Jurnal Informatika: Jurnal pengembangan IT
Vol. 11, No. 2, 2026
ISSN : 2477-5126, DOI:10.30591/jpit.v11i2.10240

Model Hybrid Particle Swarm Optimization, Correlation Feature Selection dan Naïve Bayes untuk Deteksi Penyakit Jantung

Renaldi Yoga Rendy Menono [1], Taghfirul Azhima Yoga Siswa [2], Wawan Joko Pranoto[3]
Teknik Informatika, Sains dan Teknologi, Universitas Muhammadiyah Kalimantan Timur
yogamenono@gmail.com, tay758@umkt.ac.id, wjp337@umkt.ac.id

Info Artikel

Riwayat Artikel:

Received 2026-01-25
Revised 2026-08-10
Accepted 2026-08-13

Abstract *– Heart disease is the leading cause of death in Indonesia, with 15.5 million cases reported in 2022. This study aims to classify heart disease using the Naïve Bayes algorithm optimized with Particle Swarm Optimization (PSO) to improve classification performance. The dataset used was obtained from the Zenodo repository, consisting of 1,025 heart disease records with 14 features. Correlation Feature Selection (CFS) was applied to select the most relevant features and eliminate redundant ones, thereby reducing computational complexity and minimizing the risk of model overfitting. The data were then preprocessed and divided into training and testing sets using the 10-fold cross-validation method. PSO was applied to optimize the var_smoothing parameter of the Naïve Bayes algorithm. Model performance was evaluated using a confusion matrix to obtain accuracy, precision, recall, and F1-score. The results show that PSO optimization improved the performance of the Naïve Bayes algorithm. On the correlation-based feature selection dataset, accuracy increased from 81.49% to 84.48%, precision slightly decreased from 82.57% to 82.37%, recall increased from 84.38% to 91.62%, and the F1-score increased from 83.00% to 86.33%. These findings indicate that the combination of Naïve Bayes and Particle Swarm Optimization (PSO) is effective in improving heart disease classification performance..*

Keywords: *Classification*, *Correlation Feature Selection*, *Heart Disease, Naïve Bayes, Particle Swarm Optimization*

Corresponding Author:
Taghfirul Azhima Yoga Siswa
Email: tay758@umkt.ac.id

Abstrak **–** Penyakit jantung adalah penyebab utama kematian di Indonesia, dengan jumlah kasus mencapai 15,5 juta pada tahun 2022. Penelitian ini bertujuan untuk mengklasifikasikan penyakit jantung menggunakan algoritma *Naïve Bayes* dengan optimasi *Particle Swarm Optimization* (PSO) untuk meningkatkan kinerja klasifikasi. Dataset yang digunakan berasal dari repositori Zenodo dengan total 1.025 data penyakit jantung yang terdiri dari 14 fitur. *Correlation Feature Selection* (CFS) diterapkan untuk memilih fitur yang paling relevan dan mengurangi fitur yang bersifat redundan, sehingga dapat menekan beban komputasi serta mengurangi potensi *overfitting* pada model. Data kemudian diproses melalui tahap preprocessing dan dibagi menjadi data latih dan data uji menggunakan metode 10-*Fold Cross Validation*. Optimasi PSO diterapkan pada parameter *var_smoothing* dari *Naïve Bayes*. Evaluasi kinerja model dilakukan menggunakan *confusion matrix* untuk memperoleh nilai *accuracy*, *precision*, *recall*, dan *f1-score*. Hasil penelitian menunjukkan optimasi *Particle Swarm Optimization* (PSO) mampu meningkatkan kinerja algoritma *Naïve Bayes*. Pada dataset hasil seleksi fitur berbasis korelasi, nilai *accuracy* meningkat dari 81,49% menjadi 84,48%, *precision* sedikit menurun dari 82,57% menjadi 82,37%, *recall* meningkat dari 84,38% menjadi 91,62%, dan *f1-score* meningkat dari 83,00% menjadi 86,33%. Temuan ini menunjukkan bahwa kombinasi algoritma *Naïve Bayes* dan *Particle Swarm Optimization* (PSO) efektif dalam meningkatkan performa klasifikasi penyakit jantung.

Kata Kunci*: Klasifikasi, Naïve Bayes, Particle Swarm Optimization, Penyakit Jantung, Seleksi Fitur Berbasis Korelasi*

I. PENDAHULUAN

Penyakit jantung merupakan kondisi ketika jantung tidak dapat berfungsi dengan optimal akibat kerusakan bagian otot jantung yang disebabkan oleh penyempitan arteri coroner [1]. Penyakit ini termasuk dalam kelompok kardiovaskular (CVD) yang menjadi penyebab utama kematian di dunia dengan 19,8 juta kasus pada tahun 2022, 85% di antaranya disebabkan oleh penyakit jantung [2]. Kondisi serupa terlihat di Indonesia, berdasarkan laporan Riskesdas (Riset Kesehatan Dasar) pada tahun 2022 melaporkan bahwa penyakit jantung menjadi salah satu penyakit mematikan dengan kasus sebanyak 15,5 juta di Indonesia [3]. Data ini diperkuat oleh laporan [4] penyakit jantung merupakan penyakit dengan jumlah kasus terbanyak, yaitu mencapai 15.495.666 kasus pada tahun 2022. Penyakit pada jantung beresiko fatal bila terlambat ditangani, karena gejala awal sering diabaikan [5]. Dengan demikian, dibutuhkan sebuah model klasifikasi yang mampu mendeteksi penyakit jantung secara akurat, efisien dan biaya yang terjangkau [6].

