Heart Disease Prediction Using Machine Learning Approach with Data Preprocessing Optimization

Akhsanul Khuluq
Directorate of Academics and Admissions, Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
khuluqahsan237@gmail.com

Abstract— Heart disease is one of the leading causes of death in the world. Hence, the development of early prediction systems based on technology has become an important need to support medical diagnosis. Several machine learning algorithms have been applied to heart disease prediction, but their performance is highly dependent on data quality, preprocessing, and the suitable selection of the algorithm. However, prior studies have focused more on the application of specific algorithms and have not studied in detail the comparative influence of data preprocessing and optimization on the performance of different classification models. This study proposes a comparative approach of five machine learning algorithms, namely Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbor (KNN), and Support Vector Machine (SVM), combined with data preprocessing through data cleaning, outlier handling using Interquartile Range (IQR), and Min-Max Scaling normalization. The dataset used contains 270 medical records of patients with heart disease, each with 13 clinical symptoms. The dataset was divided into 80% training data and 20% testing data for the model training and evaluation process. Evaluation metrics used to assess the performance of each algorithm include accuracy, precision, recall, and F1-score. By analyzing the resulting data, the most effective algorithm in predicting heart disease based on the given clinical attributes was determined. The evaluation was conducted using the accuracy, precision, recall, and F1-score matrices. The testing results indicate that the K-Nearest Neighbor (KNN) algorithm provides the best performance with an accuracy of 93%, followed by Logistic Regression at 91%, while the Decision Tree has the lowest performance. The research results also show that the preprocessing stage also plays a significant role in improving the stability of data and classification model performance, especially for algorithms that are sensitive to the scale of features and the distribution of data. This study shows that by integrating preprocessing and choosing the right algorithm, it can increase the accuracy of heart disease prediction and has the potential to support the development of AI-based clinical decision support systems.

Keywords—Heart disease prediction; Machine learning; K-Nearest Neighbor (KNN); Support Vector Machine (SVM); Preprocessing data; Clinical decision support system

I. INTRODUCTION

Cardiovascular disease remains one of the leading causes of death worldwide and a major challenge to the global health system. Cardiovascular disease is an important contributor to global mortality, according to the World Health Organization, and early detection is an important step to reducing the risk of complications and death [1]. Implementing effective screening programs and promoting heart-healthy lifestyles can significantly lower the incidence of these diseases. Furthermore, increasing public awareness about risk factors such as diet, exercise, and smoking cessation is crucial in the fight against cardiovascular disease [2], [3], [4].

Several classification algorithms, such as Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbor (KNN), and Support Vector Machine (SVM), have been used to predict heart disease. Several studies have proven that those algorithms have competitive performance, although the results obtained vary depending on the characteristics of the dataset used[5], [6]. Ekong [7]demonstrated that the Random Forest and Support Vector Machine-based model achieved high accuracy in early diagnosis of heart disease. Meanwhile, Dayana et al. [8] found that K-Nearest Neighbor has a good performance on medical data with a strong feature proximity pattern. In the national context, Miftakhudin et al. [9], Damayanti et al. [6], and Hikmayanti et al. [10] also revealed that the machine learning-based classification approach has the potential to help data-based diagnosis systems in the health domain.

Data quality is crucial for predictive model performance. Algorithm selection is also crucial. Clinical data often contains outliers, imbalanced features, and varying attribute scales. These issues can degrade classification accuracy. Existing research recommends preprocessing methods, including normalization, outlier handling, and feature selection. Solution to increase the model performance [11], [12]. Studies by Setiawan et al. [13] and Miftakhudin et al. [9] have shown that proper preprocessing can improve model stability, especially in determining distance-based algorithms such as K-Nearest Neighbor and Support Vector Machine..

Despite the rapid growth of research on heart disease prediction, there are still some knowledge gaps that need to be filled. Most of the previous studies only focus on the application of a certain algorithm without any comprehensive multi-algorithm comparison. In addition, many research works focus more on improving the accuracy of the model without evaluating the contribution of preprocessing and data optimization stages to the performance of each algorithm [11]. Moreover, the previous works have also lacked the integration of outlier handling, data normalization, and multi-model evaluation in a systematic experimental framework, especially in the heart disease prediction studies based on clinical data. Thus, there has not been a comprehensive study to show the effect of preprocessing on the stability and performance of various classification algorithms on a heart disease dataset [11].

