J-INTECH (Journal of Information and Technology)
Accredited Sinta 4 Ministry of Higher Education, Science and Technology Republic of Indonesia SK No. 10/C/C3/DT.05.00/2025
E-ISSN: 2580-720X || P-ISSN: 2303-1425

Hybrid PSO Feature Selection Correlation and Support Vector Machine Model for Heart Disease Detection

Sarina Safitri[1], Taghfirul Azhima Yoga Siswa[2*], Wawan Joko Pranoto[3]

[1,2,3]*Department of Computer Science, Muhammadiyah University of East Kalimantan, Jl. Ir. H. Juanda, Samarinda City, Indonesia*

Keywords
Correlation, Heart, Machine Learning, Particle Swarm Optimizatio, Support Vector Machine

Corresponding Author:
tay758@umkt.ac.id

Abstract
Heart disease remains a major health problem worldwide. The World Health Organization (WHO) reports that in 2022, approximately 19.8 million people died from heart disease, highlighting the need for the implementation of an appropriate early detection model. This study proposes a hybrid SVM–PSO model with correlation-based feature selection, duplicate data handling, and a multi-metric fitness function to enhance classification performance. PSO is employed to optimize the C parameter and RBF kernel of SVM, producing a more robust and balanced model compared to existing approaches**.** This study uses a heart disease dataset consisting of 1,025 rows with 13 attributes and 1 target variable obtained from the Kaggle repository and republished on the Zenodo platform in 2024. The research stages include Pre-Processing, Standardization, Feature Selection based on Correlation, and evaluation using the 10-Fold Cross Validation technique with Accuracy, precision, recall, and F1-score metrics. The results show that Support Vector Machine (SVM) achieved an Accuracy of 82.80%, Precision of 79.31%, Recall of 91.70%, and an F1-score of 84.88%. After optimization using PSO, the performance improved to an accuracy of 84.46%, precision of 80.54%, recall of 92.72%, and an F1-score of 86.04%. The experimental results indicate performance improvements of 2.00% in accuracy, 1.55% in precision, 1.11% in recall, and 1.37% in F1-score after PSO optimization. These results prove that the applied hybrid approach successfully improved the ability to detect heart disease. Therefore, this study contributes by demonstrating that PSO-based hyperparameter optimization can effectively enhance SVM classification performance for heart disease detection. The proposed model also has practical implications as a decision support tool for early heart disease detection that can assist medical practitioners in improving diagnostic accuracy and supporting preventive treatment strategies.

1. Introduction

Heart disease is a condition in which the heart's function is impaired, resulting in abnormal blood flow. This is one of the leading causes of death worldwide. According to the World Health Organization (WHO) and the Department of Health, approximately 19.8 million people died from heart disease in 2022, representing about 32% of all global deaths, with 85% of these deaths caused by heart attacks and strokes [1]. However, many people are still unaware of heart disease risks due to low awareness of regular medical check-ups, which is further worsened by unhealthy lifestyles such as smoking, consuming high-fat and high-salt fast food, and lack of physical activity that increase the risk of heart disease [2].

Heart health can be maintained by early detection of symptoms and risk factors so that medical action can be taken more quickly to reduce mortality rates while improving patients' quality of life with more appropriate and targeted treatment [3]. Currently, treatment can be carried out through a diagnostic process, one of which is through the use of artificial intelligence such as machine learning to improve accuracy and reduce human error in understanding medical data. [4].

Machine learning is a branch of artificial intelligence that enables systems to learn automatically from data and experience, thereby improving their capabilities over time without requiring direct instruction, including determining classifications or making predictions [5]. Classification is a machine learning technique used to divide data into specific categories based on its features or characteristics [6]. Machine learning contributes greatly, especially in the field of health, particularly for early diagnosis and prediction of many diseases such as diabetes, hypertension, heart disease, lung cancer, and kidney failure through the application of classification algorithms such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Random Forest, Logistic Regression, Neural Network, Naive Bayes, and Decision Tree [7].

The application of classification in case studies of heart disease yielded accuracy results of 98.54% for Decision Tree, 85.01% for Naive Bayes, and 81.83% for Neural Network [8]. Another study found accuracy results of 94% for Random Forest, 84.15% for K-Nearest Neighbors (KNN), 83.81% for Logistic Regression, and 82.49% for Support Vector Machine (SVM) [9]. Then, the research used the Support Vector Machine (SVM) algorithm with an accuracy of 86.92% [10]. However, in other studies, there was a decrease in accuracy with the Naive Bayes algorithm at 71%, Decision Tree at 72%, and Random Forest at 75% [11]. Meanwhile, in a study on breast cancer classification, there was a decrease in the accuracy value of the Support Vector Machines (SVM) algorithm to 65.22% [12].

A more significant decrease related to heart disease was found by taking the original data from the Cleveland database of the UCI Machine Learning Repository, which consisted of 303 data rows with a K-Nearest Neighbors (KNN) algorithm accuracy value of 64.03% [13]. Meanwhile, another study that took the dataset source from four leading medical institutions, namely Cleveland, Hungary, Switzerland, and Long Beach V with a total of 1,025 data rows, obtained an increase in the accuracy value of the Support Vector Machine (SVM) algorithm to 85% [14]. Thus, combining datasets from various sources has been proven to increase the accuracy of the Support Vector Machine (SVM) algorithm to 98–100% after preprocessing and feature optimization, making the model more stable and effective in detecting heart disease [15].

Support Vector Machines (SVM) is a Machine Learning algorithm that works on the principle of Structural Risk Minimization (SRM) with the aim of finding the best Hyperplane (separator) that separates two Classes in the input space [16]. The SVM algorithm is considered to still be superior in predicting and classifying heart disease data, showing an accuracy value of 89% compared to the Decision Tree algorithm, which is only 77% [17]. Several optimization approaches that can be used to improve accuracy include the Whale Optimization Algorithm (WOA), Bat Algorithm (BA), Firefly Algorithm (FA), Ant Colony Optimization (ACO), Genetic Algorithm (GA), Dragonfly Algorithm (DA), Grey Wolf Optimizer (GWO), Cuckoo Search (CS), Artificial Bee Colony (ABC), and Particle Swarm Optimization (PSO) [18].

