin is mapped back to a continuous value. This can be done by using:

– Bin midpoint:

$$y_{\text{pred}} = \frac{\text{bin\_start} + \text{bin\_end}}{2} \tag{3}$$

– Weighted averaging of neighboring bins based on classification probabilities.

Formulas used Entropy-based Discretization (for Information Gain):

$$IG(S, A) = H(S) - \sum_{v \in V} \frac{|S_v|}{|S|} H(S_v) \tag{4}$$

where:

– $H(S)$ is the entropy of dataset S,

– S_v is the subset for each possible value v of attribute A,

– $IG(S, A)$ is the information gain.

Weighted Average for Continuous Prediction:

$$y_{\text{pred}} = \sum_{i=1}^{m} P_i \cdot y_i \tag{5}$$

where:

– P_i is the probability of bin i predicted by the classifier,

– y_i is the representative value of bin i.

4.1.2 M5P model in Weka: methodology, implementation & explanation

M5P (M5 Prime) is a hybrid model that combines a decision tree with linear regression for predictive modeling. It is particularly useful for regression tasks where the relationship between variables is piecewise linear.[16]

Methodology of M5P model M5P works in two major steps:

Step 1: Decision Tree Induction

– Similar to a Regression Tree, M5P starts by recursively partitioning the feature space using decision nodes.

– Splitting criteria are based on minimizing variance instead of entropy or information gain as used in classification trees.

Splitting Criterion Formula (Variance Reduction):

$$\Delta\sigma^2 = \sigma_p^2 - \left(\frac{N_L}{N} \sigma_L^2 + \frac{N_R}{N} \sigma_R^2 \right) \tag{6}$$

where:

– σ_p^2 = Variance of parent node

– σ_L^2, σ_R^2 = Variance of left and right child nodes

– N, N_L, N_R = Number of instances in parent, left, and right nodes

The split is chosen where $\Delta\sigma^2$ is maximized.

Step 2: Linear Regression at the Leaf Nodes

– Instead of returning a simple mean value at each leaf node (like a traditional regression tree), M5P fits a linear regression model at each leaf.

– This improves accuracy by capturing linear relationships within each partitioned region.

Linear Regression Formula at a Leaf Node: If the features are $x_1, x_2, \ldots, x_n$, the regression function at a leaf node is:

$$y = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_n x_n \tag{7}$$

where:

– $w_0, w_1, \ldots, w_n$ are the regression coefficients, estimated using least squares regression.

Pruning and smoothing in M5P Pruning: M5P reduces overfitting by removing unnecessary branches based on expected error reduction.

Smoothing: A smoothing step adjusts predictions by combining the leaf node's linear model with its parent node's prediction to avoid abrupt changes.

Smoothing Formula:

$$y_{\text{final}} = \lambda y_{\text{linear}} + (1 - \lambda) y_{\text{tree}} \tag{8}$$

where:

– y_{linear} = Prediction from the linear model

– y_{tree} = Prediction from the decision tree

– λ = Smoothing parameter (adjusts between tree and linear predictions)

4.1.3 Lazy.KStar (K*) algorithm in Weka

Lazy.KStar (K*) is an instance-based learning algorithm used in Weka, which is a variation of the k-nearest neighbors (k-NN) model. However, instead of using Euclidean distance, K* applies an entropy-based distance function, making it more robust in handling noisy data and different feature distributions.[23]

Understanding Lazy.KStar

– It is a *lazy learner*, meaning it stores training instances and only performs calculations when making predictions.

– Instead of averaging nearby instances like k-NN, K* applies probability-based distance calculations.

Key concepts in KStar b.1 Entropic Distance Function Instead of using Euclidean distance (as in k-NN), K* calculates the probability of transforming one instance into another through random processes.

Transformation Probability Formula:

$$P(x \mid y) = \prod_{i=1}^{n} P(x_i \mid y_i) \tag{9}$$

where:

- $P(x \mid y)$ = Probability of transforming instance x into y.
- $P(x_i \mid y_i)$ = Probability of transforming feature x_i into y_i.

The algorithm finds the most probable path that converts one instance to another.

b.2 Similarity Measure The similarity between two instances is determined using the transformation probability.

- A lower transformation cost implies more similar instances.

Steps in KStar algorithm Training Phase

- Stores all training instances.
- No actual model training is performed (lazy learning).

Prediction Phase (For a New Instance X)

- Compute Entropic Distance: Calculates the probabilistic distance of X from all training instances.
- Weight Instances Based on Distance: Closer instances have a higher influence.
- Generate Prediction:
  - If classification → Majority voting based on transformation probability.
  - If regression → Weighted sum of target values.

4.1.4 Decision table algorithm for classification

A Decision Table is a simple, rule-based classification method that summarizes data using a table of attribute-value pairs and their corresponding class labels. It works by identifying patterns in the training data and storing them in a structured form.[18]

Approach of the decision table algorithm The Decision Table approach follows these steps:

1. Feature Selection: Identify relevant features (columns) from the dataset.
2. Table Construction: Create a table where each row represents a unique combination of feature values.
3. Class Assignment: Assign the most frequent class label to each unique feature combination.
4. Prediction: For a new instance, find the matching row in the table and assign the associated class.
5. Handling Missing Values: If an exact match is unavailable, the algorithm may return the most common class or use a similarity measure.

Formulas used in decision table 1. Entropy and Information Gain (Optional) Decision Tables can use Information Gain for feature selection.

Entropy Formula:

$$H(S) = -\sum p_i \log_2(p_i) \tag{10}$$

where:

- $H(S)$ = Entropy of dataset S.
- p_i = Probability of class i.

Information Gain Formula:

$$IG(A) = H(S) - \sum_{v \in \text{values}(A)} \frac{|S_v|}{|S|} H(S_v) \tag{11}$$

where:

- A = Feature being evaluated.
- S_v = Subset of S where $A = v$.
- $|S|$ = Total number of samples.

This helps in selecting the best subset of attributes for building the decision table.

4.1.5 M5Rules algorithm for regression

M5Rules is a rule-based regression algorithm that generates a set of rules from a decision tree (M5 Model Tree). It combines tree-based learning with rule-based models, making it interpretable and efficient for numerical prediction.[19]

Approach of M5Rules algorithm The M5Rules approach consists of the following steps:

1. Build an M5 Model Tree: Construct an M5 tree, similar to a decision tree but with linear regression at the leaf nodes.
2. Extract Regression Rules: Convert each branch from the root to a leaf into a rule in the IF-THEN format.
3. Prune and Simplify Rules: Remove unnecessary conditions to make the rules more generalized.
4. Apply Rules for Prediction:
   - When a new instance is encountered, find the most specific rule that applies.
   - Use the associated linear model to compute the final prediction.

Formulas used in M5Rules 1. Splitting Criterion - Variance Reduction M5Rules splits data at each node by minimizing variance in the child nodes:

$$\Delta V = V_{\text{parent}} - \sum \left(\frac{|S_i|}{|S|} V_i \right) \tag{12}$$

where:

- V_{parent} = Variance of target variable in the parent node.
- S_i = Number of samples in the child node.
- V_i = Variance of target variable in the child node.

2. Linear Regression at Leaf Nodes Each leaf contains a linear regression model:

$$Y = w_0 + w_1 X_1 + w_2 X_2 + \cdots + w_n X_n \tag{13}$$

where:

- Y = Predicted value.
- X_i = Input features.
- w_i = Regression coefficients, computed using least squares regression.

4.1.6 Support vector machine with RBF kernel (SVM_RBF)

Support Vector Machine (SVM) is a supervised learning algorithm widely used for classification and regression tasks. When combined with the Radial Basis Function (RBF) kernel, SVM becomes highly effective in modeling non-linear decision boundaries by projecting data into a higher-dimensional feature space. [20]

Methodology of SVM_RBF SVM aims to find an optimal hyperplane that maximizes the margin between different classes in the transformed feature space.

