Data Science : Journal Of Computing And Applied Informatics
Journal homepage: https://jocai.usu.ac.id

Optimizing K-Nearest Neighbor Using Ant Colony Optimization for Heart Disease Classification

Florentina Yuni Arini [*1], Patcharanikarn Pongthanoo [2], Kansa Maulina Salsabila [3], Muhammad Raihan [4]
[1,3,4] *Informatics Engineering, Universitas Negeri Semarang, Semarang, 50229, Indonesia*
[2] *Faculty of Management Science and Information Technology, Nakhon Phanom University, 48000, Thailand*
[*]Corresponding Author: floyuna@mail.unnes.ac.id

ARTICLE INFO
Article history:
Received 3 December 2025
Revised 16 January 2026
Accepted 23 January 2026
Available online 31 January 2026

E-ISSN: 2580-829X
P-ISSN: 2580-6769

How to cite:
F. Y. Arini, P. Pongthanoo, K. M. Salsabila, M. Raihan, and N. H. Muzakki, "Optimizing K-Nearest Neighbor Using Ant Colony Optimization for Heart Disease Classification," Data Science : Journal Of Computing And Applied Informatics, vol. V10, no. 1, Jan. 2026, doi: 10.32734/jocai.v10.i1-23647

http://doi.org/10.32734/jocai.v10.i1-23647

ABSTRACT
Heart disease is one of leading causes of death globally, making early detection essential for improving clinical outcomes. This study presents a heart disease prediction approach using the K-Nearest Neighbor (KNN) algorithm, addressing class imbalance with Synthetic Minority Over-sampling Technique (SMOTE) and enhancing feature selection through Ant Colony Optimization (ACO). Exploratory data analysis identified age, gender, cholesterol, blood pressure, exercise-induced angina, ST-segment depression, number of affected vessels, and thalassemia status as key indicators of disease severity. KNN model achieved 0.90 accuracy with balanced precision and recall. The employment of SMOTE improved sensitivity for the minority class, slightly reducing overall accuracy to 0.88. However, ACO as hyperparameter tuning KNN able to produce promising accuracy 0.91. This result indicate that combining KNN with metaheuristic optimization provides a reliable, interpretable method for heart disease prediction, offering valuable support for clinical decision-making and risk assessment. **Keyword:** Classification, K-Nearest Neighbor, SMOTE, Ant Colony Algorithm, Heart Disease

ABSTRAK
Penyakit jantung merupakan salah satu penyebab utama kematian secara global, sehingga deteksi dini penting untuk meningkatkan hasil klinis. Penelitian ini mengajukan pendekatan prediksi penyakit jantung menggunakan algoritma K-Nearest Neighbor (KNN), dengan menangani ketidakseimbangan kelas melalui Synthetic Minority Over-sampling Technique (SMOTE) dan meningkatkan pemilihan fitur menggunakan Ant Colony Optimization (ACO). Analisis data eksploratif mengidentifikasi usia, jenis kelamin, kadar kolesterol, tekanan darah, angina akibat olahraga, depresi segmen ST, jumlah pembuluh darah yang terdampak, dan status talasemia sebagai indikator utama tingkat keparahan penyakit. Model KNN mencapai akurasi 0,90 dengan presisi dan recall yang seimbang. Penerapan SMOTE meningkatkan sensitivitas pada kelas minoritas, meskipun akurasi keseluruhan sedikit menurun menjadi 0,88. Integrasi ACO lebih lanjut mengoptimalkan model, mencapai akurasi 0,91 dengan performa klasifikasi yang kuat. Hasil ini menunjukkan bahwa kombinasi KNN dengan optimisasi metaheuristik menyediakan metode prediksi penyakit jantung yang andal dan dapat diinterpretasikan, serta memberikan dukungan penting untuk pengambilan keputusan klinis dan penilaian risiko.
Keyword: Klasifikasi, K-Nearest Neighbor, SMOTE, Ant Colony Algorithm, Penyakit Jantung

1. Introduction

Over the past decade, the prevalence of heart disease, also known as heart disease, has increased significantly and has become one of the leading causes of death worldwide [1]. Heart disease caused an estimated 17.9 million deaths in 2019, representing 32% of global mortality, with 85% resulting from heart

attacks and strokes. Although many CVDs are preventable by mitigating risk factors such as smoking, poor diet, obesity, physical inactivity, alcohol consumption, and air pollution, these factors complicate rapid diagnosis. Consequently, early detection using artificial intelligence has emerged as a critical approach to improving clinical outcomes [2].

To assist in diagnosing this disease, the machine learning model K-Nearest Neighbor (KNN) algorithm will be applied as the main algorithm to create a classification model. KNN is the most widely used classification algorithm because it has the ability to overcome distance-based classification problems [3]. KNN algorithm works by grouping new data according to proximity to existing data, so this does not require many parameters. This algorithm has the advantage of overcoming data with irregular structures [4]. The accuracy of the KNN method is influenced by the selection of a parameter, especially the closest neighbor (k), for the classification process [5]. Since the KNN algorithm is a classic algorithm, there are several things that can affect the final accuracy results, including data balance.