Particle Swarm Optimization (PSO) is an optimization method inspired by the movements and behavior of animals such as fish and birds when searching for food or prey [19]. Several case studies on heart disease show that PSO can improve the accuracy of machine learning algorithms, where the accuracy of Random Forest increased from 91.4% using Genetic Algorithm (GA) optimization to 95.6% after applying PSO [20]. Meanwhile,

Multi-Layer Perceptron (MLP) and PSO optimization produced an accuracy of 84.6%, higher than Multi-Layer Perceptron (MLP) and Backpropagation (BP), which was only 80.2% [21].

Previous SVM+PSO studies mainly emphasize parameter optimization while overlooking feature redundancy and data duplication, which may introduce noise, increase model complexity, and lead to overfitting [22]. To address these limitations, this study integrates correlation-based feature selection to eliminate redundant features and duplicate data removal to improve data quality, combined with PSO optimization to enhance SVM performance. Model robustness is further evaluated using 10-fold cross-validation.

2. Research Method

This research procedure consists of problem identification and data collection followed by data pre-processing, including data cleaning and standardization. Next, feature selection is performed using correlation analysis to select the most relevant features for heart disease classification. The processed data is then divided using the K-10 Fold Cross Validation technique into training and testing data. Furthermore, the classification process is carried out using Support Vector Machine (SVM) and Support Vector Machine (SVM) optimized using Particle Swarm Optimization (PSO). Once the model is formed, the results are evaluated using a confusion matrix and performance metrics such as accuracy, precision, recall, and F1-score to assess the model's performance.

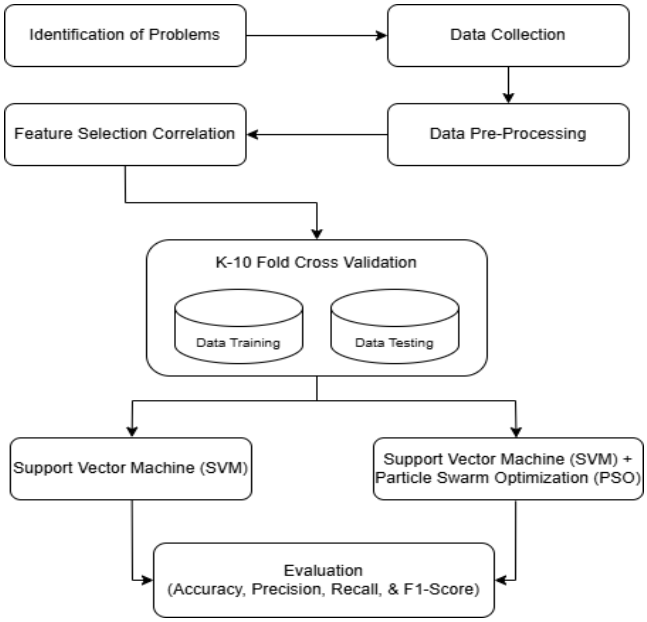

Figure 1. Research Flow

2.1 Data Collection

Table 1. Data Attributes

Attributes	Data Type	Description
Age	Numeric	Age (in years)
Sex	Binary	Gender (1 = male, 0 = female)
CP	Categorical	Type of chest pain (4 types) (0 = due to reduced blood supply to the heart, 1 = not related to the heart, 2 = esophageal spasm, 3 = no signs of disease).
Trestbps	Numeric	Resting blood pressure (in mm Hg upon admission)

Attributes	Data Type	Description
Cholestrol	Numeric	Serum cholesterol (mg/dl)
FBS	Categorical	Fasting blood sugar (>120 mg/dl, 0 = false, 1 = true)
Restecg	Categorical	Resting ECG (electrocardiogram) results, values (0 = normal, 1 = mild to severe heart rhythm abnormalities, 2 = enlargement of the main chambers of the heart.
MaxHR	Numeric	Maximum heart rate achieved
Exang	Categorical	Patient's condition during exercise/physical activity, value (0 = no pain, 1 = pain).
Oldpeak	Numeric	Exercise-induced ST depression relative to rest
Slope	Categorical	Decrease or increase in the ST line on the ECG when a person exercises to maximum capacity.
Ca	Categorical	Number of major blood vessels (0-3)
Talasemia	Categorical	Congenital blood disorders (0 = normal; 1 = permanent defect; 2 = reversible defect, 3 = unknown)
Target	Binary	0 = no heart disease, 1 = heart disease

The data collection in this study came from the Kaggle website, which originated from four well-known medical sources, namely: Cleveland, Hungary, Switzerland, and Long Beach V. It was compiled and republished through the Zenodo Platform in 2024 [23]. The dataset consists of 14 attributes for classifying heart conditions with positive and negative classes, covering patient characteristics to target features.

2.2 Data Pre-Processing

Data Preprocessing is the initial stage in machine learning that cleans and prepares data before it is used for model training. The goal is to produce the best quality data through the removal of irrelevant features, data cleaning, and data transformation. Based on the data collection results, there were 1,025 rows, with details of 499 data for patients without heart disease and 526 data for patients with heart disease. The difference between the two classes was 27 rows, so the class distribution showed balance.

2.3 Data Cleaning

Data cleaning is an important stage in which incorrect data is corrected or deleted to maintain the validity of the analysis, but it is time-consuming in data science [24]. Data cleaning is carried out by identifying data, handling missing values, and removing duplicate data.