Machine Learning merupakan pendekatan kecerdasan buatan yang dirancang untuk meniru perilaku manusia dalam menyelesaikan masalah [7]. *Machine Learning* berfokus pada pengembangan algoritma yang

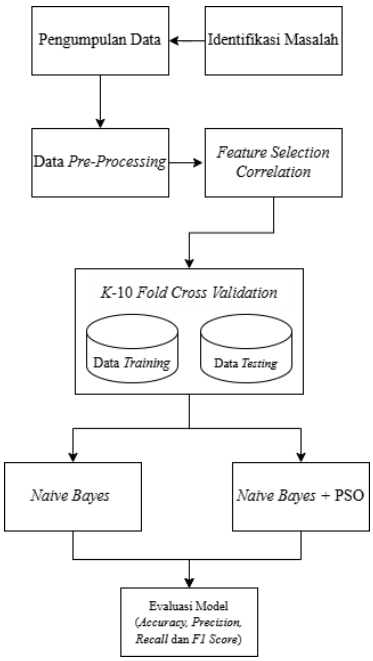
dapat belajar atau menyesuaikan diri terhadap pola data tanpa instruksi pemrograman secara langsung [8]. *Machine Learning* memiliki beberapa teknik untuk mengembangkan suatu model seperti klasifikasi, prediksi, dan pengelompokkan data [9]. Sejumlah penelitian terdahulu terkait klasifikasi data medis telah mengadopsi berbagai algoritma *Machine Learning*, seperti *K-Nearest Neighbor* (KNN), *Support Vector Machine* (SVM), *Random Forest* (RF), *Naïve Bayes* (NB) dan lain-lain [10], [11], [12], [13].

Berbagai penelitian telah menunjukkan bahwa algoritma *Naïve Bayes* memiliki kinerja yang baik pada berbagai kasus klasifikasi medis, meskipun performanya bergantung pada karakteristik dataset yang digunakan. Pada klasifikasi penyakit paru-paru, *Naïve Bayes* mampu mencapai akurasi sebesar 97,06% [14]. Sementara itu, pada penelitian lain mengenai perbandingan algoritma *Machine Learning* pada klasifikasi penyakit jantung didapatkan hasil akurasi *K-Nearest Neighbor* (KNN) 91%, *Logistic Regression* 87%, dan *Naïve Bayes* 83% [15]. Hal senada juga ditunjukkan pada penelitian tentang klasifikasi penyakit jantung menggunakan algoritma *Naïve Bayes* didapat akurasi sebesar 69,96% [16]. Dengan demikian performa algoritma *Naïve Bayes* menurun untuk klasifikasi penyakit jantung namun performanya membaik saat diterapkan pada klasifikasi penyakit paru-paru. *Naïve Bayes* merupakan algoritma klasifikasi berbasis probabilistik yang bekerja dengan menghitung kemungkinan setiap artibut kemudian menggabungkan nilai-nilai tersebut untuk menentukan kelas dari suatu data [17]. Salah satu kelemahan metode ini adalah adanya anggapan bahwa setiap atribut pada data bersifat independen dan tidak saling memengaruhi [18].

Beberapa penelitian yang menerapkan optimasi pada *Naïve Bayes* sudah pernah dilakukan seperti klasifikasi penyakit diabetes menggunakan *Genetic Algorithm* (GA) dan *Naïve Bayes* memperoleh akurasi 93,2% [19]. Selain itu penelitian mengenai klasifikasi data bank *marketing* menggunakan optimasi GA pada *Naïve Bayes* didapatkan hasil akurasi sebesar 89,73% [20]. Sementara itu ada penelitian dalam klasifikasi tingkat keberhasilan *cryotherapy* pada pasien penyakit kutil menggunakan PSO dan *Naïve Bayes* memperoleh akurasi sebesar 97,22% [21]. Penelitian lain tentang klasifikasi teks klinis menggunakan *Naïve Bayes* dan *Ant Colony* (ACO) didapatkan hasil sebesar 81,05% [22]. Sementara itu, penelitian terkait klasifikasi data penerima bantuan sosial dengan pendekatan optimasi *Particle Swarm Optimization* (PSO) pada *Naïve Bayes* diperoleh hasil akurasi sebesar 90% [23]. Temuan tersebut menunjukkan bahwa metode optimasi berpotensi meningkatkan kemampuan klasifikasi *Naïve Bayes*, meskipun implementasinya pada klasifikasi penyakit jantung masih relatif terbatas.

Particle Swarm Optimization (PSO) merupakan teknik optimasi yang memanfaatkan kecerdasan kolektif untuk menentukan solusi terbaik dari suatu variabel dan telah banyak digunakan karena memiliki kecepatan komputasi lebih tinggi dibandingkan dengan algoritma lainnya [24]. Beberapa penelitian sebelumnya yang mengadopsi optimasi PSO pernah dilakukan oleh [25] dalam deteksi penyakit gagal jantung menggunakan PSO dan *Decision Tree* meningkatkan akurasi sebesar 6%. Penelitian lain yang dilakukan oleh [26] mengombinasikan PSO dengan SVM dalam klasifikasi stroke berhasil meningkatkan akurasi sebesar 9%. Sementara itu penelitian yang dilakukan [27] menggunakan PSO dan KNN untuk klasifikasi diabetes didapatkan peningkatan akurasi sebesar 2,21%. Penelitian lainnya yang dilakukan oleh [28] menggabungkan PSO dan *Naïve Bayes* pada klasifikasi status *stunting* menghasilkan peningkatan akurasi 2,37%. Meskipun demikian, penelitian yang mengombinasikan PSO dengan *Naïve Bayes* pada klasifikasi penyakit jantung, khususnya dengan dukungan *Correlation Feature Selection* (CFS), masih terbatas. Oleh karena itu, penelitian ini mengusulkan penerapan CFS dan optimasi *Particle Swarm Optimization* (PSO) pada algoritma *Naïve Bayes* untuk meningkatkan kinerja klasifikasi penyakit jantung.

II. METODE



Gambar 1. Diagram Prosedur Penelitian

Pada Gambar 1 menggambarkan proses pada penelitian ini terdiri dari beberapa tahap utama, dimulai dari identifikasi masalah, pengumpulan data dari dataset penyakit jantung, data *pre-processing*, *correlation feature selection* kemudian dilakukan pembagian data menggunakan metode *k-fold cross validation*. Selanjutnya diterapkan algoritma *Naïve Bayes* dengan optimasi PSO, dan tahap akhir dilakukan evaluasi kinerja menggunakan *accuracy*, *precision*, dan *f1-score*.