To make up for these shortcomings, efforts are needed to handle data imbalance. The approach of Synthetic Minority Over-sampling Technique (SMOTE) is used to handle the data imbalance [6]. Data imbalance is a common occurrence in various data sets [7]. In general, this condition occurs due to the uneven distribution of classes in the data. In the dataset used, there are fewer samples of heart disease patients compared to healthy individuals, which causes the classification model to perform poorly in identifying minority groups. To address this, the Synthetic Minority Over-sampling Technique (SMOTE) is employed to generate additional samples for the minority class [8]. The use of SMOTE enhances the model's sensitivity to patterns in heart disease patients by creating new samples that closely resemble the existing minority data, thereby improving classification performance [9]. The SMOTE procedure is describe as follows. First, N, the desired amount of oversampling, must be set as an integer. This value can be chosen to balance the dataset, potentially achieving a 1:1 ratio between classes. Then, the following three main steps are applied iteratively: (1) select a sample from the minority class, (2) identify its K nearest neighbors (default K = 5), and (3) randomly select N neighbors for interpolation to generate new synthetic samples.

In general, this condition occurs due to the uneven distribution of classes in the data. In the dataset used, there is less data available in the case of heart disease patients than healthy patients, causing the classification model to be unable to accurately identify minority groups. By using the SMOTE Technique (Synthetic Minority Over-sampling Technique), so as to add data obtained from minority classes. The use of SMOTE increases the sensitivity to patterns in heart disease patients with a new sample that is very similar to the minority data, thus making the classification better. The procedure of SMOTE is described by. First, N, which is the desired amount of oversampling, must be set as an integer number. This number can be chosen within the dataset, balanced in a 1:1 ratio within different classes. Then, three main steps will be used iteratively. The first step is to select samples from the minority class, then samples from K (default 5) nearest neighbors are selected, and finally, N neighbors are randomly selected for interpolation and made into new samples [9].

Ant Colony algorithm (ACO) exhibit ants behavior in finding the path with the shortest distance to get to the food source, so this will find the optimal solution to the classification problem [10], [11]. ACO has the capability in finding global solutions adaptively and flexibly, showing robustness, using efficient probabilistic exploration, and can be easily combined with other algorithms [12], [13], [14]. Moreover, ACO is algorithm is useful for optimizing in selection of the most appropriate features for the classification process [15]. Therefore, in proposed, ACO algorithm is employed to improve the accuracy and efficiency of KNN model. With the proposed, features that have a major influence in predicting heart disease will be prioritized so that the results obtained from this classification process will be efficient.

2. Method

In this research, KNN, SMOTE, and ACO algorithms will be used to develop an accurate heart disease classification model. The stages in this research method include data collection, data pre-processing, SMOTE implementation, KNN implementation, optimization with ACO, and model performance evaluation.

2.1. Data Collection

The research data was taken from the Kaggle website with the title UCI Heart Disease Data https://www.kaggle.com/datasets/redwankarimsony/heart-disease-data/data is a dataset taken from the UCI repository [16], [17]. Datasets are downloaded and then used in this study, which contains 14 attributes and 303 instances. The Heart Disease dataset used is the Kaggle version of the classic UCI Heart Disease dataset, which aims to detect the presence of heart disease based on patients' clinical data.

The dataset consists of 303 patient samples with 14 main attributes, namely (1) **age**: patient's age in years; (2) **sex**: patient's gender (1 = male, 0 = female); (3) **cp**: type of chest pain (4 categories); (4) **trestbps**: resting blood pressure (mmHg); (5) **chol**: serum cholesterol level (mg/dl); (6) **fbs**: fasting blood sugar > 120 mg/dl (1

= yes, 0 = no); (7) **restecg**: resting electrocardiogram (ECG) results (3 categories); (8) **thalach**: maximum heart rate achieved; (9) **exang**: exercise-induced angina (1 = yes, 0 = no); (10) **oldpeak**: ST depression induced by exercise relative to rest; (11) **slope**: the slope of the peak exercise ST segment; (12) **ca**: number of major vessels colored by fluoroscopy (0–3); (13) **thal**: thalassemia (3 categories: normal, fixed defect, reversible defect); (14) **target**: target label, 1 indicates the presence of heart disease, 0 indicates absence. This dataset is widely used for heart disease prediction research due to its structured and comprehensive nature, encompassing clinical variables that are often key indicators of cardiovascular risk.

2.2. Data Preprocessing

Data preprocessing is an important step in machine learning-based research, especially for classifying heart disease. Medical datasets usually contain information that is incompatible, incomplete, and inconsistent to be used directly. Therefore, data preprocessing is required to improve data quality and to generate more accurate classification models. Several steps are taken at this stage to prepare heart disease dataset before it is applied for the next stage.