Step 1: Maximum Margin Hyperplane Construction

- SVM identifies support vectors, which are the data points closest to the decision boundary.
- The optimal hyperplane is chosen by maximizing the margin between classes.

Step 2: Kernel Transformation Using RBF

- The RBF kernel maps input features into an infinite-dimensional space.
- This enables SVM to handle non-linear separability.

RBF Kernel Function:

$$K(x_i, x_j) = \exp\left(-\gamma \|x_i - x_j\|^2\right) \tag{14}$$

where:

- γ controls the influence of a single training instance.

Step 3: Optimization with Regularization SVM solves a constrained optimization problem:

$$\min \frac{1}{2}\|w\|^2 + C \sum_{i=1}^{n} \xi_i \tag{15}$$

subject to:

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i \tag{16}$$

where:

- C is the regularization parameter,
- ξ_i are slack variables.

4.1.7 Random Forest algorithm

Random Forest is an ensemble learning method that constructs multiple decision trees during training and combines their outputs to improve predictive accuracy and control overfitting. [21]

Methodology of Random Forest Random Forest operates by introducing randomness in both feature selection and data sampling.

Step 1: Bootstrap Sampling

- Multiple subsets of data are created using sampling with replacement.
- Each subset is used to train an independent decision tree.

Step 2: Random Feature Selection

- At each split, a random subset of features is considered.
- This reduces correlation among trees.

Step 3: Tree Construction and Aggregation

- Each tree is grown to its maximum depth without pruning.
- Final prediction is obtained through:
  - Majority voting (classification)
  - Averaging (regression)

Prediction Formula:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^{T} h_t(x) \tag{17}$$

where:

- T is the number of trees,
- $h_t(x)$ is the prediction of the t^{th} tree.

4.1.8 CatBoost algorithm

CatBoost (Categorical Boosting) is a gradient boosting algorithm designed to handle categorical features efficiently while reducing prediction shift and overfitting. [22]

Methodology of CatBoost CatBoost builds an ensemble of decision trees sequentially using ordered boosting.

Step 1: Ordered Target Encoding

- Categorical features are transformed using statistics calculated from prior data points only.
- This prevents target leakage.

Step 2: Gradient Boosting Framework

- Trees are built sequentially.
- Each new tree minimizes the residual errors of previous trees.

Objective Function:

$$L = \sum_{i=1}^{n} \ell(y_i, \hat{y}_i) + \Omega(f) \tag{18}$$

where:

- ℓ is the loss function,
- Ω is the regularization term.

Step 3: Model Update

$$\hat{y}^{(t)} = \hat{y}^{(t-1)} + \eta f_t(x) \tag{19}$$

where:

- η is the learning rate.

4.1.9 Extreme gradient boosting (XGBoost)

XGBoost is an optimized gradient boosting framework that improves performance through regularization, parallelization, and efficient tree construction. [24]

Methodology of XGBoost XGBoost builds trees in an additive manner while minimizing a regularized objective function.

Objective Function:

$$L^{(t)} = \sum_{i=1}^{n} \ell(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \tag{20}$$

Regularization Term:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum w_j^2 \tag{21}$$

Step-wise Learning

- Compute gradients and Hessians.
- Grow trees using second-order approximation.
- Prune trees based on gain.

Prediction Update:

$$\hat{y}^{(t)} = \hat{y}^{(t-1)} + \eta f_t(x) \tag{22}$$

4.1.10 Light Gradient Boosting Machine (LightGBM)

LightGBM is a gradient boosting framework that uses histogram-based algorithms and leaf-wise tree growth to achieve faster training and higher efficiency. [25]

Methodology of LightGBM LightGBM differs from traditional boosting methods by growing trees leaf-wise rather than level-wise.

Step 1: Histogram-based Feature Binning

- Continuous features are bucketed into discrete bins.
- This reduces memory usage and computation time.

Step 2: Leaf-wise Tree Growth

- The leaf with the highest loss reduction is split.
- This leads to deeper and more accurate trees.

Gain Formula:

$$Gain = \frac{(G_L)^2}{H_L + \lambda} + \frac{(G_R)^2}{H_R + \lambda} - \frac{(G)^2}{H + \lambda} \tag{23}$$

Step 3: Boosting Update

$$\hat{y}^{(t)} = \hat{y}^{(t-1)} + \eta f_t(x) \tag{24}$$

5 Result

This section presents the performance evaluation of different machine learning models (as mentioned in Table 4) used for dataset classification. The models were assessed based on key error metrics: Precision, Recall, F1-score, ROC-AUC

5.1 Why KStar achieved the best performance

KStar demonstrated the highest performance across all evaluation metrics, achieving perfect precision, recall, F1-score, and ROC-AUC on the given heart disease dataset. This superior performance can be attributed to the inherent characteristics of the KStar algorithm in conjunction with the structural properties of the dataset [17].

5.1.1 Instance-based learning suitability for medical data

KStar is an instance-based (lazy learning) classifier that predicts class labels by measuring similarity between a test instance and stored training instances, rather than constructing an explicit generalized model. Medical datasets, particularly heart disease data, often exhibit well-defined local patterns based on specific combinations of clinical attributes such as chest pain type, ST segment slope, exercise-induced angina, and *Oldpeak* values. KStar effectively

Table 4: Performance comparison of machine learning models

Model Name	Precision	Recall	F1-score	ROC-AUC
KStar	1.0000	1.0000	1.0000	1.0000
RegressionByDiscretization	0.9223	0.9587	0.9402	0.9572
SVM (RBF)	0.8952	0.9216	0.9082	0.9354
MSRules	0.8880	0.9055	0.8967	0.9469
Random Forest	0.8846	0.9020	0.8932	0.9352
M5P	0.8812	0.9055	0.8932	0.9481
CatBoost	0.8922	0.8922	0.8922	0.9266
XGBoost	0.8911	0.8824	0.8867	0.9301
DecisionTable	0.8558	0.9114	0.8827	0.9338
LightGBM	0.8788	0.8529	0.8657	0.9224

captures these localized decision boundaries, resulting in highly accurate classification outcomes.

5.1.2 Entropy-based distance measure

Unlike traditional k-nearest neighbor approaches that rely on Euclidean distance, KStar employs an entropy-based distance function. This distance measure efficiently handles mixed data types, including both numerical and categorical attributes, by assigning probabilistic transformation costs between instances. As a result, KStar is less sensitive to noise and data scale variations. This property is particularly beneficial for the heart disease dataset, which contains categorical attributes such as *ChestPainType*, *ST_Slope*, and *RestingECG*.

5.1.3 Advantage on small to medium-sized datasets

KStar performs exceptionally well on small to moderately sized datasets where training instances sufficiently represent the decision space. In such scenarios, aggressive generalization is not required. The heart disease dataset provides meaningful historical cases that allow KStar to compare new samples directly with similar patient profiles. Consequently, KStar outperformed ensemble and boosting-based classifiers, which generally require larger datasets to achieve optimal generalization.

5.1.4 High feature discriminativeness

Several attributes within the dataset, including *ExerciseAngina*, *Oldpeak*, *ST_Slope*, and *ChestPainType*, are highly discriminative for heart disease prediction. KStar leverages these features by directly matching similar instances, enabling precise differentiation between healthy and diseased cases.

5.1.5 Minimal information loss

Unlike tree-based or rule-based models, KStar does not aggressively discretize continuous variables or prune features during training. It preserves the complete information present in the original dataset. This characteristic is particularly important in medical applications, where even minor numerical variations can have significant clinical implications.