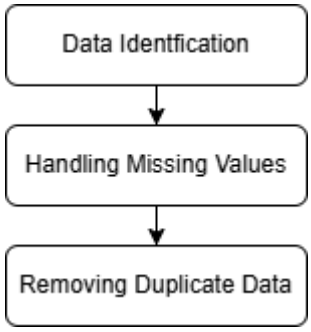

Figure 2. Data Cleaning Stage

The data cleaning process consists of three main steps. First, data identification is performed to understand the condition and quality of existing data. Second, addressing missing values aims to address missing data, either

by filling it in using a specific method or deleting it if it's irrelevant. Third, removing duplicate data is performed to ensure there are no duplicates that could affect the analysis results.

2.4 Data Standardization

Data standardization is performed to equalize the scale of numerical features so that machine learning models work more optimally using StandardScaler, which produces a mean of zero and a standard deviation of one [25], [26].

$$x' = \frac{x - \mu}{\sigma} \quad (1)$$

Explanation:

x = Input data.
M = Data average.
σ = Data standard deviation.

2.5 Feature Selection Corrrelation

This study uses correlation-based feature selection to identify and remove irrelevant and redundant attributes based on their relationship with the target class, as this method helps improve SVM performance by reducing overfitting, enhancing model generalization, and providing a computationally efficient approach compared to more complex feature selection techniques. [27], [28].

$$Corr_{x,y} = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}} \quad (2)$$

Explanation:

$\text{Corr}_{x,y}$ = Correlation value between attribute x and target attribute y.
x_i = Observation value i of variable x.
y_i = Observation value i of variable$y$.
$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$ = Average of variable x.
$\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n}$ = Average of variable y.
n = Number of data pairs (x_i, y_i).
$\sum$ = Average value of all data on attribute x.

2.6 10-Fold Cross Validation

Table 2. 10-Fold Cross Validation

K-Fold	Cross Validation									
1	Test	Train	Train	Train	Train	Train	Train	Train	Train	Train
2	Train	Test	Train	Train	Train	Train	Train	Train	Train	Train
3	Train	Train	Test	Train	Train	Train	Train	Train	Train	Train
4	Train	Train	Train	Test	Train	Train	Train	Train	Train	Train
5	Train	Train	Train	Train	Test	Train	Train	Train	Train	Train
6	Train	Train	Train	Train	Train	Test	Train	Train	Train	Train
7	Train	Train	Train	Train	Train	Train	Test	Train	Train	Train
8	Train	Train	Train	Train	Train	Train	Train	Test	Train	Train
9	Train	Train	Train	Train	Train	Train	Train	Train	Test	Train
10	Train	Train	Train	Train	Train	Train	Train	Train	Train	Test

Data division was performed using the K-Fold Cross Validation technique with K variation, where K = 10 was chosen because it is commonly used, dividing the data into 10 equal folds and repeating it 10 times to produce a more stable and minimally biased model performance evaluation [29], [30].

2.7 Support Vector Machine (SVM)

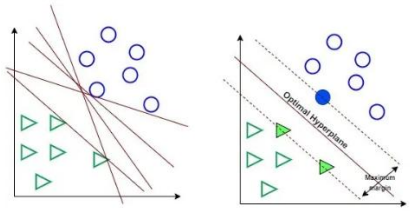

Figure 3. Support Vector Machine (SVM) Modeling

Support Vector Machine (SVM) is a machine learning algorithm that works based on the principle of Structural Risk Minimization (SRM) with the aim of finding the best hyperplane to separate different classes in the data where the optimal separating function is defined as the one that produces the maximum margin between two vectors from different classes and lies in the middle of those vectors and in this study the separating function used is a linear function [16].

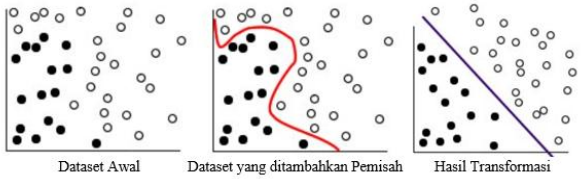

Figure 4. How Support Vector Machines (SVM) Work

A hyperplane is a decision boundary that distinguishes two classes in SVM. Data points that fall on either side of the hyperplane can be associated with different classes [31]. The SVM process begins with data that has not been separated between classes. Through the use of the kernel trick, the data is mapped to a higher-dimensional feature space, enabling optimal linear separation between classes.

$$f(x) = \sum_{i=1}^{n} \alpha i y i e^{-\gamma ||xi-x||^2} + b \quad (3)$$

Explanation:

$f(x)$	= The decision function used to determine the class of an input data.
x	= Input data vector or test data (e.g., blood pressure and other attributes).
xi	= Training data vector i that acts as a support vector.
αi	= Lagrange coefficient that indicates the contribution level of each support vector to the decision function.
yi	= Class label of the i-th training data, usually +1 or −1.
γ	= RBF kernel parameter that regulates the influence of the distance between input data and support vectors on the kernel value.
$e^{-\gamma ||xi-x||^2}$	= Radial Basis Function (RBF) kernel function that calculates the degree of similarity between input data and support vectors based on Euclidean distance.
$\|\|xi - x\|\|^2$	= The squared Euclidean distance between the training data and the input data.
b	= The bias that shifts the decision plane to optimize class separation.
n	= The number of support vectors used in the model.

The proposed model was implemented in Python using the scikit-learn library, with additional support from pandas and NumPy for data preprocessing. The experiments were conducted using Visual Studio Code as the development environment. An SVM classifier with a Radial Basis Function (RBF) kernel was employed. Hyperparameter optimization was performed using Particle Swarm Optimization (PSO), where the search space for C and γ was defined within [0.01–1000] and [10^{-4}–10], respectively. The optimization process yielded optimal values of C = 127.211359 and γ = 0.000100. Model performance was evaluated using 10-fold cross-validation to ensure robustness.