A. Identifikasi Masalah

Identifikasi masalah merupakan tahap awal penelitian yang menentukan arah dan fokus penelitian. Permasalahan utama dalam penelitian ini adalah pemilihan metode yang tepat untuk meningkatkan kinerja deteksi penyakit jantung. Meskipun algoritma *Naïve Bayes* banyak digunakan, beberapa penelitian menunjukkan bahwa kinerjanya masih lebih rendah dibandingkan algoritma lain, sehingga diperlukan penerapan optimasi *Particle Swarm Optimizatio*n (PSO) untuk meningkatkan performa klasifikasi.

B. Pengumpulan Data

Dataset yang digunakan dalam penelitian ini diperoleh dari repositori *Zenodo* yang merupakan hasil penggabungan beberapa dataset penyakit jantung yang bersumber dari *Kaggle*. Dataset tersebut berasal dari empat institusi medis, yaitu *Cleveland, Hungary, Switzerland*, *dan Long Beach V*.

TABEL 1
ATRIBUT DATA PENYAKIT JANTUNG

No	Atribut	Tipe Data	Deskripsi
1	*Age*	*Integer*	Usia pasien (tahun).
2	*Sex*	*Category*	Jenis kelamin *Sex* (1: Laki-laki, 0: Perempuan).
3	*Cp*	*Category*	Nyeri dada (0: akibat suplai darah ke jantung berkurang, 1: tidak terkait jantung, 2: kejang esofagus, 3: tidak menunjukkan tanda penyakit).
4	*RestBP*	*Integer*	Tekanan darah saat istirahat (mmHg)
5	*Cholesterol*	*Integer*	Kolesterol
6	*Fbs*	*Category*	*Fasting Blood Sugar* (1: >120mg/dl, 0: jika tidak).
7	*Restecg*	*Category*	Hasil EKG saat istirahat (0: Normal, 1: Kelainan ringan hingga berat pada detak jantung, 2: Pembesaran ruang utama jantung).
8	*Thalach*	*Integer*	Detak jantung maksimum
9	*Exang*	*Category*	Kondisi pasien saat beraktifitas/ olahraga (0: tidak nyeri, 1: nyeri)
10	*Oldpeak*	*float*	Depresi ST saat melakukan aktifitas.

11	*Slope*	*Category*	Kemiringan segmen ST Latihan puncak (0: *Upsloping*, 1: *Flatsloping*, 2: *Downslopins*)
12	*Ca*	*Category*	Jumlah pembuluh utama yang diwarnai *fluoroskopi*.
13	*Thal*	*Category*	Kelainan darah bawaan (0: normal, 1: kerusakan permanen, 2: kelainan yang dapat kembali normal).
14	*Target*	*Category*	Label penyakit jantung (1: ada penyakit, 0: normal).

Tabel 1 mendeskripsikan atribut yang digunakan dalam data penyakit jantung. Beberapa istilah pada Tabel 1 akan diberikan penjelasan tambahan agar dapat dengan mudah dipahami. *Chest Pain* (CP) adalah jenis nyeri dada yang terbagi menjadi tiga jenis, nyeri akibat suplai darah ke jantung berkurang, nyeri dada yang tidak terkait pada jantung dan nyeri dada yang tidak menunjukkan tanda penyakit. *Fasting Blood Sugar* (FBS) merupakan kadar gula yang diukur setelah pasien berpuasa, hal ini umumnya digunakan sebagai gejala indikasi awal untuk mendeteksi risiko penyakit jantung. *Restecg* mengacu pada kondisi penurunan pada hasil elektrodiaogram (EKG) yang menjadi tanda adanya gangguan aliran darah ke jantung. *Slope* atau kemiringan segmen dibedakan menjadi tiga kategori yaitu, *upsloping* (garis naik), *flat* (garis datar), dan *downsloping* (garis menurun) yang masing masing dapat mengindikasikan kondisi normal maupun adanya iskemia. *Ca* adalah jumlah pembuluh darah utama yang terlihat melalui pemeriksaan fluoroskopi, yaitu teknik pencitraan berbasis sinar-x. Sementara itu, *thal* merupakan kelainan darah turunan yang mempengaruhi produksi hemoglobin, dalam dataset ini dijadikan menjadi tiga kategori yaitu, normal, *fixed defect* (kerusakan permanen), dan *reversible defect* (kelainan yang dapat kembali normal).

C. Data Pre-Processing

1. Data Cleaning

Data Cleaning merupakan tahapan yang bertujuan untuk memastikan tidak ada data yang hilang, duplikat, dan tidak konsisten sehingga dataset siap digunakan dalam analisis maupun permodelan [29]. Dataset yang digunakan pada penelitian ini diperoleh dari repositori *Zenodo* dengan jumlah awal sebanyak 1.025 data. Pada tahap *preprocessing*, dilakukan proses penghapusan data duplikat berdasarkan keseluruhan atribut sehingga jumlah data berkurang menjadi 303 data unik. Dataset hasil *preprocessing* tersebut selanjutnya digunakan sebagai data penelitian pada proses pelatihan dan evaluasi model menggunakan metode *10-Fold Cross Validation*.

2. Data Standardization

Pada tahap ini dilakukan data scaling untuk memastikan keakuratan pemodelan prediktif ketika terdapat perbedaan skala antar fitur. *Data standardization* merupakan salah satu metode yang digunakan dengan cara menormalkan data berdasarkan rata-rata menggunakan deviasi standar satuan [30]. Persamaan *Standardization* [30].

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

Keterangan:
x : Data yang di input.
μ : Rata-rata data.
σ : Standar deviasi data.

D. Correlation Feature Selection

Pada penelitian ini menggunakan *Correlation feature Selection* (CFS) untuk menganalisis tingkat hubungan antara fitur-fitur yang ada dalam dataset [31]. *Correlation Feature Selection* berfungsi untuk menyeleksi atribut yang tidak diperlukan [32]. Nilai korelasi divisualisasikan dalam bentuk gradasi warna dengan rentang nilai antar -1 hingga 1. Korelasi positif direpresentasikan dengan warna yang lebih Gelap, sedangkan korelasi negatif ditunjukkan dengan warna yang lebih terang. Persamaan (CFS) [33].

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

Keterangan:

$\text{Corr}_{x,y}$: Nilai korelasi antara atribut x dan atribut target *y*.
x_i : Nilai observasi ke-*i* dari variabel *x.*
y_i : Nilai observasi ke-*i* dari variabel *y.*
$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$: Rata-rata dari variabel *x.*
$\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n}$: Rata-rata dari variabel *y.*
n : Jumlah pasangan data (x_i, y_i).
$\sum$: Nilai rata-rata dari seluruh data pada atribut x.