Based on the identified knowledge gap, this study proposes a comparative approach to five machine learning algorithms—Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbor (KNN), and Support Vector Machine (SVM)—integrated with data preprocessing stages through data cleaning, outlier handling using the Interquartile Range (IQR) method, and Min-Max Scaling normalization. The novelty of this study lies in integrating IQR- and Min-Max Scaling-based preprocessing with a multi-algorithm comparative study to obtain an optimal heart disease prediction model. This approach differs from previous studies that generally focus on a single model or do not systematically evaluate the impact of preprocessing on the performance of classification algorithms [14], [15].

This study aims to develop a machine learning-based heart disease prediction model by comparing the performance of five classification algorithms, namely Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbor (KNN), and Support Vector Machine (SVM). This study analyzes the influence of pre-processing stages, including data cleaning, outlier handling using the Interquartile Range (IQR) method, and Min-Max Scaling normalization, on producing model accuracy and stability. The main contribution of this study lies in the systematic evaluation of the influence of pre-processing on classification performance and a comparative analysis of several algorithms on heart disease datasets. The results of this study are expected to produce a predictive model that is not only accurate but also suitable for use as a support system in artificial intelligence-based decision-making for the early diagnosis of heart disease [16], [17].

This article is organized as follows. The second section outlines the research method, detailing the dataset, preprocessing steps, classification algorithms, and model evaluation procedures. The third section presents the experimental results along with a thorough discussion. The fourth section offers conclusions and suggests potential directions for future research.

II. RESEARCH METHODS

2.1 Research Design

This study used a quantitative method to compare machine learning algorithms. It focused on predicting heart disease. The research involved several stages: collecting the dataset, preprocessing, splitting the data into training and test sets, training the model, evaluating performance, and comparing the results. Figure 1 shows the flow of the research.

2.2. Dataset Acquisition

This study utilizes the Heart Disease Dataset sourced from a public repository on Kaggle https://www.kaggle.com/datasets/shivanshdwivedi/heart-disease-prediction. This dataset was selected due to its clinical attributes that are pertinent to heart disease prediction and its frequent use in prior research, which enhances the validity of our experiment. The dataset consists of 270 patient data points with 13 independent attributes and one dependent attribute as the classification target. Independent attributes include medical parameters such as age, blood pressure, cholesterol, maximum heart rate, chest pain type, exercise angina, and several other clinical attributes. The target variable is classified into two classes, namely presence (1) for patients indicated by heart disease and absence (0) for patients not indicated by heart disease. The acquisition process is carried out by importing the dataset into the Python computing environment using the Pandas library. Next, data structure examination, attribute type validation, missing value identification, and initial data distribution analysis are carried out before entering the preprocessing and classification modeling stages.

Fig. 1. Research Flowchart

2.3. Data Preprocessing

The preprocessing stage is carried out to improve the quality of the dataset before the classification process. The process includes data normalization, outlier handling, and feature selection. Normalization uses the Min-Max Scaling method to equalize the range of feature values to a 0–1 scale to prevent the dominance of certain attributes in distance-based algorithms such as KNN and SVM. Outlier handling is carried out using the Interquartile Range (IQR) method with a capping technique to reduce the influence of extreme values without deleting data. In addition, feature selection is carried out to identify the most relevant attributes to the classification target based on correlation analysis and feature contributions to the model. This preprocessing stage aims to improve data stability and the performance of the heart disease prediction model.

2.4. Exploratory Data Analysis (EDA)

The Exploratory Data Analysis (EDA) phase provided insights into the dataset's characteristics prior to modeling. We evaluated the distributions of numerical features through histograms and boxplots, and examined relationships between features using correlation heatmaps. The EDA identified that several features, especially cholesterol and blood pressure, were skewed or had outliers. Furthermore, variables like Chest Pain Type and Maximum Heart Rate showed a correlation with the target heart disease classification. This stage guided our preprocessing and model selection decisions.

2.5. Training-testing Data Sharing

After the preprocessing stage is complete, the dataset is split into training and testing data using the train-test split method. The data is divided into 80% for training and 20% for testing. The training data trains the machine learning model to learn patterns between features, while the testing data evaluates the model's ability to predict new, unseen data. To keep the

results consistent and reduce bias, the data is split randomly with a fixed state.

2.6. Implementation of The Classification Model

The model implementation phase involved five machine learning algorithms: Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbor (KNN), and Support Vector Machine (SVM). We started by training each algorithm with the preprocessed training data. Once training was completed, we used the models to make predictions on the testing data. We then compared the prediction results with the actual labels to evaluate the performance of each algorithm. Each model was evaluated using accuracy, precision, recall, F1-score, and confusion matrix metrics to determine the model's ability to classify patients with and without indications of heart disease. This process aimed to determine the algorithm with the best performance based on the evaluation results on the test data.