2.8 Support Vector Machine (SVM) + Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) is an optimization technique inspired by the movements and behavior of animals such as fish and birds in their behavior, such as searching for food or prey [19].

$$\mathcal{V}_i^t = \mathcal{w}\mathcal{v}_i^t + c1r1(pbest_i - x_i^t) + c2r2(gbest - x_i^t) \qquad (4)$$
$$x_i^{t+1} = x_i^t + \mathcal{v}_i^{t+1}$$

Explanation:

x_i = Position of the i-th particle (SVM parameters: C, $^{\gamma}$)
$\mathcal{v}_i$ = Particle velocity
$pbest_i$ = Local best position
$gbest$ = Global best position
$\mathcal{w}$ = Inertia weight
$c1, c2$ = Acceleration constant
$r1, r2$ = Random number between 0 and 1

Particle Swarm Optimization (PSO) is used to optimize the Support Vector Machine (SVM) parameters, namely the regularization parameter C and the Radial Basis Function (RBF) kernel parameter γ. PSO works by utilizing a set of particles, each of which represents a candidate solution in the form of a combination of C and γ values. Each particle moves in the search space based on a speed influenced by the inertial weight $\boldsymbol{\mathcal{w}}$, the cognitive acceleration constant (c1) and the social acceleration constant (c2). The inertial weight functions to control the influence of the previous speed, while c1 and c2 regulate the balance between the best experience of individual particles $\boldsymbol{pbest}$ and the best experience of all particles $\boldsymbol{gbest}$. In its implementation,

Particle Swarm Optimization (PSO) was implemented using a custom Python-based approach with 40 particles and a maximum of 50 iterations to balance optimization performance and computational cost. The optimization process was terminated based on the maximum number of iterations. A fixed random seed 42 was applied to ensure reproducibility, and particles were initialized using a uniform random distribution [32].

$$Fitness\ (\ C\ ,\gamma\ ) = \frac{1}{4}\ (\ Accuracy + Precision + Recall + F1 - Score\ ) \qquad (5)$$

The optimization process uses a fitness function that combines Accuracy, Precision, Recall, and F1-score, each assigned an equal weight of 0.25. Fitness(C,γ) represents the objective function, where C and γ are the SVM parameters. Each metric is computed as the average result of 10-fold cross-validation and derived from the confusion matrix. The best parameter combination obtained by PSO is then used to improve SVM classification performance.

2.9 Evaluation

Evaluation is the process of measuring the accuracy of the results of the classification algorithm model implemented using the Confusion Matrix technique. Confusion Matrix is used to measure the performance or performance of the model by calculating the Accuracy, Precision, Recall, and F1-Score values [33].

Table 3. Confusion Matrix

	Positive Prediction	Negative Prediction
Actual Positive	TP	FN
Actual Negative	FP	TN

Explanation:
True Positive (TP) = The number of positive data that are correctly predicted as positive.
True Negative (TN) = The number of negative data that are correctly predicted as negative.
False Positive (FP) = The number of negative data that are incorrectly predicted as positive.
False Negative (FN) = The number of positive data that are incorrectly predicted as negative.

Accuracy is the result of dividing all correct prediction values by the total data [34].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \; x \; 100\% \tag{6}$$

Precision measures the proportion of positive predictions that are actually positive (True Positive divided by total positive predictions) in performing classification [35].

$$Precision = \frac{TP}{TP + FP} \; x \; 100\% \tag{7}$$

Recall is a metric used to measure the extent to which a classification model can identify all actual positive cases. Recall is calculated by comparing true positive predictions to the total number of actual positive cases [36].

$$Recall = \frac{TP}{TP + FN} \; x \; 100\% \tag{8}$$

F1-Score is a better metric than accuracy in class imbalance conditions because it considers data distribution in model performance evaluation [37].

$$F1 - Score = 2 \; x \; \frac{Precision \; x \; Recall}{Precisoon + Recall} \; x \; 100\% \tag{9}$$

2.10 Implementation Details

This study utilized several libraries, including scikit-learn for machine learning, NumPy and pandas for data processing, and Matplotlib and Seaborn for visualization. The development environment used was Visual Studio Code. Experiments were conducted on a system with an Intel Core i3 (11th Generation) processor and 4 GB of RAM running Windows 11. The SVM classifier used a Radial Basis Function (RBF) kernel, with optimized hyperparameters obtained using Particle Swarm Optimization (PSO), resulting in C = 127.21 and $\gamma = 1 \times 10^{-4}$. The PSO algorithm was implemented using a custom Python-based approach with 40 particles and a maximum of 50 iterations. A fixed random seed 42 was applied to ensure reproducibility.

3. Result and Discussions

The research results were analyzed to assess the effectiveness of the method using SVM with PSO optimization on the heart disease dataset from Zenodo, which consisted of 13 features and 1 target.

Table 4. Heart Disease Data

No	Age	Sex	Cp	Trest bps	Chol	Fbs	Reste cg	Thala ch	Exa ng	Oldp eak	Slo pe	Ca	Thal	Tar get
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3	0

No	Age	Sex	Cp	Trest bps	Chol	Fbs	Reste cg	Thala ch	Exa ng	Oldp eak	Slo pe	Ca	Thal	Tar get
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3	0
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3	0
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2	0
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
1020	59	1	1	140	221	1	1	164	1	0.0	2	0	2	1
1021	60	1	0	125	258	0	0	141	1	2.8	1	1	3	0
1022	47	1	0	110	275	0	0	118	1	1.0	1	1	2	0
1023	50	0	0	110	254	0	0	159	0	0.0	2	0	2	1
1024	54	1	0	120	188	0	1	113	0	1.4	1	1	3	0

The dataset, sourced from Zenodo, includes clinical features such as age, sex, chest pain type, blood pressure, cholesterol, and thalassemia, along with a binary target variable indicating heart disease (1 = presence, 0 = absence).

Table 5. Result of Data Cleaning

Explanation	Number of Rows
Number of Rows Before Handling	1,025
Number of rows identified as duplicates and deleted	723
Number of Rows After Handling	302

Results from 1,025 data rows, 723 duplicate records were identified and removed, resulting in 302 instances, consisting of 164 patients with heart disease and 138 without. The high number of duplicates occurred because the dataset was compiled from multiple sources containing identical patient records. Duplicate removal was performed using the df.duplicated() function to prevent bias and data leakage, as repeated records can lead to over-representation of certain patterns and artificially inflated model performance.