E. K-10 Fold Cross Validation

K-Fold Cross Validation digunakan dalam *data mining* untuk meningkatkan akurasi secara optimal dengan membagi data menjadi dua bagian, yaitu data latih dan data uji [34]. Salah satu cara pengujian dalam *k-fold cross validation* yang digunakan untuk mengevaluasi kinerja model dengan cara membagi data sampel secara acak ke dalam K kelompok [35]. Pada penelitian ini memanfaatkan *metode k-fold cross validation* dengan nilai K-10, yang di pilih karena mampu mendapatkan akurasi yang optimal [36].

TABEL 2
SKEMA K-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*

Tabel 2 menunjukkan skema *10 K-Fold Cross Validation.* Pada tahap pertama, fold 1 digunakan sebagai data uji (*test*), sedangkan *fold* 2 hingga *fold* 10 berfungsi sebagai data latih (*train*). Dengan demikian, model dilatih menggunakan data dari *fold* 2 sampai 10 kemudian diuji pada data *fold* 1. Selanjutnya pada tahap kedua, *fold* 2 dijadikan sebagai data uji, sementara itu *fold* 1 serta *fold* 3 sampai 10 digunakan sebagai data latih. Model dilatih pada data tersebut, kemudian diuji dengan data *fold* 2, dan hasil evaluasinya dicatat sebagai *fold* 2. Dengan cara ini, model selalu menggunakan sekitar 90% data untuk pelatihan dan 10% data berbeda untuk pengujian, sehingga proses evaluasi menjadi lebih merata dan tidak bergantung pada satu kali pemisahan data latih-uji.

F. Naïve Bayes

Naïve Bayes adalah algoritma klasifikasi yang bekerja dengan menghitung probabilitas suatu data berdasarkan nilai-nilai atribut yang dimilikinya. Dalam proses klasifikasi Naïve Bayes tidak membutuhkan jumlah data pelatihan yang besar untuk memperoleh estimasi parameter yang akurat [17].

$$P(H|X) = \frac{\left(P(X|Ci) * P(Ci) \right)}{P(X)} \qquad (2)$$

Keterangan:

X : Data baru dengan kelas yang belum diketahui.
Ci : Hipotesis atau kelas tertentu.
P(Ci|X) : Probabilitas hipotesis *Ci* berdasarkan data *X* (*Posterior Probability*).
P(Ci) : Probabilitas hipotesis *Ci* (*Prior Probabilitas*).
P(X|Ci) : Probabilitas data *X* berdasarkan kondisi *Ci*.
P(X) : Probabilitas dari semua data *X* yang ada.

Pada penelitian ini digunakan model Gaussian Naïve Bayes, yaitu varian Naïve Bayes yang digunakan untuk data numerik atau kontinu. Gaussian Naïve Bayes mengasumsikan bahwa setiap fitur mengikuti distribusi normal (gaussian) pada masing-masing kelas [37].

$$P(x_i \mid y_k) = \frac{1}{\sqrt{2\pi\sigma_{y,k}^2}} \exp\left(-\frac{(x_i - \mu_{y,k})^2}{2\sigma_{y,k}^2}\right) \quad (3)$$

Keterangan:

x_i : Nilai dari fitur ke-*i* dari data yang akan di klasifikasikan.
$\mu_{y,k}$: *Mean* dari kelas y_k
$\sigma_{y,k}$: *Varians* dari kelas y_k
exp : Fungsi eksponensial.

Persamaan tersebut digunakan untuk menentukan nilai probabilitas suatu variabel pada distribusi normal dengan memanfaatkan mean dan varians, sehingga mendukung proses pengambilan keputusan dalam klasifikasi berbasis probabilistik.

G. Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) adalah algoritma optimisasi yang bekerja dengan pendekatan populasi, yang prinsip dasarnya diambil dari prilaku kolektif kawanan hewan seperti burung atau ikan [38]. Pada penelitian ini, proses optimasi menggunakan *Particle Swarm Optimization* (PSO) dilakukan dengan jumlah partikel (*n_particles*) sebanyak 20 dan jumlah iterasi maksimum (*n_iterations*) sebanyak 30. Parameter bobot inersia (*inertia weight*) ditetapkan sebesar 0,7, sedangkan koefisien percepatan kognitif (*c1*) dan sosial (*c2*) masing-masing bernilai 1,5. Optimasi dilakukan terhadap parameter *var_smoothing* pada algoritma *Naïve Bayes* dengan rentang pencarian sebesar 1×10^{-9} hingga 1. Konfigurasi parameter tersebut digunakan untuk memperoleh nilai *var_smoothing* yang optimal sehingga dapat meningkatkan kinerja model klasifikasi.

$$V_k^{(i+1)} = V_k^{(i)} + c_1 r_1 \left(pbest_{k,i} - X_k^{(i)}\right) + c_2 r_2 \left(gbest_i - X_k^{(i)}\right) \quad (4)$$

Keterangan:

$V_k^{(i+1)}$: Kecepatan partikel ke -*k* pada iterasi ke (*i* +1).
$V_k^{(i)}$: Kecepatan partikel ke -*k* pada iterasi ke -*i*.
$X_k^{(i)}$: Poisisi partikel ke -*k* pada iterasi ke -*i*.
$pbest_{k,i}$: Posisi terbaik individu partikel ke -*k* pada iterasi ke -*i*.
$gbest_i$: Posisi terbaik dari seluruh partikel pada iterasi ke -*i*.
c_1, c_2 : Koefisien percepatan yang menentukan pengaruh kedua posisi terbaik terhadap kecepatan artikel.
$r_1 r_2$: Bilangan acak dalam rentang 0-1 yang digunakan untuk memastikan pencarian tetap terjaga.

H. Confusion Matrix

Confusion Matrix menjadi dasar dalam proses perhitungan dengan memberikan representasi visual mengenai hubungan antara hasil klasifikasi model dan kelas aktual [39]. *confusion matrix* dapat digunakan untuk mengevaluasi kinerja model melalui metrik seperti, *accuracy*, *precision*, *recall*, dan *f1-score* [40].