2.7. Model Evaluation

The evaluation phase was conducted to measure the performance of each classification algorithm in predicting heart disease. After the model generated predictions on the testing data, the prediction results were compared with the actual labels using a confusion matrix. Based on the confusion matrix, several evaluation metrics were calculated, namely accuracy, precision, recall, and F1-score. Accuracy was used to measure the overall level of prediction accuracy; precision was used to measure the accuracy of positive class predictions; recall was used to determine the model's ability to detect positive cases; while the F1-score was used to assess the balance between precision and recall. We measured overall prediction performance using accuracy, while precision focused on the accuracy of positive class predictions. Recall evaluated the model's ability to identify positive cases, and the F1-score provided insight into the balance between precision and recall. After evaluating all algorithms, we compared the results to find the model that excelled in predicting heart disease.

2.8. Comparison of Results and Analysis

The comparison phase involved evaluating the performance of various classification algorithms using the results from the testing data. We analyzed the accuracy, precision, recall, and F1-score of each model to identify their strengths and weaknesses in predicting heart disease. Additionally, a confusion matrix helped us uncover patterns of misclassification, including false positives and false negatives. We also examined the impact of preprocessing techniques, such as outlier handling and normalization, on enhancing model performance. From these comparisons, we identified the best-performing algorithm, which demonstrated the highest accuracy and predictive stability. This phase also allowed us to draw conclusions and address our research objectives regarding the effectiveness of machine learning algorithms in predicting heart disease.

III. RESULTS AND DISCUSSION

3.1 Exploratory Data Analysis (EDA) Results

The Exploratory Data Analysis (EDA) phase was conducted to comprehensively understand the underlying data characteristics, distributions, and relationships prior to the modeling stage. To achieve this, various visual analyses were performed, including histograms, boxplots, and correlation heatmaps.

1) Data Distribution Analysis

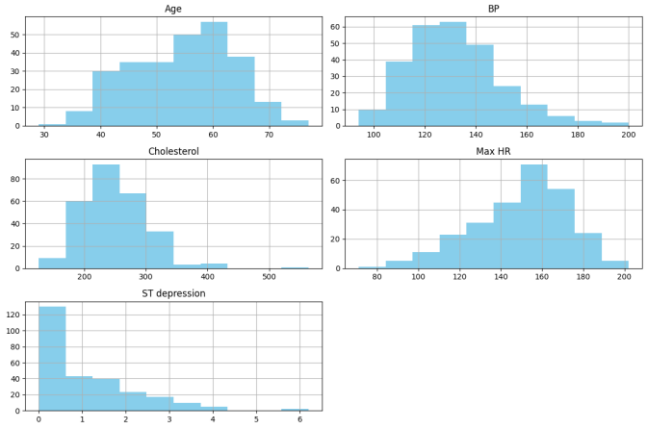

Fig. 2. Histogram visualization of numeric variables

As shown in Figure 1, histogram visualizations were utilized to examine the distribution of the numeric variables. The analysis indicated that several features, particularly cholesterol and blood pressure, exhibited a prominently right-skewed distribution. This skewness suggests the presence of extreme values and a long tail in the data. Identifying this distribution early is critical, as extreme values can negatively affect the classification process, especially for algorithms that are highly sensitive to distance metrics and spatial data distribution, such as K-Nearest Neighbors (KNN) and Support Vector Machines (SVM).

2) Outlier Detection and Handling

To further investigate the extreme values observed in the histograms, boxplot visualizations were generated (Figure 3). The boxplots clearly identified the presence of distinct outliers across various clinical features, most notably in cholesterol levels, blood pressure, and ST-segment depression. This specific finding supported the immediate need for robust outlier handling. Consequently, the Interquartile Range (IQR) method was applied to cap these outliers, as outlined in the preprocessing stage. After applying this capping process, the overall data distribution became demonstrably more stable and much better suited for distance-based modeling.

3) Feature Relationships and Correlation

Additionally, a correlation heatmap analysis was conducted to uncover the linear relationships between the different independent features and the target variable (heart disease). As depicted in Figure 4, the heatmap revealed several key relationships. Notably, 'chest pain type' and 'maximum heart rate' demonstrated a significantly stronger correlation with the classification target compared to other clinical features. These key insights suggest that classification algorithms can effectively utilize these specific patterns and strong relationships among medical features to improve predictive performance.

Summary Ultimately, the findings derived from these EDA visualizations emphasize that ensuring data quality and

applying effective preprocessing steps—such as handling skewed distributions and addressing outliers—are crucial, foundational factors that directly influence the reliability and performance of predictive models.