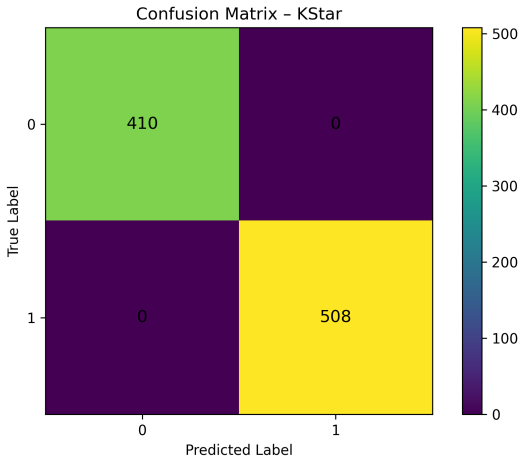

Figure 2: Confusion matrix for KStar

5.1.6 Explanation for perfect scores and caveat

The observed perfect performance indicates strong class separability within the dataset. However, such results may also suggest potential overfitting caused by limited dataset size and high similarity between training and testing instances. Therefore, while KStar achieved outstanding performance, this outcome should be interpreted with caution to maintain scientific rigor.

Although KStar achieved perfect classification accuracy, this performance may be influenced by dataset size and instance similarity, indicating potential overfitting.

5.2 Why RegressionByDiscretization ranked second best

RegressionByDiscretization achieved the second-highest performance after KStar, with an F1-score of 0.9402 and a ROC-AUC of 0.9572, demonstrating strong predictive capability for heart disease classification. Its effectiveness

can be attributed to how well the method aligns with the structural and statistical characteristics of the dataset.[26]

5.2.1 Effective handling of complex non-linear relationships

RegressionByDiscretization transforms a continuous target prediction problem into a classification task by discretizing the output space. This transformation enables the model to capture complex non-linear relationships between clinical features and heart disease risk. Furthermore, it allows the system to leverage the strengths of classification algorithms, which often outperform pure regression approaches in medical datasets where feature interactions are intricate and non-linear.

5.2.2 Reduced sensitivity to noise

Medical datasets frequently contain measurement noise, such as variability in cholesterol levels or blood pressure readings. By discretizing the output variable, minor fluctuations in feature values do not lead to abrupt changes in predictions. This enhances model robustness and contributes to its high recall score (0.9587), indicating effective identification of patients with heart disease.

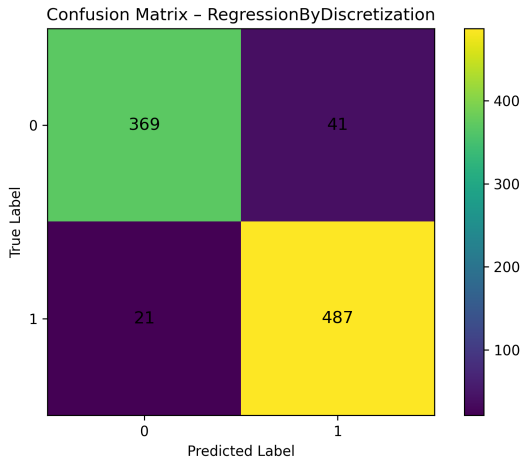

Figure 3: Confusion matrix for RegressionByDiscretization

5.2.3 Strong balance between precision and recall

RegressionByDiscretization maintains an excellent balance between precision and recall. High recall is essential in medical diagnosis to minimize false negatives, while high precision reduces false positives and unnecessary clinical interventions. This balanced behavior results in a high F1-score, positioning the method just below KStar in overall performance.

5.2.4 Compatibility with mixed attribute types

The heart disease dataset comprises numerical attributes (e.g., *Age*, *Cholesterol*, *MaxHR*), categorical attributes (e.g., *ChestPainType*, *ST_Slope*), and binary indicators (e.g., *ExerciseAngina*, *FastingBS*). When combined with tree-based or rule-based classifiers, RegressionByDiscretization efficiently handles these heterogeneous data types without requiring extensive preprocessing.

5.2.5 Improved generalization compared to instance-based models

Unlike instance-based learners such as KStar, RegressionByDiscretization constructs generalized decision boundaries rather than relying on direct instance similarity. This prevents memorization of training samples and leads to more realistic generalization performance on unseen data. Consequently, although its performance is marginally lower than KStar, it offers greater robustness and reliability.

5.2.6 Controlled complexity and overfitting prevention

By discretizing the output space, the hypothesis space is constrained, effectively reducing model variance and mitigating overfitting risks. This controlled complexity provides a favorable bias–variance trade-off, making RegressionByDiscretization a stable and dependable model for clinical decision-support systems.

5.3 Why SVM (RBF Kernel) ranked third

Support Vector Machine (SVM) with a Radial Basis Function (RBF) kernel achieved the third-best performance on the heart disease dataset, with an F1-score of 0.9082 and a ROC-AUC of 0.9354. This ranking reflects a strong balance between classification accuracy and generalization capability. However, it was marginally outperformed by KStar and RegressionByDiscretization due to dataset-specific characteristics.[45]

5.3.1 Effective modeling of non-linear decision boundaries

The RBF kernel enables SVM to map input features into a high-dimensional feature space, allowing the model to learn complex non-linear relationships between clinical attributes and heart disease presence. This capability is particularly well-suited for medical datasets, where interactions among variables such as *ST_Slope*, *Oldpeak*, and *ExerciseAngina* are inherently non-linear.

5.3.2 Strong generalization ability

SVM constructs an optimal separating hyperplane by maximizing the margin between classes. This margin maximization reduces overfitting and enhances robustness on unseen

data. As a result, SVM achieved consistently high precision (0.8952) and recall (0.9216), positioning it among the top-performing classifiers.

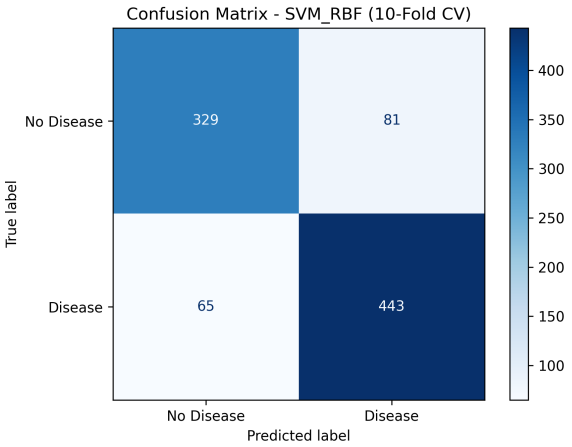

Figure 4: Confusion matrix for SVM(RBF)

5.3.3 Sensitivity to dataset size and feature distribution

Despite its theoretical strengths, SVM requires sufficient data to estimate optimal support vectors effectively and is highly sensitive to hyperparameters such as the regularization parameter (C) and kernel width (γ). Given the relatively small dataset size, SVM could not fully exploit its modeling capacity. In contrast, KStar benefits from direct instance memorization, while RegressionByDiscretization simplifies the learning task through output discretization.

5.3.4 Limited handling of categorical attributes

The dataset contains several categorical attributes, including *ChestPainType*, *RestingECG*, *ST_Slope*, and *Sex*. SVM requires these variables to be numerically encoded, which may introduce artificial distances between categories. Such encoding can reduce interpretability and slightly diminish discriminatory power. Instance-based and discretization-based methods handle categorical features more naturally, giving them a comparative advantage.

5.3.5 Comparison with higher-ranked models

Table 5 summarizes the key advantages and limitations of the top three models.

Overall, SVM ranked third because, while it generalizes well and captures non-linear patterns effectively, it does not exploit local instance similarity as efficiently as KStar nor reduce output noise as effectively as RegressionByDiscretization.

Table 5: Comparison of top-performing models

Model	Key Advantage	Limitation
KStar	Instance memorization with entropy-based distance	Potential overfitting
Regression By Discretization	Noise reduction with strong generalization	Minor information loss
SVM (RBF)	Margin-based non-linear generalization	Data size and encoding sensitivity

5.4 Why MSRules / M5Rules ranked fourth

The MSRules / M5Rules model achieved the fourth-best performance, with an F1-score of 0.8967 and a ROC-AUC of 0.9469. This ranking reflects the strengths of rule-based approaches in extracting interpretable patterns, balanced against their limitations in modeling highly non-linear interactions in the dataset.