To evaluate the impact of duplicate data, an additional experiment was conducted using the original dataset without duplicate removal. In this scenario, the model achieved an unrealistically perfect performance 100% across all evaluation metrics. This indicates the presence of data leakage, where identical records may appear in both training and testing sets during cross-validation, leading to severe overfitting and unreliable evaluation results.

In contrast, after removing duplicate records, the model produced more realistic performance results, reflecting its true generalization ability. Although the dataset size was significantly reduced, the class distribution remained relatively balanced, and the use of 10-fold cross-validation ensured effective utilization of the available data. However, the smaller dataset size may still limit generalization, which is acknowledged as a limitation of this study.

Table 6. Heart Disease Data Result After Standardization

No	Age	Trestbps	Chol	Thalach	Oldpeak
0	-0.257180	-0.355165	-0.709016	0.808268	-0.004522
1	-0.147015	0.529153	-0.905348	0.230779	1.942086
2	1.725795	0.823925	-1.537974	-1.101888	1.478608
3	0.734307	1.000789	-0.905348	0.497312	-0.931478
4	0.844473	0.411244	1.079790	-1.945911	0.829739
…	…	…	…	…	…
723	1.505464	-0.649938	-0.730830	-1.546110	0.458956

No	Age	Trestbps	Chol	Thalach	Oldpeak
733	-1.138503	-1.357392	-2.257860	1.119224	-0.375304
739	-0.257180	-0.178301	0.229016	0.497312	-0.931478
843	0.513977	1.708243	0.621681	-1.101888	-0.931478
878	-0.036850	-0.649938	-1.232569	-1.634955	0.366261

Heart disease data is displayed after the Standardization process using StandardScaler, where each numerical feature is changed to have a mean of zero and a standard deviation of one. This process equalizes the scale between variables so that the classification model can learn more effectively.

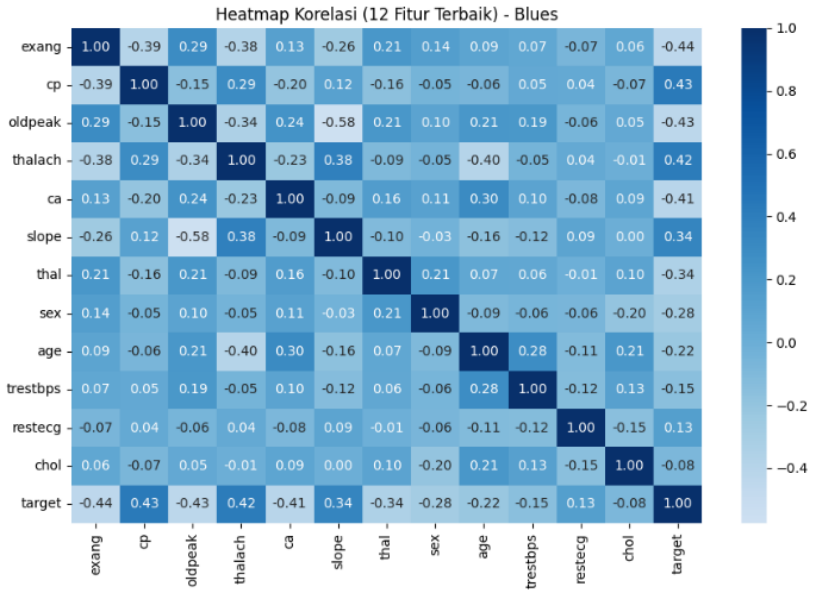

Figure 5. Correlation Heatmap

The results of the correlation calculation between each attribute and the target variable yielded the 12 best features with the highest correlation values: Exang (-0.44), cp (+0.43), oldpeak (-0.43), thalach (+0.42), ca (-0.41), slope (+0.34), thal (-0.34), sex (-0.28), age (-0.22), trestbps (-0.15), restecg (+0.13), and chol (-0.08), thus having a high contribution value and eliminating only 1 feature, fasting blood sugar (fbs), with a low correlation value (-0.02) close to zero, which is considered to have no high contribution in the heart disease classification process.

Table 7. Distribution of 10-Fold Cross Validation

Fold	Training Data	Test Data
1	271	31
2	271	31
3	272	30
4	272	30
5	272	30
6	272	30
7	272	30
8	272	30
9	272	30
10	272	30

Data division was performed using the 10-Fold Cross Validation method with 10 folds on a dataset of 302 rows of data. Since the total data could not be divided evenly, two folds had 31 test data and 271 training data, while eight folds had 30 test data and 272 training data. This division was done automatically and the remaining data was placed in the initial fold.

Table 8. SVM Model Prediction

Fold	Accuracy	Precision	Recall	F1-Score
1	77,42%	70,00%	93,33%	80,00%
2	83,87%	76,47%	92,86%	83,87%
3	83,33%	76,47%	92,86%	83,87%
4	76,67%	76,19%	88,89%	82,05%
5	86,67%	85,71%	94,74%	90,00%
6	90,00%	89,47%	94,44%	91,89%
7	76,67%	76,92%	71,43%	74,07%
8	83,33%	81,82%	94,74%	87,80%
9	90,00%	85,00%	100%	91,89%
10	80,00%	75,00%	93,75%	83,33%
Total	82,80%	79,31%	91,70%	84,88%

The SVM model prediction results with evaluation obtained a total Accuracy of 82.80%, Precision of 79.31%, Recall of 91.70%, F1-Score of 84.88%, and Best Fold was found in Fold 9, then the average Confusion Matrix was obtained.