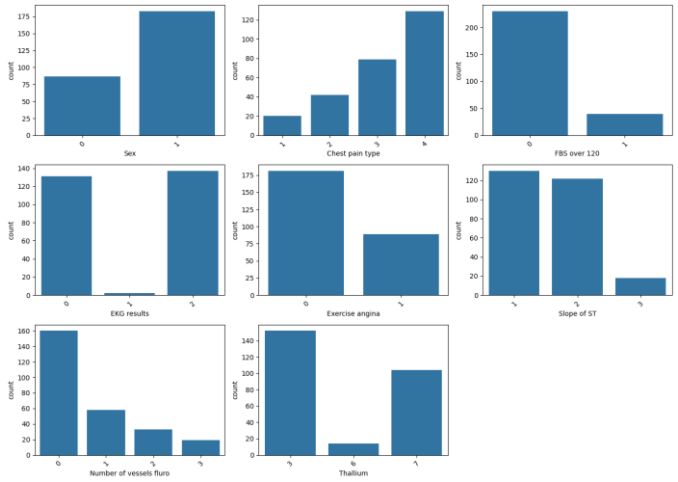

Fig. 3. Boxplot visualizations of clinical features

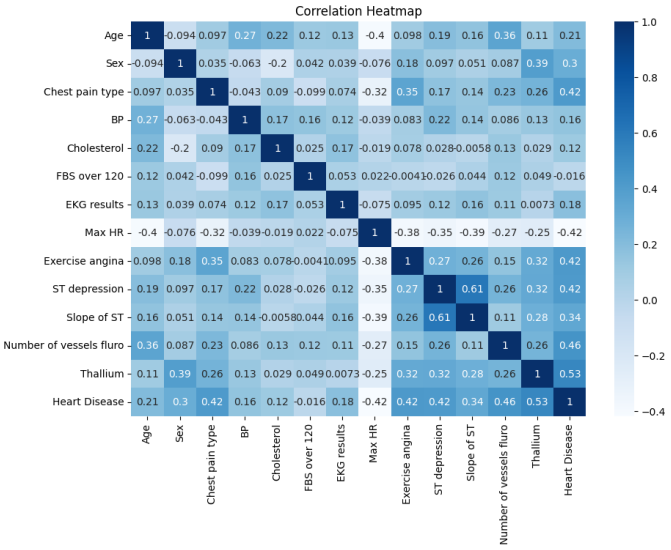

Fig. 4. Correlation heatmap of all variables

3.2 *Classification Model Test Results*

After preprocessing and splitting the data into 80% for training and 20% for testing, we assessed five classification algorithms and compared their performance based on accuracy, precision, recall, and F1-score metrics. The results are shown in Table 1.

The performance evaluation of five classification algorithms—Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbor (KNN), and Support Vector Machine (SVM)—was carried out using an 80:20 training and testing data split. As shown in Table 1, the results indicate that the KNN algorithm consistently outperforms the other models across all evaluation metrics. KNN achieved the highest performance with an accuracy rate of 93%, along with a precision of 0.93, a recall of 0.91, and an F1-score of 0.92. Logistic Regression followed closely, demonstrating strong performance with an accuracy of 91%. Random Forest and SVM exhibited similar competitive performance, each achieving an accuracy rate of 89%. In contrast, Decision Tree showed the least favorable performance, with an accuracy of only 74% and scores ranging from 0.73 to 0.74 on other metrics. The advantage of KNN in classifying heart disease risk suggests that the distribution of clinical features in the dataset can be effectively mapped using a distance-based pattern recognition approach [14]. The KNN model's high recall value of 0.91 is significant in medical diagnosis, as it highlights the model's ability to reduce the false negative rate. This ensures that patients with heart disease are not incorrectly identified as healthy [9]. Conversely, the Decision Tree algorithm often underperforms because single decision tree models can be sensitive to data variance and are prone to overfitting during the training phase. This can hinder their ability to generalize effectively to new data [19]. The results suggest that the selection of algorithm architecture plays a crucial role in prediction quality, with KNN identified as the most dependable model for the heart disease detection system evaluated in this study.