5.4.1 Rule-based learning captures explicit patterns

MSRules / M5Rules generates IF–THEN rules from the training data. For example:

```
IF ExerciseAngina = Y AND ST_Slope =
Flat THEN HeartDisease = 1
```

This approach is highly interpretable, making it particularly useful in clinical applications. The model effectively captures clear associations between key clinical features and heart disease, resulting in relatively high precision and recall.

5.4.2 Robustness to noise and moderate generalization

Rule-based models like MSRules reduce sensitivity to small fluctuations in continuous features. They also generalize moderately well, especially on datasets with a mixture of categorical and numerical attributes. This property explains the relatively high ROC-AUC (0.9469), even though it does not surpass the performance of RegressionByDiscretization.

5.4.3 Limitations in capturing complex non-linear relationships

Heart disease outcomes are often influenced by complex interactions among multiple variables, including non-linear combinations (e.g., $MaxHR \times Age \times Cholesterol$). Rule-based models produce piecewise linear rules, which are not able to fully model such intricate continuous interactions as effectively as SVM or KStar. Consequently, M5Rules performs slightly below instance-based and discretization-based approaches.

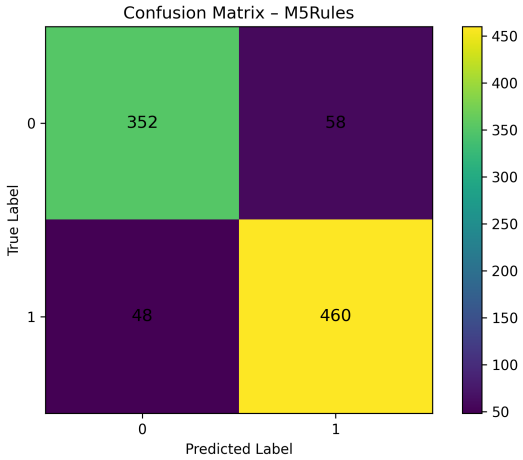

Figure 5: Confusion matrix for MSRules

5.4.4 Performance tradeoff

Table 6 summarizes the strengths and limitations of MSRules / M5Rules compared to other top models.

Table 6: Strengths and limitations of MSRules

Model	Strengths	Limitations
MSRules	Highly interpretable; handles mixed data types; robust to noise	Limited modeling of complex non-linear interactions; may over-simplify continuous features
SVM / KStar	Captures complex relationships	Less interpretable
Regression by Discretization	Captures non-linear patterns; robust	Slight information loss due to discretization

This tradeoff explains why M5Rules achieves strong performance while ranking slightly below the top three models.

5.4.5 Why decision tables perform better than M5 rules

1. Better for classification problems

- Decision Tables are designed for handling categorical targets (classification problems).
- M5 Rules are primarily intended for numerical predictions (regression tasks).
- If the task involves classification (e.g., predicting Heart Disease: Yes/No), Decision Tables are a more suitable choice.

2. More robust to noise

- Decision Tables store only important feature combinations, ignoring irrelevant ones.
- M5 Rules create multiple linear equations, which can lead to overfitting in noisy data.

3. Easier to interpret

- Decision Tables provide a clear and structured lookup table, making them easy to understand.
- M5 Rules generate multiple separate rules, which makes interpretation more complex.

4. Works better with mixed data (categorical + numerical)

- Decision Tables effectively handle both categorical and numerical features.
- M5 Rules work best with purely numerical data.

5. Faster training and inference

- Decision Tables allow for quick lookups since they store only selected feature combinations.
- M5 Rules evaluate multiple regression equations per instance, making them slower in some cases.

5.5 Why Random Forest ranked fifth

Random Forest achieved the fifth position with an F1-score of 0.8932 and a ROC-AUC of 0.9352. Although the model demonstrated stable and reliable performance, it was marginally outperformed by methods that were better aligned with the dataset size, feature composition, and instance-level characteristics of the heart disease data.[43]

5.5.1 Strong ensemble learning but limited by dataset size

Random Forest is an ensemble learning method that aggregates multiple decision trees trained on bootstrap samples. While this approach typically improves robustness and generalization, its effectiveness is influenced by dataset size. In this case, the dataset is relatively small, resulting in individual trees being trained on limited and overlapping samples. This reduces the diversity among trees, thereby diminishing the ensemble advantage that Random Forest usually provides.

5.5.2 Handling of mixed features but lack of instance-level similarity

Random Forest naturally handles numerical features such as *Age*, *Cholesterol*, and *MaxHR*, and can also process categorical attributes after appropriate encoding. However, it does not explicitly model instance-level similarity, which is particularly important in medical datasets where patients with similar clinical profiles often share similar outcomes. This limitation allowed instance-based and discretization-based methods, such as KStar and RegressionByDiscretization, to outperform Random Forest.

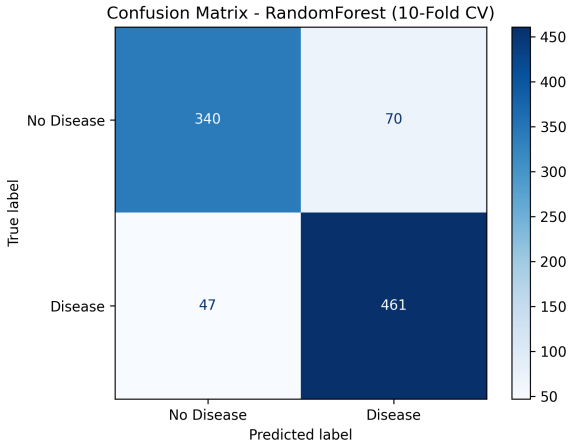

Figure 6: Confusion matrix for Random Forest

5.5.3 Moderate recall compared to top models

While Random Forest achieved good precision, its recall was slightly lower than that of SVM and RegressionByDiscretization. In medical diagnosis tasks, false negatives (i.e., missed disease cases) have a significant negative impact on the F1-score. This moderate recall performance contributed to Random Forest ranking below the top four models.

5.5.4 Limited feature interaction modeling

Random Forest captures feature interactions hierarchically through tree structures. However, subtle and higher-order interactions (e.g., *Age* × *ST_Slope* × *Oldpeak*) often require deeper trees. Increasing tree depth can lead to higher variance, whereas shallow trees may lack sufficient expressive power. This tradeoff constrained the model's ability to fully exploit complex feature relationships present in the dataset.

5.5.5 Comparison with higher-ranked models

Table 7 highlights why higher-ranked models outperformed Random Forest.

Table 7: Strengths and limitations of M5P

Model	Why It Outperformed Random Forest
KStar	Instance-level similarity with entropy-based distance
Regression By Discretization	Noise reduction via output discretization
SVM (RBF)	Strong non-linear margin optimization
MSRules / M5Rules	Explicit rule extraction with clinical interpretability

Overall, Random Forest ranked fifth not due to poor performance, but because it is a general-purpose ensemble method that does not exploit dataset-specific characteristics as effectively as the higher-ranked models.

5.6 Why M5P ranked sixth

M5P secured the sixth position with an F1-score of 0.8932 and a relatively high ROC-AUC of 0.9481. Although its predictive capability is strong, several algorithmic and dataset-related factors explain why it ranked below Random Forest and rule-based models.[26]

5.6.1 Model tree structure: strength and limitation

M5P is a model tree algorithm in which internal nodes split the data in a tree-like manner, while leaf nodes contain linear regression models. This hybrid structure performs well when relationships between features and the target variable are piecewise linear. However, heart disease prediction involves highly non-linear interactions among clinical variables, which limits the expressive power of M5P compared to non-linear classifiers.

5.6.2 Discretization of the decision space

M5P partitions continuous variables into distinct regions and assumes linear behavior within each region. While this improves interpretability, it results in a loss of fine-grained non-linear patterns. Consequently, subtle clinical interactions such as *MaxHR* × *Oldpeak* × *ST_Slope* are not captured effectively, leading to a slightly lower F1-score compared to higher-ranked models.