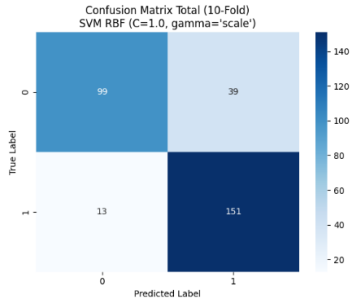

Figure 6. Confusion Matrix SVM

The correct model classified 99 rows of positive data and 151 rows of negative data, while the incorrect model classified 39 rows of positive data as negative and 13 rows of negative data as positive. The calculation results show that the model obtained an Accuracy score of 82.78%, Precision of 79.47%, Recall of 92.07%, and F1-Score of 85.29%, experiencing a small difference that is not significant and therefore does not affect the overall performance of the model.

Table 9. SVM + PSO Model Prediction

Fold	Accuracy	Precision	Recall	F1-Score
1	80,65%	73,68%	93,33%	82,35%
2	80,65%	72,22%	92,86%	81,25%
3	83,33%	76,47%	92,86%	83,87%
4	80,00%	80,00%	88,89%	84,21%
5	90,00%	86,36%	100%	92,68%
6	93,33%	90,00%	100%	94,74%
7	76,67%	76,92%	71,43%	74,07%
8	93,33%	90,48%	100%	95,00%
9	86,67%	84,21%	94,12%	88,89%

Fold	Accuracy	Precision	Recall	F1-Score
10	80,00%	75,00%	93,75%	83,33%
Total	84,46%	80,54%	92,72%	86.04%

The results of the SVM + PSO model evaluation with the Correlation dataset using 10-Fold Cross Validation showed an Accuracy of 84.46%, Precision of 80.54%, Recall of 92.72%, F1-Score of 86.04%, and Best Fold on fold 8. The results of this study are supported by previous research on diabetes classification, where SVM-PSO achieved an accuracy of 83.60%, slightly higher than conventional SVM (83.39%). In comparison, this study obtained a higher accuracy of 84.46%, indicating that the proposed approach performs better. However, differences in performance may be influenced by variations in dataset characteristics, feature selection methods, and experimental settings [22]. In addition, the table also displays the average Confusion Matrix. The performance improvement after PSO optimization indicates that hyperparameter tuning contributes to better class separation and model generalization.

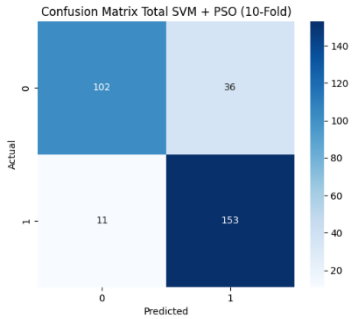

Figure 7. Confusion Matrix SVM + PSO

The correct model classified 102 rows of positive data and 153 rows of negative data, while the incorrect model classified 36 rows of positive data as negative and 11 rows of negative data as positive. Manual calculations using SVM + PSO with the Correlation dataset show that the model obtained an Accuracy of 84.43%, Precision of 80.95%, Recall of 93.29%, and F1-score of 86.74%. These values experienced relatively small changes, thus not significantly affecting the overall performance of the model. Compared to previous studies, the SVM accuracy obtained in this study (84.46%) is higher than the SVM result of 82.49% reported in [10]. and comparable to the 85% accuracy reported in [16]. showing that the proposed method provides competitive performance.

This study analyzes Feature Selection Correlation used to eliminate features with low correlation levels so that only features with significant contributions are used in the classification process. This study applies Correlation-based Feature Selection to eliminate features with low Correlation levels so that only features with significant contributions are used in the heart disease classification process. The results of Correlation-based Feature Selection obtained 13 best features with the highest contribution, while 1 feature, Fasting Blood Sugar (fbs), was eliminated because it had a very low correlation value (-0.02) and was close to zero. Furthermore, model performance was evaluated using a Confusion Matrix with Accuracy, Precision, Recall, and F1-Score metrics.

Table 10. Model Improvement

Modeling	Accuracy	Precision	Recall	F1-Score
SVM	82,80%	79,31%	91,70%	84,88%
SVM + PSO	84,46%	80,54%	92,72%	86,04%
Model Improvement	+2,00%	+1,55%	+1,11%	+1,37%

The test results show that the application of PSO optimization on the SVM model improves performance across all evaluation metrics. The SVM+PSO model produced an Accuracy value of 84.46%, Precision of 80.54%, Recall of 92.72%, and F1-score of 86.04%, which each experienced an increase of 2.00%, 1.55%, 1.11%, and 1.37% compared to SVM without optimization. These findings are further reinforced by research results on stroke cases, where the combination of PSO and SVM was able to increase accuracy from 85% to 94% compared to SVM without optimization [38]. Thus, it can be concluded that PSO is able to effectively improve model performance.

4. Conclusions and Future Works

Based on the results of the research conducted, it can be concluded that the application of Support Vector Machine (SVM) with the Particle Swarm Optimization (PSO) optimization approach and correlation-based feature selection can improve the overall performance of heart disease classification. Although the amount of data used is relatively limited, the application of this method still shows stable and consistent performance through 10-Fold Cross-Validation testing. The results of this study prove that the combination of feature selection and parameter optimization plays an important role in increasing the effectiveness of the heart disease classification model by approximately ±2%. This improvement occurs because PSO is able to search for optimal SVM hyperparameters, especially the C and gamma values in the RBF kernel, which directly affect the decision boundary and classification accuracy. Several studies also show similar findings where PSO improves machine learning performance in heart disease classification. Increase in Random Forest accuracy from 91.4% using Genetic Algorithm (GA) optimization to 95.6% using PSO [22]. This finding is consistent with the results of this study, confirming that PSO is effective as a global optimization method for improving classification model performance.

As a suggestion for further research, the scope of the data can be expanded by adding to the number and variety of datasets so that the PSO-based SVM model has better generalization capabilities. Additionally, other optimization methods such as Genetic Algorithm (GA), Ant Colony Optimization (ACO), or statistical optimization approaches can be explored as alternatives to PSO. Future research is also recommended to test other classification algorithms to compare the effectiveness and stability of methods in heart disease detection.