TABLE I. ALGORITHM PERFORMANCE COMPARISON RESULTS

Algorithm	Accuracy	Precision	Recall	F1-Score
Logistic Regression	91%	0.91	0.90	0.90
Decision Tree	74%	0.73	0.74	0.73
Random Forest	89%	0.90	0.87	0.88
K-Nearest Neighbor	93%	0.93	0.91	0.92
Support Vector Machine	89%	0.89	0.87	0.88

3.3 *Confusion Matrix Analysis*

1) *Logistic Regression*

Table 2 presents the confusion matrix results of the Logistic Regression model. The performance of the proposed classification algorithm was assessed using a confusion matrix with 54 test samples. The results showed a high level of prediction accuracy, identifying 31 instances as True Negative (TN) and 18 as True Positive (TP). The model's error rate was minimal, with only 2 cases of False Positive (FP) and 3 cases of False Negative (FN). Tharwat [20] highlighted that the confusion matrix is a vital metric for evaluating the reliability of machine learning models in detail. Overall, the aggregate accuracy reached 90.74%, indicating that the model demonstrates strong classification and class discrimination abilities.

TABLE II. CONFUSION MATRIX RESULTS OF THE LOGISTIC REGRESSION MODEL

Actual / Predicted	Negative	Positive
Negative	31	2
Positive	3	18

2) *Decision Tree*

The confusion matrix results for the Decision Tree classification model are summarized in Table 3.

TABLE III. CONFUSION MATRIX MODEL OF DECISION TREE

Actual / Predicted	Negative	Positive
Negative	24	9
Positive	5	16

According to the evaluation results in Table 3, the Decision Tree algorithm showed moderate classification performance, accurately predicting 24 True Negatives (TN) and 16 True Positives (TP) from a total of 54 test samples. However, it also resulted in 9 False Positives (FP) and 5 False Negatives (FN). The analysis yielded an accuracy of 74.07%, precision of 64.00%, recall of 76.19%, and an F1-score of 69.57%. The high recall indicates good sensitivity in identifying the positive class, while the lower precision suggests the model may be overly aggressive, leading to false positive predictions. Overall, the model's performance is deemed reliable for basic classification needs, though further optimization, such as hyperparameter tuning, is necessary to improve the misclassification rate in future research.

3) Random Forest

The confusion matrix results for the Random Forest classification model are summarized in Table 4.

TABLE IV. CONFUSION MATRIX MODEL RANDOM FOREST

Actual / Predicted	Negative	Positive
Negative	32	1
Positive	5	16

According to Table 4, the Random Forest algorithm's performance was thoroughly assessed using a confusion matrix based on 54 test samples. The results showed 32 True Negatives (TN), 1 False Positive (FP), 5 False Negatives (FN), and 16 True Positives (TP). This indicates strong classification capability, with an overall accuracy of 88.89%. The model excels in minimizing false positive predictions (Type I errors), as evidenced by its high precision (94.12%) and specificity (96.97%). Additionally, the F1-Score of 84.21% highlights the model's balanced performance between precision and sensitivity, affirming its reliability for practical use.

While the model demonstrates solid stability, there is potential for improvement in recall (sensitivity), which stood at 76.19% due to five misclassified positive cases. The trade-off between high precision and moderate recall is typical in ensemble modeling and is consistent with recent literature. For example, Pratama and Suryono [21] emphasized that recall efficiency in Random Forest can be significantly improved through hyperparameter tuning, such as optimizing the number of estimators (*n_estimators*) and tree depth. Similarly, Lu et al [22]. noted that when false negatives occur due to imbalanced data distribution, employing data balancing techniques like the Synthetic Minority Over-sampling Technique (SMOTE) is advisable. Therefore, if the goal is to reduce false negatives, advanced parameter calibration paired with data balancing strategies could be effective avenues for future model enhancement.

4) K-Nearest Neighbor

The confusion matrix results for the K-Nearest Neighbor (KNN) classification model are summarized in Table 5.

TABLE V. *CONFUSION MATRIX* MODEL KNN

Actual / Predicted	Negative	Positive
Negative	32	1
Positive	3	18

Based on the model evaluation results presented in Table V, the K-Nearest Neighbor (KNN) algorithm proved to be the best classification model in this study. Analysis of the confusion matrix from a total of 54 test data points shows that the model has excellent pattern recognition capabilities. The model successfully predicted the "Negative" class correctly (True Negative) for 32 data sets and the "Positive" class (True Positive) for 18 data sets. The resulting error rate is very minimal, where there was only 1 case of actual negative data being incorrectly predicted as positive (False Positive) and 3 cases of actual positive data being predicted as negative (False Negative).

Quantitatively, the performance metrics of the matrix show that the KNN model achieved an accuracy level of 92.59%. Furthermore, the model recorded a very high precision of 94.74%, indicating that the system is highly reliable in minimizing false positives (false alarms). The obtained sensitivity (recall) value of 85.71% and specificity of 96.97% prove that this algorithm is balanced in recognizing both classification classes. This harmonization between precision and recall is further validated by the F1-Score metric, which reached 90.00%, confirming the model's resilience to potential data imbalance.