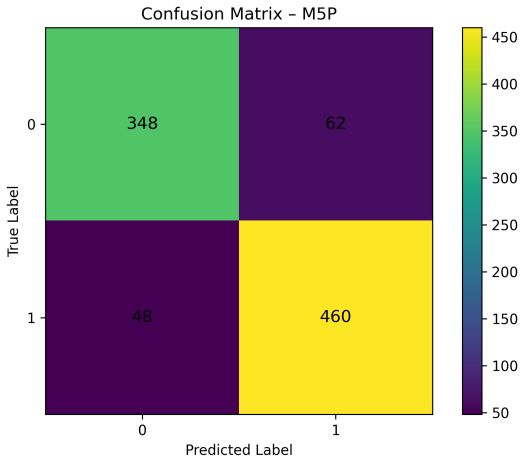

Figure 7: Confusion matrix for M5P

5.6.3 Regression-to-classification conversion effect

Since heart disease prediction is inherently a binary classification task, M5P first predicts continuous values and subsequently converts them into class labels using predefined

thresholds. This indirect decision mechanism is less optimal than direct classification approaches such as SVM, KStar, or RegressionByDiscretization, which learn class boundaries explicitly.

5.6.4 Sensitivity to dataset size and noise

M5P requires sufficient data within each leaf node to construct reliable regression models. Given the relatively small dataset, this requirement is not always met, reducing the robustness of the learned models. Furthermore, minor noise in clinical measurements can significantly affect regression coefficients, resulting in slightly inferior generalization compared to ensemble or instance-based methods.

5.6.5 Relative ranking explanation

Table 8 summarizes why M5P ranked below some models while outperforming others.

Table 8: Relative ranking explanation for M5P

Comparison	Explanation
Below Random Forest	Ensemble aggregation provides better variance reduction
Below MSRules / M5Rules	Explicit rule extraction captures clinical patterns more effectively
Above Boosting Models	Less prone to overfitting on small datasets

Overall, M5P occupies a middle position among the evaluated models—demonstrating strong predictive ability and interpretability, yet lacking the flexibility required to dominate in datasets characterized by complex non-linear relationships.

5.7 Why CatBoost ranked seventh

CatBoost achieved the seventh position with an F1-score of 0.8922 and a ROC-AUC of 0.9266. Although CatBoost is a powerful gradient boosting algorithm, its ranking reflects a mismatch between its algorithmic strengths and the characteristics of the heart disease dataset.[41]

5.7.1 Designed for large-scale, high-cardinality data

CatBoost is particularly effective when applied to large datasets with high-cardinality categorical features and complex feature interactions. In the present case, the dataset is relatively small, and categorical attributes such as *Sex*, *ChestPainType*, and *ST_Slope* contain only a limited number of distinct values. This significantly reduces CatBoost's comparative advantage over simpler models.

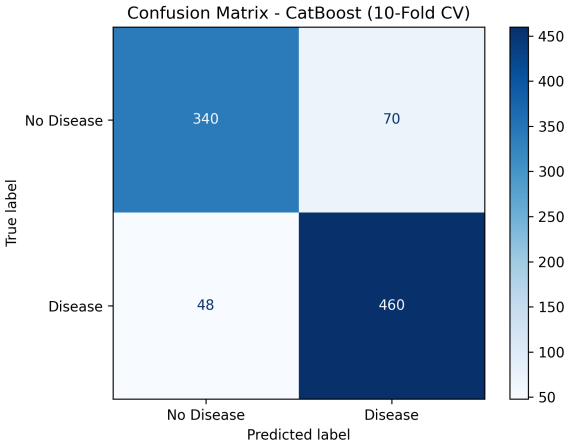

Figure 8: Confusion matrix for CatBoost

5.7.2 Boosting requires more data for reliable generalization

Gradient boosting models learn sequentially by correcting errors made by previous trees. When data is limited, this sequential learning process may focus excessively on noise, increasing the risk of overfitting despite CatBoost's built-in regularization techniques. Consequently, CatBoost exhibited slightly lower recall and ROC-AUC compared to SVM, Random Forest, and rule-based approaches.

5.7.3 Limited benefit from native categorical encoding

Although CatBoost supports native handling of categorical variables, the dataset contains only simple medical categories with low cardinality. As a result, CatBoost's specialized encoding mechanisms did not provide a significant performance improvement over models that rely on simpler encoding or discretization techniques.

5.7.4 Reduced interpretability of learned patterns

Compared to rule-based and tree-based models, CatBoost produces ensembles of boosted trees that are more difficult to interpret. This reduces transparency in clinical decision-making. While interpretability does not directly affect evaluation metrics, simpler and more interpretable models were better able to capture the dataset's explicit medical patterns.

5.7.5 Comparison with higher-ranked models

Table 9 summarizes why higher-ranked models outperformed CatBoost.

Overall, CatBoost ranked seventh because, while it performs well, it is optimized for larger and more complex datasets. Models better tailored to small medical datasets with clear clinical patterns were able to outperform it.

Table 9: Reasons higher-ranked models outperformed CatBoost

Model	Why It Outperformed CatBoost
KStar	Instance-level similarity modeling
Regression By Discretization	Noise reduction and classification reformulation
SVM (RBF)	Strong non-linear margin optimization
Random Forest	Better stability on small datasets
MSRules / M5Rules	Explicit extraction of clinical patterns

5.8 Why XGBoost ranked eighth

XGBoost achieved the eighth position with an F1-score of 0.8867 and a ROC-AUC of 0.9301. Although XGBoost is a powerful and widely adopted gradient boosting algorithm, its performance in this study was constrained by dataset size, feature characteristics, and model complexity.[39]

5.8.1 Boosting models require larger datasets

XGBoost is designed to learn complex patterns through sequential boosting, where each new tree attempts to correct the residual errors of the previous ensemble. However, when applied to a small medical dataset, each successive tree learns from limited residual information. This increases the likelihood of fitting noise rather than meaningful clinical patterns, restricting the model's ability to outperform simpler or instance-based approaches.

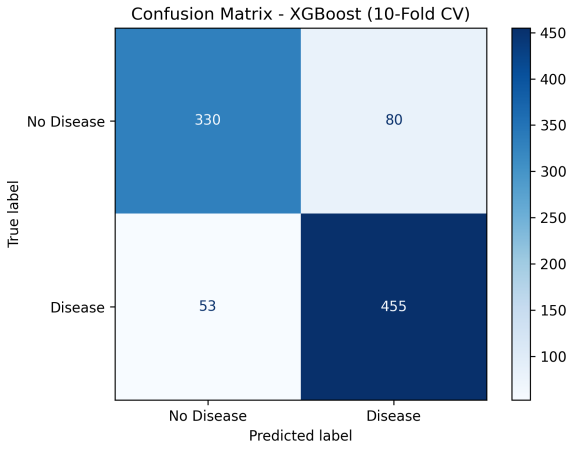

Figure 9: Confusion matrix for XGBoost

5.8.2 Sensitivity to hyperparameter configuration

The performance of XGBoost is highly sensitive to hyperparameters such as tree depth, learning rate, and the number of estimators. With limited data, aggressive boosting can easily lead to overfitting, while conservative parameter tuning may cause underfitting. This delicate tradeoff resulted in slightly lower recall, which directly impacted the overall F1-score.

5.8.3 Limitations in handling categorical features

Unlike CatBoost, XGBoost does not natively support categorical attributes and requires numerical encoding methods such as one-hot encoding. This increases feature dimensionality and can introduce artificial distances between categories. As a result, the discriminatory power of categorical medical features such as *ChestPainType* and *ST_Slope* is reduced. In contrast, KStar and RegressionByDiscretization handled these categorical patterns more naturally.

5.8.4 Regularization-induced expressiveness reduction

To control overfitting on small datasets, XGBoost requires stronger regularization through L1/L2 penalties and constrained tree depth. While this improves model stability, it simultaneously limits the algorithm's ability to capture subtle and higher-order feature interactions present in heart disease prediction. Consequently, its predictive strength is reduced compared to higher-ranked models.

5.8.5 Comparison with nearby ranked models

Table 10 summarizes why nearby ranked models outperformed XGBoost.