The first thing that will be done is data cleaning, where the dataset will be checked because there are often errors such as incomplete records (not zero) or errors during data collection [13]. The first step that must be done is to identify whether there is missing data in the dataset. After that, it will be followed by replacing these values with a certain strategy. For numerical features, the missing values will be replaced with mean and median values using imputation techniques. Meanwhile, for categorical features, the values are replaced with frequently occurring values. If the missing data presentation missing data is very large, the sample with many missing values can be deleted.

To overcome this, normalization will be performed to equalize the range of values among numerical features. With the min-max method, where each value in the dataset will be changed so that it is in the range of 0 to 1. The formula used to disguise normalization value is in equation (1).

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{1}$$

where X present as original value. Meanwhile X_{min} denote as minimum value and X_{max} denote as maximum value. X' exhibit as normalized value. This process ensures that all features are on the same scale so that the model can give a balanced weight to each feature. After all the data is ready to be used, the dataset will be divided into two parts, including training data to train a model and test data to evaluate the performance of the model on the data.

2.3. Synthetic Minority Over-sampling Technique (SMOTE)

In many medical classification cases, such as heart disease, the datasets used are often imbalanced. This means the number of majority or non-suffering classes is very large compared to the minority or suffering classes. In the heart disease classification process, data imbalance will be very dangerous because the model will fail to identify patients. Addressing this can be done by the Synthetic Minority Over-sampling Technique (SMOTE). A method to balance data that works by oversampling the minority class by creating new synthetic data. The application of the SMOTE technique [18], [19] is done by finding some of the closest neighbors of the same class. Once identified, a new sample will be generated which is the result of linear interpolation between two minority class samples, so that the variation in the minority data increases without introducing duplicate data. The formula for synthesized sample generation used is shown in equation (2).

$$X_{smote} = X_{minority} + \lambda \times \left(X_{nearest} - X_{minority}\right) \tag{2}$$

Based on the SMOTE formula X_{smote} in Equation (2), each component plays a role in creating new data points within the minority class. The term $X_{minority}$represents a chosen minority sample, while $X_{nearest}$is one of its nearest neighbors within the same class. The subtraction of $X_{nearest} - X_{minority}$ shows the move direction vector of the new point the original sample toward its neighbor. A random value $\boldsymbol{\lambda}$ present values between 0 and 1 to direct distance of the synthetic instance will be placed. By performing this process on every feature, SMOTE produces new samples that follow the original pattern and variation of the minority class, helping improve data balance without simply duplicating existing points.

2.4. K-Nearest Neighbor (KNN) Algorithm

After data is balanced, the KNN algorithm [20], [21] is applied for the classification process. The way it works is by calculating the distance between the new sample and the existing sample and then predicting the class according to the majority of the nearest neighbors or the k value. The first thing this KNN algorithm will do is

calculate the distance to be classified on the training data. After that this algorithm will select the k nearest neighbors based on the smallest distance. This k value is the most important parameter to determine how many neighbors should be considered in the classification process. To be able to measure the distance between two points of data, the Euclidean distance is used [22], [23]. The distance calculation formula can be used with the Euclidean formula in equation (3). And for the flow of the KNN algorithm in the flowchart is in Figure 1.

$$d(x,y) = \sqrt{\Sigma_{i=1}^{n}(x_i - y_i)^2} \quad (3)$$

The formula for equation (3) above can be seen that d(x,y) is the distance, x_i present as training data i, y_i denote as testing data i and n is a variable and data dimension. Calculation with this formula is used to know the value of the distance between 2 data, namely training data and data for testing. Then the data will be sorted from the smallest to the largest. Figure 1 presents the procedural steps KNN algorithm in performing classification. The process starts by entering the data sample that requires prediction. Then, an appropriate **k** value is selected to determine how many nearby data points will influence the decision. The algorithm measures the distance between the new data and all instances in the dataset. These distances are then ranked from the smallest to the largest. The closest **k** points are chosen as reference. The class of the new data is determined according to the majority class among those neighbors. The final stage displays the resulting classification output.

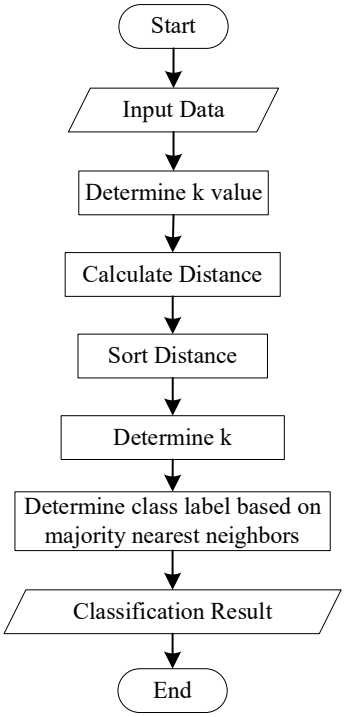

Figure 1. Flowchart KNN Algorithm

2.5. Ant Colony Optimization (ACO)

Ant Colony Optimization (ACO) [24], is a well-established metaheuristic optimization technique introduced by Marco Dorigo in the early 1990s [25], inspired by the cooperative foraging behavior of real ant colonies in identifying the shortest and most efficient paths to food sources. In nature, ants communicate indirectly through pheromone trails deposited along their routes. Paths with higher pheromone concentrations attract more ants, enabling the colony to collectively reinforce optimal routes while gradually abandoning less effective ones. This biological phenomenon forms the conceptual foundation of ACO, where artificial ants construct candidate solutions through probabilistic transitions that are guided by pheromone intensity and heuristic information. Throughout the iterative process, pheromone levels are updated by rewarding superior solutions while allowing inferior ones to evaporate. The global update rule of pheromone based on its level on each edge (i, j) define in equation (4).