The superior performance of KNN in this test is in line with its characteristics as a non-parametric, instance-based learning algorithm. The high levels of accuracy and specificity indicate that the distance metric used in the feature space has successfully mapped the decision boundary optimally between data classes. This is relevant to recent machine learning literature reviews [23], which state that the KNN algorithm tends to outperform traditional parametric models when faced with datasets with complex spatial feature distributions. With a misclassification ratio of only 7.41%, this KNN model is considered very feasible and representative for implementation in this case study.

5) Support Vector Machine

The confusion matrix results for the Support Vector Machine (SVM) classification model are summarized in Table 6.

TABLE VI. *CONFUSION MATRIX* MODEL SVM

Actual / Predicted	Negative	Positive
Negative	31	2
Positive	4	17

Based on the results of the classification test using the Support Vector Machine (SVM) algorithm, the model's performance was evaluated using a confusion matrix, as presented in Table VI. Of the 54 test samples, the SVM model correctly classified 31 instances as Negative (True Negative)

and 17 instances as Positive (True Positive). Conversely, 2 Negative instances were misclassified as Positive (False Positive), and 4 Positive instances were misclassified as Negative (False Negative). From this distribution, the SVM model achieved an overall accuracy of 88.89%. This high accuracy was further supported by a precision of 89.47% and a recall (sensitivity) of 80.95%. These evaluation results demonstrate that the SVM model possesses robust classification capabilities, with strong majority class recognition and a minimal false positive error rate.

This finding is consistent with recent literature, which highlights the effectiveness and stability of the SVM algorithm in binary classification tasks due to its ability to construct an optimal hyperplane that maximizes the margin between classes, thereby reducing the risk of overfitting on test data [24], [25].

3.4 Comparison of Errors Between Models

The comparison of classification errors among the evaluated models is summarized in Table VII.

TABLE VII. COMPARISON OF CLASSIFICATION ERRORS AMONG THE EVALUATED MODELS

Model	FP	FN	Total Error
Logistic Regression	2	3	5
Decision Tree	9	5	14
Random Forest	1	5	6
KNN	1	3	4
SVM	2	4	6

According to Table 7, the analysis of classification errors across models indicates that the K-Nearest Neighbors (KNN) algorithm achieved the best performance, with a total of 4 errors (1 False Positive and 3 False Negatives). KNN slightly surpassed Logistic Regression, which recorded 5 errors in total. In contrast, the Decision Tree model showed the least effective performance, with a total of 14 errors, primarily due to a high number of false positives (FP = 9).

This aligns with recent machine learning research highlighting that single Decision Tree algorithms are particularly susceptible to overfitting and sensitive to noise in training data. This vulnerability often leads to poor generalization on unseen test data, resulting in increased classification errors [26].

Moreover, when the Decision Tree method was improved using the Random Forest ensemble technique, the model successfully reduced the number of false positives to just one case. This illustrates that the voting mechanism in Random Forest effectively addresses the high variance commonly associated with a single Decision Tree. However, Random Forest still produced five false negatives, making its total error count (six cases) equivalent to that of the Support Vector Machine (SVM). Overall, the evaluation results indicate that, given the characteristics of the dataset, distance-based approaches such as K-Nearest Neighbors (KNN) or probabilistic linear models such as Logistic Regression exhibit greater robustness in balancing the trade-off between false positives and false negatives compared to other algorithms [24].

3.5 Discussion of Algorithm Comparison

Furthermore, when the Decision Tree approach was enhanced through the ensemble method of Random Forest, the model successfully reduced the number of false positives to only one case. This demonstrates that the voting mechanism in Random Forest is effective in mitigating the high variance typically associated with a single Decision Tree. Nevertheless, Random Forest still produced five false negatives, resulting in a total error count of six, which is equivalent to that of the Support Vector Machine (SVM). Overall, the evaluation results indicate that, given the characteristics of the dataset, distance-based approaches such as K-Nearest Neighbors (KNN) and probabilistic linear models such as Logistic Regression exhibit greater robustness in balancing the trade-off between false positives and false negatives compared to other algorithms.

IV. CONCLUSION

This study concludes that data quality and algorithm selection are essential for the performance of machine learning-based heart disease prediction systems. The experimental results showed that the K-Nearest Neighbor (KNN) algorithm achieved the highest classification accuracy at 93%, followed by Logistic Regression at 91%. In contrast, the Decision Tree recorded the lowest accuracy at 74% due to its tendency to overfit. The strong performance of KNN and Logistic Regression was significantly influenced by preprocessing steps, where outlier handling using the Interquartile Range (IQR) method and Min-Max normalization greatly improved model stability and accuracy, especially for distance-based algorithms like KNN and Support Vector Machine (SVM).