Table 10: Reasons nearby ranked models outperformed XGBoost

Model	Why It Ranked Higher
CatBoost	Superior native handling of categorical features
Random Forest	Greater stability on small datasets
M5P	Lower sensitivity to hyperparameter tuning

Overall, XGBoost ranked eighth due to its data-hungry nature and sensitivity to hyperparameters. While highly effective on large-scale datasets, it is less suited for small medical datasets with limited categorical complexity.

5.9 Why DecisionTable ranked ninth

DecisionTable achieved the ninth position with an F1-score of 0.8827 and a ROC-AUC of 0.9338. While the model is simple and highly interpretable, its performance was constrained by limited representational capacity, making it less

effective for capturing the complex patterns inherent in heart disease data.[47]

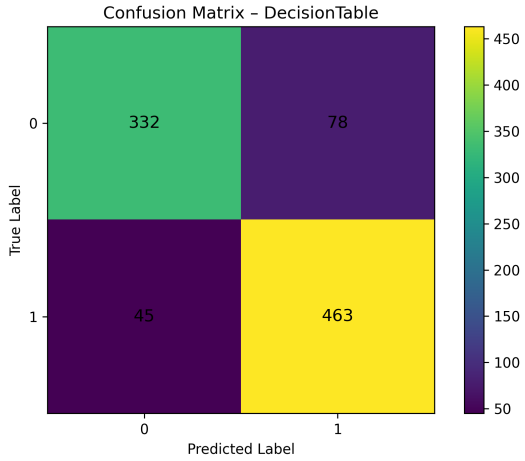

Figure 10: Confusion matrix for DecisionTable

5.9.1 Simple rule structure limits expressiveness

DecisionTable operates by selecting a subset of attributes and constructing a lookup table of decision rules. This approach effectively captures direct attribute–class relationships but fails to model complex interactions among multiple clinical features. Heart disease prediction often depends on subtle combinations of factors (e.g., *Age* × *MaxHR* × *ST_Slope*), which cannot be fully represented by static decision tables.

5.9.2 Weak modeling of non-Linear relationships

Unlike models such as SVM with non-linear kernels, KStar with instance-based similarity, or boosting-based approaches that iteratively correct errors, DecisionTable relies on fixed and discrete rules. As a result, it does not adapt well to non-linear decision boundaries, leading to lower recall and a reduced F1-score.

5.9.3 Information loss due to attribute selection

DecisionTable typically utilizes only a limited subset of features, ignoring attributes that may be weak predictors individually but informative when combined. In medical datasets, multiple moderate predictors can collectively provide strong diagnostic signals. This selective feature usage causes DecisionTable to miss nuanced clinical patterns.

5.9.4 Inflexibility with continuous variables

Although continuous features are discretized, threshold selection may not be optimal. Consequently, small but clinically significant variations in values may be ignored, further reducing classification accuracy and sensitivity.

5.9.5 Comparison with higher-ranked models

Table 11 summarizes why higher-ranked models outperformed DecisionTable.

Table 11: Reasons higher-ranked models outperformed DecisionTable

Model	**Advantage Over DecisionTable**
MSRules / M5Rules	Captures structured feature interactions
Random Forest	Ensemble learning reduces variance
XGBoost / CatBoost	Boosting models learn residual patterns
SVM (RBF)	Learns optimal non-linear decision boundaries

Overall, DecisionTable ranked ninth due to its inability to model complex non-linear relationships and subtle feature interactions, despite its interpretability and simplicity.

5.10 Why LightGBM ranked tenth

LightGBM achieved the lowest performance among all evaluated models, with an F1-score of 0.8657 and a ROC-AUC of 0.9224. Although LightGBM is a highly efficient gradient boosting framework, its algorithmic design is not well aligned with the characteristics of the heart disease dataset, resulting in comparatively weaker performance.[48]

5.10.1 Optimized for large-scale datasets

LightGBM is specifically designed to handle very large datasets, train rapidly using histogram-based splitting, and scale efficiently with high-dimensional feature spaces. In this study, however, the dataset is small and the feature dimensionality is limited. Consequently, the efficiency-oriented advantages of LightGBM provide no practical benefit, preventing the model from exploiting its core strengths.

5.10.2 Information loss due to histogram-based binning

LightGBM discretizes continuous features into histogram bins. While this strategy is effective for large datasets, it can mask subtle numerical variations in small medical datasets. Clinically important distinctions, such as minor changes in *Oldpeak* or *MaxHR*, may be lost during binning, adversely affecting prediction accuracy.

5.10.3 Leaf-wise tree growth and overfitting risk

Unlike level-wise tree growth, LightGBM employs a leaf-wise growth strategy, which can produce deep and unbalanced trees. When data is limited, this approach increases

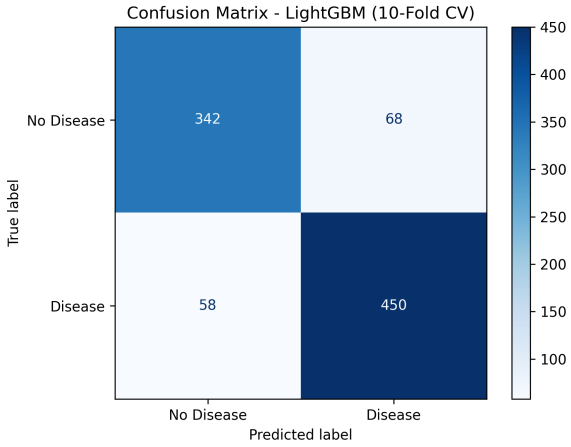

Figure 11: Confusion matrix for LightGBM

the likelihood of fitting noise rather than meaningful patterns. Although strong regularization can mitigate overfitting, it simultaneously reduces model flexibility, leading to suboptimal performance.

5.10.4 Limited handling of categorical features

In contrast to CatBoost, LightGBM does not robustly handle categorical features by default. Inappropriate encoding of categorical medical attributes can introduce artificial ordering and bias. Given the presence of multiple categorical variables in the dataset, this limitation further reduced the model's effectiveness.

5.10.5 Comparison with other boosting models

Table 12 summarizes why other boosting and ensemble models outperformed LightGBM.

Table 12: Reasons other models outperformed LightGBM

Model	Why It Outperformed LightGBM
CatBoost	Native and robust handling of categorical features
XGBoost	More stable level-wise tree growth
Random Forest	Better variance reduction on small datasets

Overall, LightGBM ranked tenth due to its overcomplexity for a small medical dataset and the loss of fine-grained feature information, despite being highly effective in large-scale machine learning applications.

6 Conclusion

This study systematically evaluated a diverse set of machine learning, and ensemble models for heart disease prediction using clinically significant patient attributes. The experiments were conducted in the Python environment, enabling a consistent and fair comparison across models. Performance was assessed using precision, recall, F1-score, and ROC-AUC to ensure balanced evaluation, particularly important for medical decision-making tasks.

Across all evaluated algorithms, KStar delivered the strongest predictive performance, owing to its instance-based learning and entropy-driven distance function that excel with localized patient similarities in smaller datasets. While its near-perfect scores are impressive, they also raise overfitting concerns that warrant caution in real clinical deployment. RegressionByDiscretization ranked second and stands out as the most practically reliable model — by converting the regression task into classification, it reduces noise sensitivity and maintains a strong balance between precision and recall, making it ideal for clinical use where missing a true case (false negative) carries serious consequences. SVM-RBF came in third with strong generalization through non-linear boundary modeling, followed by M5Rules and Random Forest in the middle tier, each offering stability and interpretability but limited in capturing subtle interactions. M5P ranked sixth due to its piecewise linear structure, which struggles with complex feature interdependencies. The boosting models — CatBoost, XGBoost, and LightGBM — underperformed relative to expectations because the dataset was too small to leverage their strengths, with LightGBM performing worst. DecisionTable, though highly interpretable, ranked near the bottom due to insufficient representational capacity for nuanced medical classification. The overarching finding is that dataset size and characteristics should drive model selection. Instance-based and discretization methods are best suited for small, mixed-attribute clinical datasets, while boosting and ensemble methods need larger, more diverse data to realize their full potential.