$$Pheromone_{ij}^{new} = (1 - \rho)Pheromone_{ij}^{t} + Total_{ij}^{t} \quad (4)$$

where $Pheromone_{ij}^{new}$ as new position of pheromone, ρ present as the pheromone evaporation rate, $Pheromone_{ij}^{t}$ is the current pheromone value at iteration t and $Total_{ij}^{t}$ represents the total pheromone

deposited by all ants that selected edge (i, j) during that iteration. The evaporation process prevents unlimited pheromone accumulation, while the reinforcement term increases the attractiveness of edges that contribute to better-quality solutions. Through these updates, future ants are more likely to follow paths with higher desirability, guiding the search toward optimal routes while still allowing exploration.

Moreover, ACO is utilized as a hyperparameter optimization approach to control and determine the most effective k value in the KNN classifier for heart disease prediction. The optimization process begins with the initialization of a population of artificial ants that explore a predefined range of k values. Each ant selects a candidate k based on the pheromone distribution across the search space. Once a k value is chosen, the ant evaluates its performance by using KNN to classify heart disease data, and the resulting accuracy score is recorded. If a particular k value yields high accuracy, the pheromone intensity on that route is strengthened; meanwhile, pheromone on less successful values decreases through evaporation, preventing early stagnation in suboptimal solutions. This cycle is repeated for several iterations until the most optimal k value is consistently reinforced, indicating that it provides the highest classification performance. Finally, the optimized k parameter is applied to the KNN model to enhance reliability and accuracy in detecting heart disease.

Figure 2 shows a flowchart of the ACO algorithm. In this flow, it can be seen the stages in the classification process using ACO, such as parameter initialization, calculating suitability, and then updating the pheromone level based on the evaluation results. Then it will be known whether the iteration has reached the maximum value if it means that this process has been completed but if it has not reached it, the process will restart from the beginning until it reaches the maximum iteration.

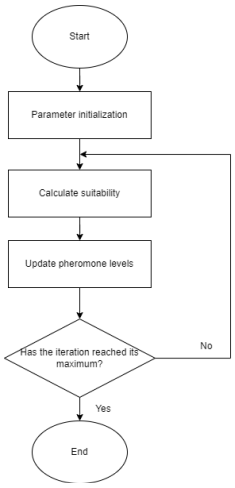

Figure 2. Ant Colony Optimization Flowchart

2.6. Performance Evaluation

The final step is the evaluation of the model using the test dataset. Model performance is assessed through standard metrics, including accuracy, precision, recall, F1-score, and the area under the ROC curve (AUC) [26], [27]. These metrics provide a comprehensive measure of the model's effectiveness and enable comparison with other relevant models. Accuracy, specifically, is defined as the proportion of correctly predicted instances—both true positives and true negatives—relative to the total number of observations in the dataset. The accuracy [28], [29] can be calculated using the formula presented in Equation (5).

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \tag{5}$$

where TP (True Positive) represents instances where the actual condition is positive, and the model correctly identifies them as positive. TN (True Negative) refers to cases where the actual condition is negative, and the model accurately identifies them as negative. FP (False Positive) occurs when the actual condition is negative, but the model incorrectly predicts it as positive. FN (False Negative) represents instances where the actual condition is positive, but the model fails to identify them, incorrectly classifying them as negative.

$$Precision = \frac{TP}{TP+FP} \tag{6}$$

Additionally, while accuracy measures the overall proportion of correct predictions, including both positive and negative instances, precision focuses specifically on positive predictions. It quantifies the proportion of true positive cases among all instances predicted as positive. Precision is therefore essential to complement accuracy, especially in imbalanced datasets or in situations where false positive errors have a high impact, as exhibit in Equation (6).

3. Result and discussion

In Results and Discussion section, the Heart Disease dataset is analyzed to extract meaningful clinical insights. Three experimental conditions are examined: (a) KNN as a baseline classifier; (b) KNN enhanced with SMOTE to address class imbalance; and (c) KNN optimized using Ant Colony Optimization (ACO). The performance of each approach is evaluated using standard classification metrics to determine the most effective model.