Additionally, analysis of the confusion matrix indicated that clinical misdiagnosis errors (false negatives) can be reduced through the combination of careful preprocessing and the selection of algorithms suited to the characteristics of medical data. Overall, this study provides a comparative evaluation of multiple algorithms and highlights that the integration of optimal data preprocessing with appropriate model selection is fundamental for developing accurate and reliable machine learning-based diagnostic systems in healthcare.

REFERENCES

[1] "World Health Organization, "Cardiovascular diseases... - Google Scholar." Accessed: May 29, 2026. [Online]. Available: https://scholar.google.co.id/scholar?hl=id&as_sdt=0%2C5&as_ylo=2020&as_yhi=2026&q=World+Health+Organization%2C+%E2%80%9Ccardiovascular+diseases+%28CVDs%29%2C%E2%80%9D+WHO%2C+Geneva%2C+Switzerland%2C+2023.&btnG=

[2] I. Akbar, F. Supriadi, and D. I. Junaedi, "Pemanfaatan Machine Learning Di Bidang Kesehatan," JATI (Jurnal Mahasiswa Teknik Informatika), vol. 9, no. 1, pp. 1744–1749, Jan. 2025, doi: 10.36040/JATI.V9I1.12663.

[3] R. Hidayat, Y. S. Sy, T. Sujana, M. Husnah, H. T. Saputra, and F. Okmayura, "Implementasi Machine Learning Untuk Prediksi Penyakit Jantung Menggunakan Algoritma Support Vector Machine," BIOS : Jurnal Teknologi Informasi dan Rekayasa Komputer, vol. 5, no. 2, pp. 161–168, Sep. 2024, doi: 10.37148/BIOS.V5I2.152.

[4] R. Muhammad, "Prediksi Penyakit Jantung dengan menggunakan Machine Learning Autogluon," 2024, Accessed: May 22, 2026. [Online]. Available: https://dspace.uii.ac.id/handle/123456789/48775

[5] A. P. Permana, A. P. Permana, K. Ainiyah, and K. F. H. Holle, "Analisis Perbandingan Algoritma Decision Tree, kNN, dan Naive Bayes untuk Prediksi Kesuksesan Start-up," JISKA (Jurnal Informatika Sunan Kalijaga), vol. 6, no. 3, pp. 178–188, Sep. 2021, doi: 10.14421/jiska.2021.6.3.178-188.

[6] A. Damayanti, … D. M.-P. J., and undefined 2026, "Analisis Klasifikasi Risiko Penyakit Jantung Menggunakan Metode Random Forest," ojs.stmik-banjarbaru.ac.id, doi: 10.35889/progresif.v22i2.3609.

[7] A. Ekong, "Anietie Ekong. Evaluation of Machine Learning Techniques Towards Early Detection of Cardiovascular Diseases," American Journal of Artificial Intelligence, vol. 7, no. 1, pp. 6–16, 2023, doi: 10.11648/j.ajai.20230701.12.

[8] D. K, S. Nandini, and S. V. R, "Comparative Study of Machine Learning Algorithms in Detecting Cardiovascular Diseases," May 2024, Accessed: May 22, 2026. [Online]. Available: https://arxiv.org/pdf/2405.17059

[9] A. Miftakhudin, N. A. Santoso, and B. A. Santoso, "Komparasi Algotirma KNN dan Random Forest untuk Diagnosa Penyakit Jantung Koroner (Studi Kasus: RSUD Dr. Soeselo Slawi)," RIGGS: Journal of Artificial Intelligence and Digital Business, vol. 4, no. 3, pp. 2962–2971, Aug. 2025, doi: 10.31004/RIGGS.V4I3.2424.

[10] H. Hikmayanti Handayani, S. Arum Puspita Lestari, Y. Cahyana, and P. Karawang, "Klasifikasi Risiko Penyakit Jantung Menggunakan Algoritma Vector Machine (SVM)," ejournal.itn.ac.id, vol. 9, no. 1, 2025, doi: 10.31603/komtika.v9i1.13450.