1. Accuracy

Perhitungan *accuracy* dihitung dengan membandingkan jumlah data yang diklasifikasi dengan benar terhadap total keseluruhan data [41].

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \times 100\% \quad (6)$$

2. Precision

Precision merupakan perhitungan yang membandingkan jumlah prediksi positif yang benar dengan total prediksi positif sehingga menunjukkan tingkat ketepatan hasil klasifikasi [42].

$$Precision = \frac{TP}{(TP+FP)} \times 100\% \quad (7)$$

3. *Recall*

Recall adalah ukuran yang menilai seberapa banyak data positif yang telah berhasil di identifikasi dengan benar oleh model dari semua data positif yang ada [42].

$$Recall = \frac{TP}{(TP+FN)} \times 100\% \quad (8)$$

4. *F1-Score*

F1-Score merupakan rata rata harmonik antara precision dan recall yang berfungsi untuk menilai keseimbangan kinerja suatu model [43]. Berikut perhitungan dari accuracy yang dapat dilihat pada persamaan 9.

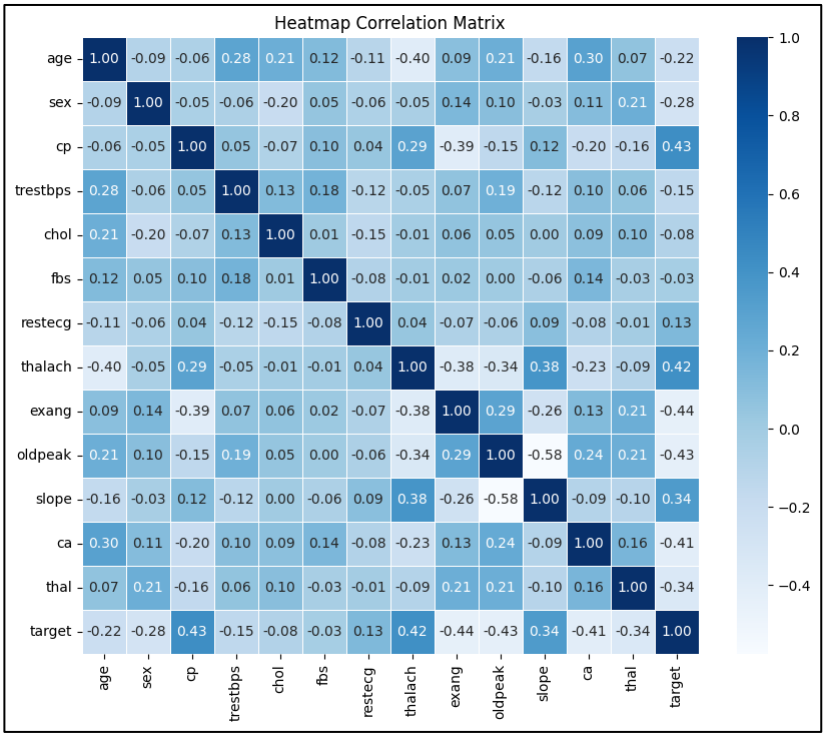
$$F1\text{-}Score = 2 \times \frac{(Presisi\ x\ Recall)}{(Presisi + Recall)} \quad (9)$$

Keterangan:

True Positive (TP)	:	Jumlah data positif yang diklasifikasikan dengan benar sebagai positif.
True Negative (TN)	:	Jumlah data negatif yang diklasifikasikan dengan benar sebagai negatif.
False Positive (FP)	:	Jumlah data negatif yang diklasifikasikan sebagai positif.
False Negative (FN)	:	Jumlah data positif yang salah di klasifikasikan.
Precision	:	Mengukur tingkat ketepatan klasifikasi terhadap kelas positif
Recall	:	Mengukur sejauh mana model mampu mengidentifikasi seluruh *instance* yang sebenarnya masuk kedalam kelas positif.

III. HASIL DAN PEMBAHASAN

A. *Hasil Seleksi Fitur Menggunakan CFS*



Gambar 2. Hasil *Correlation Heatmap*

Gambar 2 menunjukkan Hasil seleksi fitur menggunakan *correlation* bahwa fitur *exang* memiliki nilai korelasi tertinggi terhadap variabel *target* sebesar 0.4356, diikuti oleh cp +0.43, *oldpeak* -0.42, dan *thalach* +0.42, yang mengindikasikan kontribusi yang lebih signifikan dalam memengaruhi target. Fitur ca menunjukkan korelasi sedang sebesar -0.41, sedangkan *slope* dan *thal* masing-masing memiliki nilai korelasi +0.34 dan -0.34. Fitur *sex*

-0.28, *age* -0.22, *trestbps* -0.15, *restecg* +0.13, dan *chol* -0.08 memiliki korelasi yang relatif rendah terhadap target, namun tetap digunakan dalam pemodelan karena masih mengandung informasi yang relevan. Sementara itu, fitur fbs menunjukkan nilai korelasi paling lemah terhadap *target* sehingga tidak digunakan pada tahap pemodelan selanjutnya.

B. Hasil Evaluasi Naïve Bayes

TABEL 3
HASIL MODEL NAÏVE BAYES

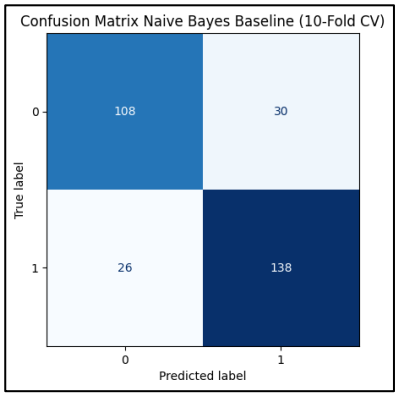
Fold	*Accuracy*	*Precision*	*Recall*	*F1-Score*
1	77.42%	85.71%	70.59%	77.42%
2	74.19%	84.62%	64.71%	73.33%
3	90.00%	88.24%	93.75%	90.91%
4	80.00%	72.73%	100.00%	84.21%
5	90.00%	88.24%	93.75%	90.91%
6	76.67%	80.00%	75.00%	77.42%
7	96.67%	94.12%	100.00%	96.97%
8	76.67%	76.47%	81.25%	78.79%
9	76.67%	77.78%	82.35%	80.00%
10	76.67%	77.78%	82.35%	80.00%

Pada Tabel 3 merepresentasikan hasil pengujian model *Naïve Bayes* menggunakan metode *K-10 Fold Cross Validation*, Berdasarkan hasil pengujian tersebut, nilai kinerja tertinggi dicapai pada *fold* ke 7 dengan *accuracy* sebesar 96,67%, *precision* 94,12%, *recall* 100,00%, dan *f1-score* 96,97%. Sementara itu, nilai terendah tercatat dengan *accuracy* 74,19%, *precision* 76,47%, *recall* 64,71%, dan *f1-score* 73,33%. Variasi nilai pada setiap fold menunjukkan adanya perbedaan distribusi data latih dan data uji pada proses validasi silang.