5. References

[1] WHO, "Cardiovascular diseases (CVDs)," Sep. 14, 2025, *World Health Organization*.

[2] Irfan Sazali Nasution, Arini Dwi Rahmadani, W. Audina, D. P. Sari, and N. D. Sari, "Systematic Review: Pengaruh Gaya Hidup dan Pengetahuan Masyarakat terhadap Risiko Penyakit Jantung Koroner," *Sehat Rakyat: Jurnal Kesehatan Masyarakat*, vol. 4, no. 2, pp. 287–298, 2025, doi: 10.54259/sehatrakyat.v4i2.4337.

[3] D. Saraswati, "Inovasi Pelayanan Kesehatan : Deteksi Dini Penyakit Jantung Koroner melalui Posbindu PTM," *Kesehatan dan Kebidanan Nusantara*, vol. 2, pp. 10–16, 2024, doi: 10.69688/jkn.v2i1.81.

[4] M. Mirbabaie, S. Stieglitz, and N. R. J. Frick, "Artificial intelligence in disease diagnostics: A critical review and classification on the current state of research guiding future direction," *Health and Technology*, vol. 11, no. 4, pp. 693–731, 2021, doi: 10.1007/s12553-021-00555-5.

[5] J. E. Black, J. K. Kueper, and T. S. Williamson, "An introduction to machine learning for classification and prediction," *Family Practice*, vol. 40, no. 1, pp. 200–204, 2023, doi: 10.1093/fampra/cmac104.

[6] F. Reynaldi Valerian, M. Syarief, and D. Abdul Fatah, "Klasifikasi Tingkat Obesitas Menggunakan Metode Gbm Dan Confusion Matrix," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 2, pp. 2242–2249, 2025, doi: 10.36040/jati.v9i2.13062.

[7] I. Akbar, F. Supriadi, and D. Indra Junaedi, "Pemanfaatan Machine Learning Di Bidang Kesehatan," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 1, pp. 1744–1749, 2025, doi: 10.36040/jati.v9i1.12663.

[8] S. Sharma, G. Singh, and P. Negi, "Heart disease Prediction Using Data Mining Techniques," *2024 IEEE 1st Karachi Section Humanitarian Technology Conference, Khi-HTC 2024*, vol. 10, no. 02, pp. 281–286, 2024, doi: 10.1109/KHI-HTC60760.2024.10482141.

[9] Y. Rimal, N. Sharma, S. Paudel, A. Alsadoon, M. P. Koirala, and S. Gill, "Comparative analysis of heart disease prediction using logistic regression, SVM, KNN, and random forest with cross-validation for improved accuracy," *Scientific Reports*, vol. 15, no. 1, pp. 1–14, 2025, doi: 10.1038/s41598-025-93675-1.

[10] L. N. Farida and S. Bahri, "Klasifikasi Gagal Jantung menggunakan Metode SVM (Support Vector Machine)," *Komputika : Jurnal Sistem Komputer*, vol. 13, no. 2, pp. 149–156, 2024, doi: 10.34010/komputika.v13i2.11330.

[11] D. H. Depari, Y. Widiastiwi, and M. M. Santoni, "Perbandingan Model Decision Tree, Naive Bayes dan Random Forest untuk Prediksi Klasifikasi Penyakit Jantung," *Informatik : Jurnal Ilmu Komputer*, vol. 18, no. 3, p. 239, 2022, doi: 10.52958/iftk.v18i3.4694.

[12] R. Resmiati and T. Arifin, "Klasifikasi Pasien Kanker Payudara Menggunakan Metode Support Vector Machine dengan Backward Elimination," *Jurnal Sistem Informasi*, vol. 10, no. 2, pp. 381–393, 2021, doi: 10.32520/stmsi.v10i2.1238.

[13] M. F. R. Angga Aditya Permana, "Implementasi Metode K-Nearest Neighbors (KNN) untuk Klasifikasi Penyakit Jantung," *G-Tech : Jurnal Teknologi Terapan*, vol. 8, no. 1, pp. 186–195, 2024, doi: 10.33379/gtech.v8i3.4495.

[14] R. Hidayat, Y. S. Sy, T. Sujana, M. Husnah, and H. T. Saputra, "Implementation of Machine Learning for Heart Disease Prediction Using Support Vector Machine Algorithm," *Teknologi Informasi dan Rekayasa Komputer*, vol. 5, no. 2, pp. 161–168, 2024, doi: 10.37148/bios.v5i2.152.

[15] M. Hasan, M. A. Sahid, M. P. Uddin, M. A. Marjan, S. Kadry, and J. Kim, "Performance discrepancy mitigation in heart disease prediction for multisensory inter-datasets," *PeerJ Computer Science*, vol. 10, pp. 1–51, 2024, doi: 10.7717/peerj-cs.1917.

[16] D. Irawan, E. B. Perkasa, Y. Yurindra, D. Wahyuningsih, and E. Helmud, "Perbandingan Klassifikasi SMS Berbasis Support Vector Machine, Naive Bayes Classifier, Random Forest dan Bagging Classifier," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 10, no. 3, pp. 432–437, 2021, doi: 10.32736/sisfokom.v10i3.1302.

[17] A. Arifuddin, G. S. Buana, R. A. Vinarti, and A. Djunaidy, "Performance Comparison of Decision Tree and Support Vector Machine Algorithms for Heart Failure Prediction," *Procedia Computer Science*, vol. 234, pp. 628–636, 2024, doi: 10.1016/j.procs.2024.03.048.

[18] A. Ashwini, V. Chirchi, S. Balasubramaniam, and M. A. Shah, "Bio inspired optimization techniques for disease detection in deep learning systems," *Scientific Reports*, vol. 15, no. 1, pp. 1–30, 2025, doi: 10.1038/s41598-025-02846-7.