[11] M. Setiawan, M. Efendi, M. A.-… of E. (NOE), and undefined 2025, "Pengaruh Teknik Preprocessing terhadap Kinerja Model Explainable Boosting Machine (EBM) untuk Prediksi Serangan Jantung," ojs.unpkediri.ac.id, Accessed: May 22, 2026. [Online]. Available: https://ojs.unpkediri.ac.id/index.php/noe/article/view/25811

[12] Moch. A. Setiawan, Moh. H. Efendi, M. F. Akbar, and W. S. Pratama, "Pengaruh Teknik Preprocessing terhadap Kinerja Model Explainable Boosting Machine (EBM) untuk Prediksi Serangan Jantung," Nusantara of Engineering (NOE), vol. 8, no. 02, pp. 317–326, Oct. 2025, doi: 10.29407/noe.v8i02.25811.

[13] N. Nasution, F. B. Nasution, M. A. Hasan, and L. Kuning, "Predicting Heart Disease Using Machine Learning: An Evaluation of Logistic Regression, Random Forest, SVM, and KNN Models on the UCI Heart Disease Dataset," journal.uir.ac.id, vol. 9, no. 2, pp. 140–150, Apr. 2025, doi: 10.25299/ITJRD.2025.17941.

[14] W. Alsabhan, A. A.-S. Reports, and undefined 2025, "Effectiveness of machine learning models in diagnosis of heart disease: a comparative study," nature.comW Alsabhan, A AlfadhlyScientific Reports, 2025•nature.com, doi: 10.1038/s41598-025-09423-y.

[15] M. M. Ahsan and Z. Siddique, "Machine learning-based heart disease diagnosis: A systematic literature review," Artif. Intell. Med., vol. 128, p. 102289, Jun. 2022, doi: 10.1016/J.ARTMED.2022.102289.

[16] S. Uddin, A. Khan, M. E. Hossain, and M. A. Moni, "Comparing different supervised machine learning algorithms for disease prediction," SpringerS Uddin, A Khan, ME Hossain, MA MoniBMC medical informatics and decision making, 2019•Springer, vol. 19, no. 1, Dec. 2019, doi: 10.1186/S12911-019-1004-8.

[17] H. SENDY, "Evaluasi Kinerja Metode Support Vector Machine (Svm), Naive Bayes Dan Decision Tree Untuk Diagnosa Penyakit Jantung," 2023, Accessed: May 22, 2026. [Online]. Available: https://digilib.unila.ac.id/75571/

[18] A. Tharwat, "Classification assessment methods," Applied Computing and Informatics, vol. 17, no. 1, pp. 168–192, Jan. 2021, doi: 10.1016/J.ACI.2018.08.003.

[19] A. Pratama, S. Assegaff, J. Jasmir, and N. Nurhadi, "Optimizing Heart Disease Classification Using C4.5, Random Forest, and XGBoost with ANOVA, Chi-Square, and AdaBoost," Jurnal Teknik Informatika (Jutif), vol. 7, no. 2, pp. 1072–1090, Apr. 2026, doi: 10.52436/1.JUTIF.2026.7.2.5430.

[20] Y. Lu, "Joint SMOTE and Random Forest for Heart Disease Prediction and Characterization," 2024, doi: 10.5220/0012816100003885.

[21] N. Khateeb and M. Usman, "Efficient heart disease prediction system using K-nearest neighbor classification technique," ACM International Conference Proceeding Series, pp. 21–26, Dec. 2017, doi: 10.1145/3175684.3175703.

[22] M. Ahsan and Z. Siddique, "Machine Learning-Based Heart Disease Diagnosis: A Systematic Literature Review A Preprint," 2021. https://arxiv.org/pdf/2112.06459

[23] C. Cortes, V. Vapnik, and L. Saitta, "Support-vector networks," Machine Learning 1995 20:3, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/BF00994018.

[24] E. Halabaku, E. B.-I. A. & Soft, and undefined 2024, "Overfitting in Machine Learning: A Comparative Analysis of Decision Trees and Random Forests.," search.ebscohost.com, vol. 39, no. 6, pp. 987–1006, 2024, doi: 10.32604/IASC.2024.059429.

[25] M. Zlobin, V. Bazylevych, M. M. Злобін, and B. M. Базилевич, "A Data-Driven Approach for Balancing Overfitting and Underfitting in Decision Tree Models," Central Ukrainian Scientific Bulletin. Technical Sciences, vol. 1, no. 11(42), pp. 14–26, Mar. 2025, doi: 10.32515/2664-262X.2025.11(42).1.14-26.

[26] P. Probst, M. N. Wright, and A. L. Boulesteix, "Hyperparameters and tuning strategies for random forest," Wiley Online Library, vol. 9, no. 3, May 2019, doi: 10.1002/WIDM.1301.