3.1. Insights Heart Disease Dataset

Based on exploratory analysis, the insight of Heart Disease Dataset, tren patterns can be observed as follows: (1) **Age and Heart Disease Risk**: As patient age increases, the likelihood of having heart disease also rises, indicating a positive relationship between age and heart disease risk; (2) **Gender**: Males show a higher prevalence of heart disease compared to females, consistent with clinical findings that gender influences cardiovascular risk; (3) **Cholesterol and Blood Pressure**: Patients with high serum cholesterol or elevated resting blood pressure are more likely to develop heart disease; (4) **Maximum Heart Rate (thalach) and Exercise-Induced Angina (exang)**: Patients with heart disease typically achieve a lower maximum heart rate during exercise and more frequently experience angina compared to healthy patients; (5) **ST Depression (oldpeak) and ST Slope**: ST depression during exercise and the slope of the ST segment show different patterns between patients with and without heart disease based on **electrocardiogram (ECG/EKG) result [30]**; (6) **Number of Vessels and Thalassemia (ca and thal)**: The number of affected vessels and thalassemia condition are related to disease severity and serve as important indicators in clinical diagnosis. These trends indicate that, although the dataset is relatively simple, the clinical attributes are sufficiently informative to distinguish patients with and without heart disease.

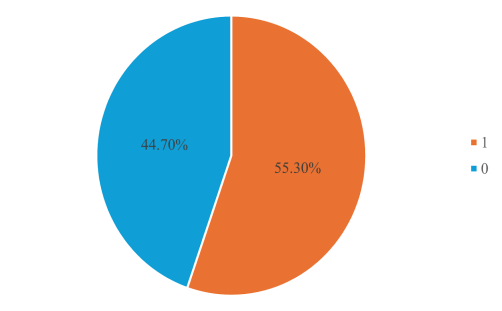

Figure 3. Prosentage Heart Disease Datases (1:heart disease; 0:no heart disease)

In the Heart Disease dataset (Kaggle version by redwankarimsony), out of 303 patients, 165 (54.5%) have heart disease (target = 1), while 138 (45.5%) do not have heart disease (target = 0), as shown in Figure 3. This indicates a slightly higher prevalence of heart disease in the dataset population. Among the 165 patients with heart disease, ages ranged from 35 to 77 years, with a mean of 56.7 years, and males accounted for 63.6%. Chest pain types varied, with 24.2% presenting typical angina and 30.3% presenting atypical angina. The mean resting blood pressure was 138.4 mmHg, and mean cholesterol level was 253.1 mg/dL. Nearly half of the patients experienced exercise-induced angina, and the maximum heart rate averaged 143.6 bpm. ST depression, ST slope, the number of affected vessels, and thalassemia type also varied, providing key clinical indicators. These statistics highlight patterns in demographics, cardiovascular function, and risk factors, offering a strong foundation for heart disease prediction and analysis.

3.2. Performance KNN

The performance of the KNN model was analyzed using a confusion matrix. As shown in Figure 4, the matrix includes two classes: 0 (not suffering from heart disease) and 1 (suffering from heart disease). True Positive (TP) refers to the number of observations that are actually positive (class 1) and correctly predicted as positive (class 1).

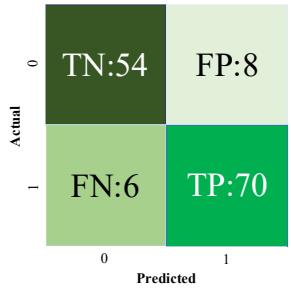

Figure 4. KNN Confusion matrix

The confusion matrix for the KNN model (Figure 4) shows a True Positive (TP) value of 70, indicating that the model correctly classified 70 positive instances. True Negative (TN) represents the number of observations that are actually negative (class 0) and correctly predicted as negative; TN is 54, meaning that 54 negative cases were correctly classified. False Positive (FP) refers to instances that are actually negative (class 0) but incorrectly predicted as positive (class 1). In this matrix, FP is 8, indicating that 8 negative cases were misclassified as positive (false alarms). False Negative (FN) represents instances that are actually positive (class 1) but incorrectly predicted as negative (class 0). FN is 6, meaning that 6 positive cases were misclassified.

Table 1. KNN Result Experiment

	Precision	**Recall**	**F1-score**	**Support**
Normal (0)	0,90	0,87	0,89	62
Heart Disease (1)	0,90	0,92	0,91	76
Accuracy	-	-	0,90	138
macro avg	0,90	0,90	0,90	138
weighted avg	0,90	0,90	0,90	138

Table 1 presents the detailed performance KNN model based on accuracy, presision, recall, F1-score, support (number of actual occurrences of the class in the dataset) [31]. Precision measures how often a positive prediction is correct. For class 0, the precision of 0.90 indicates that 90% of the data predicted as class 0 were correctly classified as class 0 (normal). Similarly, for class 1, a precision of 0.90 means that 90% of the data predicted as class 1 were correctly classified. Recall measures the proportion of actual positive instances that were correctly identified. For class 0, the recall of 0.87 shows that 87% of all actual class 0 data were correctly predicted. For class 1, the recall of 0.92 indicates that 92% of all actual class 1 data were correctly classified. The F1-score provides a balance between precision and recall. The F1-score of 0.89 for class 0 reflects a good balance, while the F1-score of 0.91 for class 1 also indicates a strong balance. The table also shows the number of samples in each class: 62 for class 0 and 76 for class 1. Overall, the model correctly classified 90% of the data, resulting in an accuracy of 0.90.