TABEL 4
HASIL RATA-RATA MODEL NAÏVE BAYES

Hasil rata-rata			
Accuracy	*Precision*	*Recall*	*F1-Score*
81.49%	82.57%	84.38%	83.00%

Pada Tabel 4 menunjukkan hasil pengujian rata rata model *Naïve Bayes* mendapat nilai *accuracy* sebesar 81.49%, *precision* 82.57%, *recall* 84.38% dan *f1-score* 83.00%. Nilai ini merupakan performa akhir dari model *Naïve Bayes* tanpa menggunakan optimasi *Particle Swarm Optimization* dan digunakan sebagai acuan pembanding pada tahap optimasi selanjutnya. Tampilan *confusion matrix* dari hasil permodel *Naïve Bayes* dapat dilihat pada Gambar 2.



Gambar 3. *Confusion Matrix* model *Naïve Bayes*

Pada Gambar 3 merepresentasikan bahwa dari total 302 data, model berhasil mengklasifikasikan 108 data sebagai *true negative* (TN) dan 138 data sebagai *true positive* (TP). Sementara itu, terdapat 30 data *false positive* (FP) dan 26 data *false negative* (FN). Berdasarkan *Confusion Matrix* tersebut, nilai *accuracy* diperoleh dari

perbandingan jumlah prediksi yang benar terhadap keseluruhan data, yaitu 81,49%. Selain itu, nilai *precision* 82,14%, *recall* 84,15%, dan *f1-score* 83,13% menunjukkan bahwa model memiliki kemampuan yang baik dan relatif seimbang dalam mengklasifikasikan penyakit jantung.

C. Hasil Evaluasi Naïve Bayes dengan optimasi PSO

TABEL 5
HASIL MODEL NAÏVE BAYES-PSO

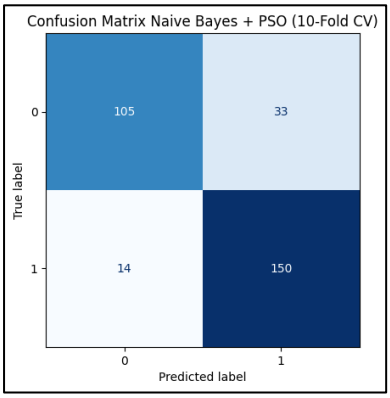
Fold	*Accuracy*	*Precision*	*Recall*	*F1-Score*
1	83.87%	87.50%	82.35%	84.85%
2	70.97%	78.57%	64.71%	70.97%
3	93.33%	88.89%	100.00%	94.12%
4	80.00%	72.73%	100.00%	84.21%
5	83.33%	78.95%	93.75%	85.71%
6	83.33%	82.35%	87.50%	84.85%
7	93.33%	93.75%	93.75%	93.75%
8	86.67%	80.00%	100.00%	88.89%
9	83.33%	80.00%	94.12%	86.49%
10	86.67%	80.95%	100.00%	89.47%

Tabel 5 menunjukkan hasil pengujian model *Naive Bayes*-PSO menggunakan *metode K-10 Fold Cross Validation*. Berdasarkan hasil pengujian, nilai kinerja tertinggi dicapai pada *fold* ke 3 dengan *accuracy* sebesar 93,33%, *precision* sebesar 93,75%, *recall* sebesar 100,00%, dan *f1-score* sebesar 94,12%. Sementara itu, nilai terendah tercatat dengan *accuracy* sebesar 70,97%, *precision* sebesar 72,73%, *recall* sebesar 64,71%, dan *f1-score* sebesar 70,97%.

TABEL 6
HASIL RATA-RATA MODEL NAÏVE BAYES-PSO

Hasil rata-rata			
Accuracy	*Precision*	*Recall*	*F1-Score*
84.48%	82.37%	91.62%	86.33%

Pada Tabel 6 menunjukkan hasil pengujian model *Naïve Bayes* dengan optimasi PSO berdasarkan pengujian menggunakan metode *K-10 Fold Cross Validation*. Dari hasil evaluasi tersebut model menghasilkan nilai *accuracy* sebesar 84.48%, *precision*, sebesar 82,37%, *recall* sebesar 91,62%, *f1-score* sebesar 86,33%. Hal ini menunjukkan bahwa penerapan PSO sebagai optimasi pada algoritma *Naïve Bayes* mampu mempertahankan peforma klasifikasi model secara keseluruhan. Tampilan *confusion matrix* dari hasil permodel *Naïve Bayes* dapat dilihat pada Gambar 3.



Gambar 4. *Confusion Matrix* model *Naïve Bayes* dengan optimasi PSO

Pada Gambar 4 Menunjukkan *confusion matrix* hasil pengujian model *Naïve Bayes*-PSO menggunakan metode *K-10 Fold Cross Validation*. Dari total 302 data, model berhasil mengklasifikasikan 105 data sebagai *true negative* (TN) dan 150 data sebagai *true positive* (TP). Sementara itu, terdapat 33 data *false positive* (FP) dan 14

data *false negative* (FN). Berdasarkan hasil tersebut, nilai *accuracy* dihitung dari perbandingan jumlah prediksi yang benar terhadap keseluruhan data, sehingga diperoleh nilai *accuracy* sebesar 84,48%. Selain itu, nilai *precision* sebesar 81,97%, *recall* sebesar 91,46%, dan *f1*-score sebesar 86,57% menunjukkan bahwa model memiliki kemampuan yang cukup baik dan seimbang dalam mengklasifikasikan penyakit jantung, khususnya dalam mendeteksi kelas positif.