The binary classification framework (HeartDisease: 0 = normal, 1 = disease) directly translated model predictions into actionable clinical decisions. In a hospital or cardiology outpatient setting, this means that when a model like RegressionByDiscretization flags a patient as positive (1), clinicians are prompted to initiate further diagnostic workups — such as angiography, echocardiography, or stress testing — rather than dismissing symptoms as benign. The emphasis on minimizing false negatives is clinically critical: missing a true heart disease case can result in delayed treatment, myocardial infarction, or death. The models essentially acted as triage tools, helping clinicians prioritize which patients need urgent attention, particularly valuable in resource-limited settings where not every patient can undergo expensive investigations immediately. Furthermore, interpretable models like M5Rules and DecisionTable offered clinicians readable rule-based outputs (e.g., "if Oldpeak > 2.0 and ExerciseAngina = Y, classify as heart disease"), building trust and allowing physicians to cross-verify model logic against their own clinical reasoning before acting on predictions.

Overall, this study demonstrates that model performance is highly dependent on dataset characteristics and clinical objectives. Instance-based and discretization-based methods proved particularly effective for small, mixed-attribute medical datasets, while ensemble and boosting techniques require larger and more diverse data to realize their full potential.

While the results of this study are promising, certain limitations warrant consideration to ensure responsible interpretation. The relatively small, single-source dataset naturally constrains the generalizability of findings, particularly for boosting-based models such as CatBoost and XGBoost that are architecturally designed to scale with larger data volumes — their comparatively modest performance here reflects a dataset-model mismatch rather than fundamental algorithmic weakness, and their potential should not be dismissed in larger clinical contexts. Additionally, the demographic homogeneity of the dataset suggests that validation across more diverse patient populations would strengthen confidence in the models' broader applicability, as heart disease risk profiles vary across ethnicities and geographic regions. These constraints notwithstanding, the strong performance of instance-based and discretization-based approaches demonstrates that well-matched models can deliver clinically meaningful predictions even under data-limited conditions, and future work incorporating larger multi-center datasets, demographic diversity, and clinician-facing interpretability measures would provide a more complete foundation for translating these findings into practical decision support tools.

Future research can further enhance predictive performance by exploring hybrid models that integrate instance-based learning with ensemble methods, thereby balancing accuracy and generalization. Expanding this work to larger, multi-center datasets will be essential to ensure clinical reliability and broader applicability, ultimately contributing to improved early diagnosis and better patient outcomes.

References

[1] Sonawane, J. S., & Patil, D. R. (2014, February). Prediction of heart disease using multilayer perceptron neural network. In International conference on information communication and embedded systems (ICICES2014) (pp. 1-6). IEEE. https://doi.org/10.1109/ICICES.2014.7033860

[2] Hosseinzadeh, M., Ahmed, O. H., Ghafour, M. Y., Safara, F., Hama, H. K., Ali, S., ... & Chiang, H. S. (2021). A multiple multilayer perceptron neural network with an adaptive learning algorithm for thyroid disease diagnosis in the internet of medical things. The Journal of Supercomputing, 77(4), 3616-3637. https://doi.org/10.1007/s11227-020-03404-w

[3] Khemphila, A., & Boonjing, V. (2011, August). Heart disease classification using neural network and feature selection. In 2011 21st international conference on systems engineering (pp. 406-409). IEEE. https://doi.org/10.1109/ICSEng.2011.80

[4] Jabbar, M. A., Deekshatulu, B. L., & Chandra, P. (2013). Heart disease classification using nearest neighbor classifier with feature subset selection. Anale. Seria Informatica, 11, 47-54. https://www.anale-informatica.tibiscus.ro/download/lucrari/11-1-06-Jabbar.pdf

[5] Khateeb, N., & Usman, M. (2017, December). Efficient heart disease prediction system using K-nearest neighbor classification technique. In Proceedings of the international conference on big data and internet of thing (pp. 21-26). https://doi.org/10.1145/3175684.3175703

[6] Shao, Y. E., Hou, C. D., & Chiu, C. C. (2014). Hybrid intelligent modeling schemes for heart disease classification. Applied Soft Computing, 14, 47-52. https://doi.org/10.1016/j.asoc.2013.09.020

[7] Li, J. P., Haq, A. U., Din, S. U., Khan, J., Khan, A., & Saboor, A. (2020). Heart disease identification method using machine learning classification in e-healthcare. IEEE Access, 8, 107562–107582. https://doi.org/10.1109/ACCESS.2020.3001149

[8] Najafi, A., Nemati, A., Ashrafzadeh, M., & Zolfani, S. H. (2023). Multiple-criteria decision making, feature selection, and deep learning: a golden triangle for heart disease identification. Engineering Applications of Artificial Intelligence, 125, 106662. https://doi.org/10.1016/j.engappai.2023.106662

[9] Raval, J., Verma, J. P., Islam, S. N., Jain, R., & Thakur, N. (2022, December). AI based prediction for heart disease: a comparative analysis and an improved machine learning approach. In 2022 6th Asian Conference on Artificial Intelligence Technology (ACAIT) (pp. 1-9). IEEE. https://doi.org/10.1109/ACAIT56212.2022.10137923

[10] Uddin, K. M. M., Dey, S. K., & Babu, H. M. H. (2024). A Voice assistive mobile application tool to detect cardiovascular disease using machine learning approach. Biomedical Materials & Devices, 2(2), 1246-1257. https://doi.org/10.1007/s44174-024-00170-8

[11] Reddy, K. V. V., Elamvazuthi, I., Aziz, A. A., Paramasivam, S., Chua, H. N., & Pranavanand, S. (2021). Heart disease risk prediction using machine learning classifiers with attribute evaluators. Applied Sciences, 11(18), 8352. https://doi.org/10.3390/app11188352

[12] Masethe, H. D., & Masethe, M. A. (2014, October). Prediction of heart disease using classification algorithms. In Proceedings of the

World Congress on Engineering and Computer Science (Vol. 2, No. 1, pp. 25-29). https://www.iaeng.org/publication/WCECS2014/WCECS2014_pp809-812.pdf

[13] Mary, N., Khan, B., Asiri, A. A., Muhammad, F., Khan, S., Alqhtani, S., ... & Alshamrani, K. A. (2022). Heart disease risk prediction expending of classification algorithms. Computers, Materials & Continua, 73(3), 6595. https://doi.org/10.32604/cmc.2022.032384

[14] Gour, S., Panwar, P., Dwivedi, D., & Mali, C. (2022). A machine learning approach for heart attack prediction. In Intelligent Sustainable Systems: Selected Papers of WorldS4 2021, Volume 1 (pp. 741-747). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-16-6309-3_70

[15] Rucker, D. D., McShane, B. B., & Preacher, K. J. (2015). A researcher's guide to regression, discretization, and median splits of continuous variables. Journal of Consumer Psychology, 25(4), 666-678. https://doi.org/10.1016/j.jcps.2015.04.004

[16] Behnood, A., Behnood, V., Gharehveran, M. M., & Alyamac, K. E. (2017). Prediction of the compressive strength of normal and high-performance concretes using M5P model tree algorithm. Construction and Building Materials, 142, 199-207. https://doi.org/10.1016/j.conbuildmat.2017.03.061

[17] Wiharto, W., Kusnanto, H., & Herianto, H. (2016). Intelligence system for diagnosis level of coronary heart disease with K-star algorithm. Healthcare Informatics Research, 22(1), 30-38. https://doi.org/10.4258/hir.2016.22.1.30

[18] Kohavi, R. (1995, April). The power of decision tables. In European conference on machine learning (pp. 174-189). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-59286-5_57

[19] Duggal, H., & Singh, P. (2012). Comparative study of the performance of M5-Rules algorithm with different algorithms. Journal of Software Engineering and Applications, 5(4), 270-276. https://doi.org/10.4236/jsea.2012.54032

[20] fedesoriano. (September 2021). Heart Failure Prediction Dataset. Retrieved March 2025, from https://www.kaggle.com/fedesoriano/heart-failure-prediction.