3.3. Performance KNN with SMOTE

The confusion matrix in Figure 5 shows the performance of the KNN algorithm after applying SMOTE to handle class imbalance. It displays the number of correctly and incorrectly classified instances for each class, offering insight into the model's accuracy, sensitivity, and specificity. The results demonstrate that the model achieves better classification, particularly for the minority class.

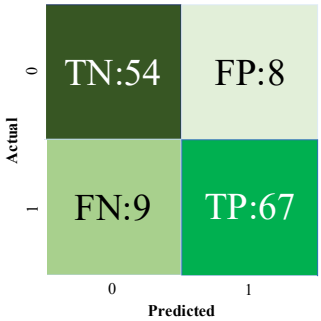

Figure 5. KNN with SMOTE Confusion matrix

Figure 5 shows the confusion matrix of KNN and SMOTE. The matrix consists of two classes: class 0 (not suffering from heart disease) and class 1 (suffering from heart disease). True Positive (TP) refers to the number of observations that are actually positive (class 1) and correctly predicted as positive. In this figure, TP is 67, indicating that the model correctly classified 67 positive cases. True Negative (TN) represents the number of observations that are actually negative (class 0) and correctly predicted as negative. TN is 54, meaning that 54 negative cases were correctly classified. False Positive (FP) refers to observations that are actually negative (class 0) but incorrectly predicted as positive (class 1). In this matrix, FP is 8, indicating that 8 negative cases were misclassified as positive (false alarms). False Negative (FN) refers to observations that are actually positive (class 1) but incorrectly predicted as negative (class 0). FN is 9, meaning that 9 positive cases were misclassified as negative (misclassifications).

Table 2. KNN with SMOTE Result Experiment

	Precision	**Recall**	**F1-score**	**Support**
Normal (0)	0,86	0,87	0,86	62
Heart Disease (1)	0,89	0,88	0,89	76
Accuracy	-	-	0,88	138
macro avg	0,88	0,88	0,88	138
weighted avg	0,88	0,88	0,88	138

Table 2 presents the performance of the KNN and SMOTE combination model. The precision for class 0 is 0.86, indicating that 86% of the data predicted as class 0 were correctly classified as class 0 (normal). For class 1, the precision is 0.89, meaning that 89% of the data predicted as class 1 were correctly classified. Recall measures the proportion of actual positive instances that were correctly identified. The recall for class 0 is 0.87, showing that 87% of all actual class 0 data were correctly predicted. For class 1, the recall is 0.88, indicating that 88% of all actual class 1 data were correctly identified. The F1-score provides a balance between precision and recall. For class 0, the F1-score is 0.86, reflecting a good balance between precision and recall. For class 1, the F1-score is 0.89, also showing a good balance. The table also shows the number of samples in each class. Class 0 has 62 samples, while class 1 has 76 samples. Overall, the model correctly classified 88% of the data, resulting in an accuracy of 0.88.

3.4. Performance KNN with ACO

The confusion matrix for KNN combined with ACO is shown in Figure 6. This figure presents two classes: class 0 (not suffering from heart disease) and class 1 (suffering from heart disease). True Positive (TP) represents the number of observations that are actually positive (class 1) and correctly predicted as positive. In this figure, TP is 117, indicating that the model correctly classified 117 positive cases. True Negative (TN) represents the number of observations that are actually negative (class 0) and correctly predicted as negative. Here, TN is 75, meaning that 75 negative cases were correctly classified. False Positive (FP) refers to observations that are actually negative (class 0) but incorrectly predicted as positive (class 1). In this matrix, FP is 37, indicating that 37 negative cases were misclassified as positive (false alarms). False Negative (FN) refers to observations that are actually positive (class 1) but incorrectly predicted as negative (class 0). FN is 47, meaning that 47 positive cases were misclassified as negative (misclassifications).

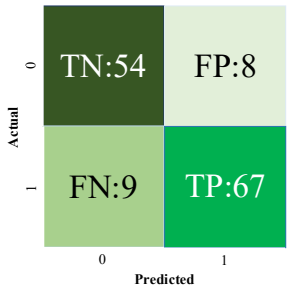

Figure 5. KNN with ACO Confusion matrix

Table 3 presents the performance of the KNN algorithm combined with ACO. The precision value of 0.87 for class 0 indicates that 87% of the data predicted as class 0 were correctly classified as class 0 (normal). For class 1, a precision of 0.93 means that 93% of the data predicted as class 1 were correctly classified. Recall measures the proportion of actual positive instances that were correctly identified. A recall of 0.90 for class 0

shows that 90% of all actual class 0 data were correctly predicted, while a recall of 0.91 for class 1 indicates that 91% of all actual class 1 data were correctly identified. The F1-score provides a balanced view of precision and recall.