D. Pembahasan

Hasil penelitian ini berupa perbandingan kinerja model klasifikasi yang menggunakan algoritma *Naïve Bayes-PSO*. Fokus utama penelitian ini diarahkan pada analisis kinerja model dengan menghitung hasil rata-rata accuracy sebelum dan sesudah diterapkan optimasi pada setiap *fold* dapat dilihat pada dilihat pada Tabel 7.

TABEL 7
HASIL PERBANDINGAN MODEL NAÏVE BAYES-PSO

Fold	*Naïve Bayes*	*Naive Bayes-PSO*	Perubahan *Naive Bayes-PSO*
1	77,42%	83,87%	+6.45%
2	74,19%	70,97%	−3.22%
3	90,00%	93,33%	+3.33%
4	80,00%	80,00%	0.00%
5	90,00%	83,33%	−6.67%
6	76,67%	83,33%	+6.66%
7	96,67%	93,33%	−3.34%
8	76,67%	86,67%	+10.00%
9	76,67%	83,33%	+6.66%
10	76,67%	86,67%	+10.00%
Rata-rata *Accuracy*	81,49%	84,48%	+2.99%

Pada Tabel 7 tersebut menunjukkan perbandingan nilai *accuracy* antara model *Naïve Bayes* dan *Naïve Bayes-PSO* pada setiap *fold* dalam *K-10 Fold Cross Validation*. Hasil pengujian memperlihatkan bahwa optimasi PSO mampu meningkatkan nilai *accuracy* pada sebagian besar *fold*, dengan peningkatan tertinggi sebesar +10,00% pada *Fold* 8 dan *Fold* 10, meskipun pada beberapa *fold* terjadi penurunan atau tidak terdapat perubahan kinerja akibat perbedaan distribusi data latih dan data uji. Secara keseluruhan, nilai rata-rata *accuracy* mengalami peningkatan dari 81,49% pada model *Naïve Bayes* menjadi 84,48% pada model *Naïve Bayes-PSO* atau meningkat sebesar +2,99%, yang menunjukkan bahwa penerapan PSO secara umum efektif dalam meningkatkan kinerja klasifikasi model *Naïve Bayes*. Analisis difokuskan pada nilai rata-rata *accuracy, precision, recall* dan *f1-score* dapat dilihat pada Tabel 8.

TABEL 8
HASIL PERBANDINGAN RATA-RATA MODEL NAÏVE BAYES

Metrik	*Naïve Bayes*	*Naïve Bayes-PSO*	Perubahan *Naïve Bayes-PSO*
Accuracy	81,49%	84,48%	+2,99%
Precision	82,57%	82,37%	-0,20%
Recall	84,38%	91,62%	+7,24%
F1-Score	83,00%	86,33%	+3,33%

Tabel 8 Menunjukkan perbandingan kinerja pada dua skenario model menunjukkan bahwa optimasi parameter *var_smoothing* menggunakan PSO secara umum meningkatkan performa algoritma *Naïve Bayes*. Berdasarkan hasil evaluasi, peningkatan terbesar terjadi pada metrik *recall* sebesar +7,24%, diikuti oleh *accuracy* sebesar +2,99% dan *f1-score* sebesar +3,33%. Sementara itu, nilai *precision* mengalami sedikit penurunan sebesar −0,20%, namun perubahannya relatif kecil sehingga tidak berdampak signifikan terhadap kinerja model secara keseluruhan. Penurunan tersebut mengindikasikan adanya sedikit peningkatan *False Positive* sebagai konsekuensi dari meningkatnya sensitivitas model dalam mengidentifikasi kelas positif. Hal ini sejalan dengan proses optimasi PSO yang menggunakan *f1-score* sebagai fungsi objektif, sehingga model cenderung menghasilkan keseimbangan yang lebih baik antara *precision* dan *recall*. Dalam konteks klasifikasi penyakit jantung, peningkatan *recall* menjadi lebih penting karena mampu mengurangi risiko *False Negative*, yaitu kondisi ketika pasien yang sebenarnya menderita penyakit jantung diprediksi sebagai sehat.. Hasil ini sejalan dengan penelitian yang menggunakan model *Naïve Bayes*-PSO menghasilkan peningkatan accuracy sebesar

+2,37% [28]. Peningkatan akurasi yang relatif kecil, yaitu sekitar 2–3%, menunjukkan bahwa model Naïve Bayes pada dasarnya telah memiliki kinerja awal yang cukup baik, sehingga ruang optimasi yang dapat dimanfaatkan oleh *Particle Swarm Optimization* (PSO) menjadi terbatas. Selain itu, karakteristik data medis yang memiliki keterkaitan antar atribut menyebabkan asumsi independensi pada *Naïve Bayes* tidak sepenuhnya terpenuhi, sehingga peningkatan akurasi tidak dapat terjadi secara signifikan. Oleh karena itu, penerapan PSO lebih berkontribusi pada peningkatan keseimbangan kinerja model, khususnya pada nilai *recall* dan *f1-score*, dibandingkan menghasilkan peningkatan akurasi yang besar, sehingga capaian akurasi di kisaran 80% merupakan hasil yang wajar dan konsisten dengan penelitian sebelumnya. Dengan demikian, optimasi PSO pada model *Naïve Bayes* terbukti efektif dalam meningkatkan kinerja model klasifikasi.

IV. SIMPULAN

Berdasarkan hasil penelitian, penerapan optimasi *Particle Swarm Optimization* (PSO) pada algoritma *Naïve Bayes* terbukti mampu meningkatkan kinerja model klasifikasi penyakit jantung. pada dataset hasil seleksi fitur berbasis *correlation* didapatkan peningkatan kinerja model dengan nilai *accuracy* dari 81.49% menjadi 84.48%, *recall* dari 84.38% menjadi 91.62%, dan *f1-score* dari 83.00% menjadi 86.33%, meskipun nilai *precision* mengalami penurunan kecil sebesar 0.20%. Hasil ini menunjukkan bahwa optimasi PSO efektif dalam meningkatkan sensitivitas dan kinerja keseluruhan model *Naïve Bayes*, terutama ketika dikombinasikan dengan seleksi fitur berbasis korelasi, sehingga model yang dihasilkan lebih baik untuk mendukung proses klasifikasi penyakit jantung. Dalam penelitian selanjutnya disarankan untuk menggunakan dataset dengan jumlah yang lebih besar dan beragam yang mungkin dapat meningkatkan kemampuan model dalam deteksi penyakit jantung. Selain itu, penelitian selanjutnya juga dapat mengeksplorasi metode optimasi lain, seperti *Genetic Algorithm* (GA), *Ant Colony Opimization* (ACO), atau metode berbasis statistik untuk memperoleh parameter model yang lebih optimal.