[19] H. Setiani, A. Sunyoto, and A. Nasiri, "Metode Naïve Bayes dan Particle Swarm Optimization untuk Klasifikasi Penyakit Jantung," *Explore*, vol. 12, no. 2, p. 6, 2022, doi: 10.35200/explore.v12i2.566.

[20] M. G. El-Shafiey, A. Hagag, E. S. A. El-Dahshan, and M. A. Ismail, "A hybrid GA and PSO optimized approach for heart-disease prediction based on random forest," *Multimedia Tools and Applications*, vol. 81, no. 13, pp. 18155–18179, 2022, doi: 10.1007/s11042-022-12425-x.

[21] A. Al Bataineh and S. Manacek, "MLP-PSO Hybrid Algorithm for Heart Disease Prediction," *Journal of Personalized Medicine*, vol. 12, no. 8, 2022, doi: 10.3390/jpm12081208.

[22] A. Agung, G. Agung, and I. M. Widiartha, "Optimasi Metode Support Vector Machine (SVM) Mengunakan Particle Swarm Optimization pada Permasalahan Klasifikasi Diabetes," *JNATIA*, vol. 3, pp. 879–888, 2025, doi: 10.24843/JNATIA.2025.v03.i04.p18.

[23] R. Hidayat, "Dataset Heart Disease," 2024, *Zenodo*. doi: 10.5281/zenodo.13208473.

[24] P. Martins, F. Cardoso, P. Váz, J. Silva, and M. Abbasi, "Performance and Scalability of Data Cleaning and Preprocessing Tools: A Benchmark on Large Real-World Datasets," *Data*, vol. 10, no. 5, pp. 1–22, 2025, doi: 10.3390/data10050068.

[25] C. Fan, M. Chen, X. Wang, J. Wang, and B. Huang, "A Review on Data Preprocessing Techniques Toward Ef fi cient and Reliable Knowledge Discovery From Building Operational Data," vol. 9, no. March, pp. 1–17, 2021, doi: 10.3389/fenrg.2021.652801.

[26] M. Anita *et al.*, "Klasifikasi Faktor Risiko Penyakit Jantung Menggunakan Machine Learning," *Teknologi Informasi*, vol. 16, pp. 68–78, 2025, doi: 10.52972/hoaq.vol16no1.

[27] P. A. Mahasiswa, "Optimasi correlation-based feature selection untuk perbaikan akurasi random forest classifier dalam prediksi performa akademik mahasiswa," *Informatika dan Komputer*, vol. 6, no. 2, pp. 251–260, 2022, doi: 10.26798/jiko.v6i2.651.

[28] E. S. Alomari *et al.*, "Malware Detection Using Deep Learning and Correlation-Based Feature Selection," *Symmetry*, vol. 15, no. 1, pp. 1–21, 2023, doi: 10.3390/sym15010123.

[29] I. K. Nti, "Performance of Machine Learning Algorithms with Different K Values in K-fold Cross-Validation," *I.J. Information Technology and Compuer Science*, vol. 6, pp. 61–71, 2021, doi: 10.5815/ijitcs.2021.06.05.

[30] R. Syaputra, T. A. Y. Siswa, and W. J. Pranoto, "Model Optimasi SVM Dengan PSO-GA dan SMOTE Dalam Menangani High Dimensional dan Imbalance Data Banjir," *Teknika*, vol. 13, no. 2, pp. 273–282, 2024, doi: 10.34148/teknika.v13i2.876.

[31] D. D. Saputra *et al.*, "Analisis Sentimen Terhadap Twitter Direktorat Jenderal Bea dan Cukai Menggunakan komparasi Algoritma Naïve Bayes dan Support Vector Machine," *J-INTECH (Journal of Information and Technology)*, no. 204, pp. 285–296, 2024, doi: 10.32664/j-intech.v12i02.1274.

[32] T. M. Shami, A. A. El-saleh, and S. Member, "Particle Swarm Optimization : A Comprehensive Survey," *Access*, vol. 10, pp. 10031–10061, 2022, doi: 10.1109/ACCESS.2022.3142859.

[33] I. Taufiq, T. A. Y. Siswa, and W. J. Pranoto, "Model Optimasi Random Forest dengan PSO-CHI-SM dalam Mengatasi High Dimensional dan Imbalanced Data Banjir Kota Samarinda," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 7, no. 3, pp. 1267–1279, 2024, doi: 10.32493/jtsi.v7i3.41632.

[34] A. Damuri, U. Riyanto, H. Rusdianto, and M. Aminudin, "Implementasi Data Mining dengan Algoritma Naïve Bayes Untuk Klasifikasi Kelayakan Penerima Bantuan Sembako," *JURIKOM (Jurnal Riset Komputer)*, vol. 8, no. 6, p. 219, 2021, doi: 10.30865/jurikom.v8i6.3655.

[35] L. Techniques, "Enhancing Heart Disease Prediction Accuracy through Machine Learning Techniques and Optimization," *Processe*, vol. 11, no. 4, p. 1210, 2023, doi: 10.3390/pr11041210.

[36] J. C. Obi, "A Comparative Study of Several Classification Metrics and Their Performances on Data," *WJAETS*, vol. 08, no. 01, pp. 308–314, 2023, doi: 10.30574/wjaets.2023.8.1.0054.

[37] A. Churcher *et al.*, "An experimental analysis of attack classification using machine learning in IoT networks," *Sensors (Switzerland)*, vol. 21, no. 2, pp. 1–32, 2021, doi: 10.3390/s21020446.

[38] Y. Ayuningtyas and I. M. Suartana, "Klasifikasi Penyakit Stroke Menggunakan Support Vector Machine (SVM) dan Particle Swarm Optimization (PSO)," *Informatics and Computer Science*, vol. 04, no. 2022, pp. 452–457, 2023, doi: 10.26740/jinacs.v4n04.p451-457.