Table 3. KNN with ACO Result Experiment

	Precision	**Recall**	**F1-score**	**Support**
Normal (0)	0,86	0,87	0,86	62
Heart Disease (1)	0,89	0,88	0,89	76
Accuracy	-	-	0,88	138
macro avg	0,88	0,88	0,88	138
weighted avg	0,88	0,88	0,88	138

For class 0, the F1-score of 0.89 reflects a good balance between precision and recall. Similarly, the F1-score of 0.92 for class 1 demonstrates a strong balance between precision and recall. The table also shows the actual number of samples in each class. Class 0 has 112 samples, while class 1 has 164 samples. Overall, the model successfully classified 91% of the data correctly, resulting in an accuracy of 0.91.

4. Conclusion

Based on the analysis of the Heart Disease dataset, several clinical and computational insights were obtained. Exploratory data analysis revealed key trends: the risk of heart disease increases with age, males have a higher prevalence than females, and elevated cholesterol and blood pressure are associated with higher disease likelihood. Additionally, lower maximum heart rate during exercise, the presence of exercise-induced angina, ST depression patterns, the number of affected vessels, and thalassemia status were identified as significant indicators for disease severity. These findings underscore that the dataset, though relatively simple, contains clinically relevant attributes capable of distinguishing patients with and without heart disease.

Performance evaluation of the KNN model demonstrated strong predictive capability, achieving an overall accuracy of 0.90. The model balanced precision and recall across both classes, with F1-scores of 0.89 for normal patients and 0.91 for heart disease patients, indicating reliable classification. When KNN was enhanced with SMOTE to address class imbalance, the model's performance slightly decreased in overall accuracy to 0.88, though classification of the minority class improved, highlighting the trade-off between sensitivity to imbalanced classes and overall accuracy.

Further optimization using Ant Colony Optimization (ACO) improved KNN performance, achieving an overall accuracy of 0.91. KNN and ACO demonstrated balanced precision, recall, and F1-scores for both classes, reflecting more robust classification performance. The increase in correctly classified instances suggests that ACO effectively optimized the model parameters, enhancing predictive performance without sacrificing interpretability.

In summary, the combination of KNN with ACO provides the most effective approach for heart disease prediction in this dataset. Clinical patterns identified support existing medical knowledge, and the computational results indicate that metaheuristic optimization can enhance standard machine learning models, offering a promising methodology for heart risk assessment.

References

[1] T. A. Gaziano, "Cardiovascular Diseases Worldwide," *Public Heal. Approach to Cardiovasc. Dis. Prev. Manag.*, pp. 8–18, 2022, doi: 10.1201/b23266-2.

[2] N. E. Almansouri *et al.*, "Early Diagnosis of Cardiovascular Diseases in the Era of Artificial Intelligence: An In-Depth Review," *Cureus*, 2024, doi: 10.7759/cureus.55869.

[3] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "KNN model-based approach in classification," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 2888, pp. 986–996, 2003, doi: 10.1007/978-3-540-39964-3_62.

[4] A. A. Amer, S. D. Ravana, and R. A. A. Habeeb, "Effective k-nearest neighbor models for data classification enhancement," *J. Big Data*, vol. 12, no. 1, 2025, doi: 10.1186/s40537-025-01137-2.

[5] R. K. Halder, M. N. Uddin, M. A. Uddin, S. Aryal, and A. Khraisat, "Enhancing K-nearest neighbor algorithm: a comprehensive review and performance analysis of modifications," *J. Big Data*, vol. 11, no. 1, 2024, doi: 10.1186/s40537-024-00973-y.

[6] D. Elreedy, A. F. Atiya, and F. Kamalov, "A theoretical distribution analysis of synthetic minority oversampling technique (SMOTE) for imbalanced learning," *Mach. Learn.*, vol. 113, no. 7, pp. 4903–4923, 2024, doi: 10.1007/s10994-022-06296-4.