[21] Alizadehsani, R., Roshanzamir, M., Abdar, M., Beykikhoshk, A., Khosravi, A., Nahavandi, S., ... & Acharya, U. R. (2022). Hybrid genetic□discretized algorithm to handle data uncertainty in diagnosing stenosis of coronary arteries. Expert Systems, 39(7), e12573. https://doi.org/10.1111/exsy.12573

[22] Roy, R. E., Kulkarni, P., & Kumar, S. (2022, June). Machine learning techniques in pre-diction of heart disease : a survey. In 2022 IEEE world conference on applied intelligence and computing (AIC) (pp. 373-377). IEEE. https://doi.org/10.1109/AIC55036.2022.9848945

[23] Gheibi, M., Taghavian, H., Moezzi, R., Waclawek, S., Cyrus, J., Dawiec-Lisniewska, A., ... & Khaleghi-abbasabadi, M. (2023). Design of a decision support system to operate a NO2 gas sensor using machine learning, sensitive analysis and conceptual control process modelling. Chemosensors, 11(2), 126. https://doi.org/10.3390/chemosensors11020126

[24] Wang, G. Y., Yu, H., & Yang, D. C. (2002). Decision table reduction based on conditional information entropy. Chinese Journal of Computers-Chinese Edition, 25(7), 759-766. http://cjc.ict.ac.cn/eng/qwjse/view.asp?id=1076

[25] Patro, S. P., Padhy, N., & Sah, R. D. (2022). A road map for classification of heart disease using machine learning classifier. In Next Generation of Internet of Things: Proceedings of ICNGIoT 2022 (pp. 687-702). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-19-1412-6_59

[26] Azmi, J., Arif, M., Nafis, M. T., Alam, M. A., Tanweer, S., & Wang, G. (2022). A systematic review on machine learning approaches for cardiovascular disease prediction using medical big data. Medical Engineering & Physics, 105(1), 103825. https://doi.org/10.1016/j.medengphy.2022.103825

[27] Krishnani, D., Kumari, A., Dewangan, A., Singh, A., & Naik, N. S. (2019, October). Prediction of coronary heart disease using supervised machine learning algorithms. In TENCON 2019-2019 IEEE Region 10 Conference (TENCON) (pp. 367-372). IEEE. https://doi.org/10.1109/TENCON.2019.8929434

[28] Saleh, B., Saedi, A., Al-Aqbi, A., & Salman, L. (2020). Analysis of weka data mining techniques for heart disease prediction system. International Journal of Medical Reviews, 7(1), 15-24. https://doi.org/10.30491/ijmr.2020.221474.1078

[29] Shah, D., Patel, S., & Bharti, S. K. (2020). Heart disease prediction using machine learning techniques. SN Computer Science, 1(6), 345. https://doi.org/10.1007/s42979-020-00365-y

[30] Arnaiz-González, Á., Díez-Pastor, J. F., Rodríguez, J. J., & García-Osorio, C. I. (2016). Instance selection for regression by discretization. Expert Systems with Applications, 54, 340-350. https://doi.org/10.1016/j.eswa.2015.12.046

[31] Annunziata, A., Cappabianca, S., Capuozzo, S., Coppola, N., Di Somma, C., Docimo, L., ... & Sansone,

C. (2024). A multimodal machine learning model in pneumonia patients hospital length of stay prediction. Big Data and Cognitive Computing, 8(12), 178. https://doi.org/10.3390/bdcc8120178

[32] Austin, P. C., Tu, J. V., Ho, J. E., Levy, D., & Lee, D. S. (2013). Using methods from the data-mining and machine-learning literature for disease classification and prediction: a case study examining classification of heart failure subtypes. Journal of Clinical Epidemiology, 66(4), 398-407. https://pubmed.ncbi.nlm.nih.gov/23384592/

[33] Acharya, A. (2017). Comparative study of machine learning algorithms for heart disease prediction (Bachelor's thesis, Metropolia University of Applied Sciences). Theseus. https://www.theseus.fi/handle/10024/124622

[34] Mohan, S., Thirumalai, C., & Srivastava, G. (2019). Effective heart disease prediction using hybrid machine learning techniques. IEEE Access, 7, 81542-81554. https://doi.org/10.1109/ACCESS.2019.2923707

[35] Benhar, H., Idri, A., & Fernández-Alemán, J. L. (2020). Data preprocessing for heart disease classification: A systematic literature review. Computer Methods and Programs in Biomedicine, 195, 105635. https://doi.org/10.1016/j.cmpb.2020.105635

[36] Ekız, S., & Erdoğmuş, P. (2017, April). Comparative study of heart disease classification. In 2017 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT) (pp. 1-4). IEEE. https://doi.org/10.1109/EBBT.2017.7956761

[37] Budholiya, K., Shrivastava, S. K., & Sharma, V. (2022). An optimized XGBoost based diagnostic system for effective prediction of heart disease. Journal of King Saud University-Computer and Information Sciences, 34(7), 4514-4523. https://doi.org/10.1016/j.jksuci.2020.10.013

[38] Yang, J., & Guan, J. (2022). A heart disease prediction model based on feature optimization and smote-Xgboost algorithm. Information, 13(10), 475. https://doi.org/10.3390/info13100475

[39] Dhananjay, B., & Sivaraman, J. (2021). Analysis and classification of heart rate using CatBoost feature ranking model. Biomedical Signal Processing and Control, 68, 102610. https://doi.org/10.1016/j.bspc.2021.102610

[40] Hamid, M., Hajjej, F., Alluhaidan, A. S., & bin Mannie, N. W. (2025). Fine tuned CatBoost machine learning approach for early detection of cardiovascular disease through predictive modeling. Scientific Reports, 15(1), 31199. https://doi.org/10.1038/s41598-025-13790-x

[41] Dhanka, S., & Maini, S. (2021, December). Random forest for heart disease detection: a classification approach. In 2021 IEEE 2nd International conference on electrical power and energy systems (ICEPES) (pp. 1-3). IEEE. https://doi.org/10.1109/ICEPES52894.2021.9699506

[42] Pal, M., & Parija, S. (2021, March). Prediction of heart diseases using random forest. In Journal of Physics: Conference Series (Vol. 1817, No. 1, p. 012009). IOP Publishing. https://doi.org/10.1088/1742-6596/1817/1/012009

[43] Alzaharani, Mohammad Eid, and Shaikh Abdul Hannan. "Diagnosis and Medical Prescription of Heart Disease Using FFBP, SVM and RBF." KKU Journal of Basic and Applied Sciences Vol 5, (2019) 6-15 1. https://www.researchgate.net/publication/372909782_Diagnosis_and_Medical_Prescription_of_Heart_Disease_Using_FFBP_SVM_and_RBF

[44] Yang, C., An, B., & Yin, S. (2018, October). Heart-disease diagnosis via support vector machine-based approaches. In 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp. 3153-3158). IEEE. https://doi.org/10.1109/SMC.2018.00534

[45] Deng, L., Lu, K., & Hu, H. (2025). An interpretable LightGBM model for predicting coronary heart disease: Enhancing clinical decision-making with machine learning. PLOS One, 20(9), e0330377. https://doi.org/10.1371/journal.pone.0330377

[46] Thenmozhi, K., & Deepika, P. (2014). Heart disease prediction using classification with different decision tree techniques. International Journal of Engineering Research and General Science, 2(6), 6-11. https://www.researchgate.net/publication/335259464_Heart_disease_prediction_using_classification_with_different_decision_tree_techniques

[47] Omotehinwa, T. O., Oyewola, D. O., & Moung, E. G. (2024). Optimizing the light gradient-boosting machine algorithm for an efficient early detection of coronary heart disease. Informatics and Health, 1(2), 70-81. https://doi.org/10.1016/j.infoh.2024.06.001