[7] M. Altalhan, A. Algarni, and M. Turki-Hadj Alouane, "Imbalanced Data Problem in Machine Learning: A Review," *IEEE Access*, vol. 13, pp. 13686–13699, 2025, doi: 10.1109/ACCESS.2025.3531662.
[8] M. Mujahid *et al.*, "Data oversampling and imbalanced datasets: an investigation of performance for machine learning and feature engineering," *J. Big Data*, vol. 11, no. 1, 2024, doi: 10.1186/s40537-024-00943-4.
[9] A. Ishaq *et al.*, "Improving the Prediction of Heart Failure Patients' Survival Using SMOTE and Effective Data Mining Techniques," *IEEE Access*, vol. 9, pp. 39707–39716, 2021, doi: 10.1109/ACCESS.2021.3064084.
[10] M. Dorigo and C. Blum, "Ant colony optimization theory: A survey," *Theor. Comput. Sci.*, vol. 344, no. 2–3, pp. 243–278, 2005, doi: 10.1016/j.tcs.2005.05.020.
[11] M. Dorigo and T. Stützle, "The Ant Colony Optimization Metaheuristic," *Ant Colony Optim.*, pp. 25–64, 2018, doi: 10.7551/mitpress/1290.003.0004.
[12] Y. Sun, S. Wang, Y. Shen, X. Li, A. T. Ernst, and M. Kirley, "Boosting ant colony optimization via solution prediction and machine learning," *Comput. Oper. Res.*, vol. 143, 2022, doi: 10.1016/j.cor.2022.105769.
[13] A. Tsagaris, P. Kyratsis, and G. Mansour, "The Integration of Genetic and Ant Colony Algorithm in a Hybrid Approach," *Int. J. Intell. Syst. Appl. Eng.*, vol. 11, no. 2, pp. 336–342, 2023.
[14] J. A. Widians, R. Wardoyo, and S. Hartati, "A Hybrid Ant Colony and Grey Wolf Optimization Algorithm for Exploitation-Exploration Balance," *Emerg. Sci. J.*, vol. 8, no. 4, pp. 1642–1654, 2024, doi: 10.28991/ESJ-2024-08-04-023.
[15] B. Chen, L. Chen, and Y. Chen, "Efficient ant colony optimization for image feature selection," *Signal Processing*, vol. 93, no. 6, pp. 1566–1576, 2013, doi: 10.1016/j.sigpro.2012.10.022.
[16] S. Arooj, S. ur Rehman, A. Imran, A. Almuhaimeed, A. K. Alzahrani, and A. Alzahrani, "A Deep Convolutional Neural Network for the Early Detection of Heart Disease," *Biomedicines*, vol. 10, no. 11, 2022, doi: 10.3390/biomedicines10112796.
[17] N. M. AbdelAziz, G. A. Fouad, S. Al-Saeed, and A. M. Fawzy, "Deep Q-Network (DQN) Model for Disease Prediction Using Electronic Health Records (EHRs)," *Sci*, vol. 7, no. 1, 2025, doi: 10.3390/sci7010014.
[18] L. Hall, W. P. Kegelmeyer, N. Chawla, and K. Bowyer, "SMOTE: Synthetic Minority Over-sampling Technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, 2002, [Online]. Available: https://arxiv.org/pdf/1106.1813.pdf%0Ahttp://www.snopes.com/horrors/insects/telamonia.asp.
[19] D. Elreedy and A. F. Atiya, "A Comprehensive Analysis of Synthetic Minority Oversampling Technique (SMOTE) for handling class imbalance," *Inf. Sci. (Ny).*, vol. 505, pp. 32–64, 2019, doi: 10.1016/j.ins.2019.07.070.
[20] M. Cover T and E. Hart P, "Nearest Neighbor Pattern Classification," *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, 1967.
[21] A. Kataria and M. D. Singh, "A Review of Data Classification Using K-Nearest Neighbour Algorithm ," *Int. J. Emerg. Technol. Adv. Eng.* , vol. 3, no. 6, 2013.
[22] I. Arora, N. Khanduja, and M. Bansal, "Effect of Distance Metric and Feature Scaling on KNN Algorithm while Classifying X-rays," *CEUR Workshop Proc.*, vol. 3176, pp. 61–75, 2022.
[23] R. Mussabayev, "Optimizing Euclidean Distance Computation," *Mathematics*, vol. 12, no. 23, 2024, doi: 10.3390/math12233787.
[24] M. Dorigo, M. Birattari, and T. Stützle, "Ant colony optimization artificial ants as a computational intelligence technique," *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28–39, 2006, doi: 10.1109/CI-M.2006.248054.
[25] M. Dorigo and D. C. Gianni, "Ant Colony Optimization: A New Meta-Heuristic," *Proc. 1999 Congr. Evol. Comput. (Cat. No. 99TH8406)*, pp. 1470–1477, 1992.
[26] D. Tests, "Basic Principles of ROC Analysis," *Semin. Nucl. Med.*, vol. VIII, no. 4, pp. 283–298, 1978.
[27] A. M. Carrington *et al.*, "Deep ROC Analysis and AUC as Balanced Average Accuracy, for Improved Classifier Selection, Audit and Explanation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 329–341, 2023, doi: 10.1109/TPAMI.2022.3145392.
[28] S. Tharwat, "'Classification assessment methods,'" *Appl. Comput. Informatics,* vol. 17, n, pp. 168–192, 2021.
[29] O. Rainio, J. Teuho, and R. Klén, "Evaluation metrics and statistical tests for machine learning," *Sci. Rep.*, vol. 14, no. 1, 2024, doi: 10.1038/s41598-024-56706-x.

[30] E. Diderholm, B. Andren, and G. Frostfeldt, “ST depression in ECG at entry indicates severe coronary lesions and large benefits of an early invasive treatment strategy in unstable coronary artery disease,” *ACC Curr. J. Rev.*, vol. 11, no. 3, p. 12, 2002, doi: 10.1016/s1062-1458(02)00620-7.
[31] D. M. W. Powers, “Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation,” 2020, [Online]. Available: http://arxiv.org/abs/2010.16061.