DAFTAR PUSTAKA

[1] D. Pradana, M. Luthfi Alghifari, M. Farhan Juna, and D. Palaguna, "Klasifikasi Penyakit Jantung Menggunakan Metode Artificial Neural Network," *Indones. J. Data Sci.*, vol. 3, no. 2, pp. 55–60, 2022, doi: 10.56705/ijodas.v3i2.35.

[2] W. H. Organization, "Cardiovascular diseases (CVDs)." Accessed: Sep. 12, 2025. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)

[3] R. I. Sari, D. W. Apriani, Y. Fauziah, and R. T. A. Rahman, "Edukasi Kesehatan Mengenai Penyakit Jantung Koroner Bagi Warga Desa Simpang Empat Sungai Baru Kabupaten Tanah Laut," *J. Pengabdi. Masy. dan Ris. Pendidik.*, vol. 4, no. 1, pp. 4455–4460, 2025, doi: 10.31004/jerkin.v4i1.2315.

[4] Kementerian Kesehatan Republik Indonesia, "Profil Kesehatan Indonesia 2022." Accessed: Sep. 12, 2025. [Online]. Available: https://kemkes.go.id/id/indonesia-health-profile-2022

[5] A. Sepharni, I. E. Hendrawan, and C. Rozikin, "Klasifikasi Penyakit Jantung dengan Menggunakan Algoritma C4.5," *STRING (Satuan Tulisan Ris. dan Inov. Teknol.*, vol. 7, no. 2, p. 117, 2022, doi: 10.30998/string.v7i2.12012.

[6] B. Hirwono, A. Hermawan, and D. Avianto, "Implementation of the Naïve Bayes Method for Heart Disease Patient Classification," *J. JTIK (Jurnal Teknol. Inf. dan Komunikasi)*, vol. 7, no. 3, pp. 450–457, 2023, doi: 10.35870/jtik.v7i3.910.

[7] M. Ula, A. F. Ulva, and M. Mauliza, "Implementasi Machine Learning Dengan Model Case Based Reasoning Dalam Mendagnosa Gizi Buruk Pada Anak," *J. Inform. Kaputama*, vol. 5, no. 2, pp. 333–339, 2021, doi: 10.59697/jik.v5i2.267.

[8] R. Kurniawan, P. B. Wintoro, Y. Mulyani, and M. Komarudin, "Implementasi Arsitektur Xception Pada Model Machine Learning Klasifikasi Sampah Anorganik," *J. Inform. dan Tek. Elektro Terap.*, vol. 11, no. 2, pp. 233–236, 2023, doi: 10.23960/jitet.v11i2.3034.

[9] N. Hasdyna, R. K. Dinata, Rahmi, and T. I. Fajri, "Hybrid Machine Learning for Stunting Prevalence: A Novel Comprehensive Approach to Its Classification, Prediction, and Clustering Optimization in Aceh, Indonesia," *Informatics*, vol. 11, no. 4, 2024, doi: 10.3390/informatics11040089.

[10] A. Yogianto, A. Homaidi, and Z. Fatah, "Implementasi Metode K-Nearest Neighbors (KNN) untuk Klasifikasi Penyakit Jantung," *G-Tech J. Teknol. Terap.*, vol. 8, no. 1, pp. 186–195, 2024, doi: 10.33379/gtech.v8i3.4495.

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

[12] A. M. A. Rahim, I. Y. R. Pratiwi, and M. A. Fikri, "Klasifikasi Penyakit Jantung Menggunakan Metode Synthetic Minority Over-Sampling Technique Dan Random Forest Clasifier," *Indones. J. Comput. Sci.*, vol. 12, no. 5, pp. 2995–3011, 2023, doi: 10.33022/ijcs.v12i5.3413.

[13] N. Fajriati and B. Prasetiyo, "Optimasi Algoritma Naive Bayes dengan Diskritisasi K-Means pada Diagnosis Penyakit Jantung," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 3, pp. 503–512, 2023, doi: 10.25126/jtiik.2023106510.

[14] M. Y. Haffandi, E. Haerani, F. Syafria, and L. Oktavia, "Klasifikasi Penyakit Paru-Paru Dengan Menggunakan Metode Naïve Bayes Classifier," *J. Tek. Inf. dan Komput.*, vol. 5, no. 2, p. 176, 2022, doi: 10.37600/tekinkom.v5i2.649.

[15] N. Musriatun and H. Sujiliani, "Perbandingan Algoritma Machine Learning pada Klasifikasi Penyakit Jantung," *J. Infortech*, vol. 6, no. 1, pp. 46–51, 2024, doi: 10.31294/infortech.v6i1.21888.

[16] A. A. Surya and Y. Yamasari, "Penerapan Algoritma Naïve Bayes (NB) untuk Klasifikasi Penyakit Jantung," *J. Informatics Comput. Sci.*, vol. 5, no. 3, pp. 447–455, 2024, doi: 10.26740/jinacs.v5n03.p447-455.

[17] M. Lutfi, S. Surorejo, and P. Septiana, "Systematic Literature Review : Penerapan Algoritma Naives Bayes Dalam Sistem Pakar," *J. Minfo Polgan*, vol. 11, no. 2, pp. 7–13, 2022, doi: 10.33395/jmp.v11i2.11635.

[18] S. R. Azizah, R. Herteno, A. Farmadi, D. Kartini, and I. Budiman, "Kombinasi Seleksi Fitur Berbasis Filter dan Wrapper Menggunakan Naive Bayes pada Klasifikasi Penyakit Jantung," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 6, pp. 1361–1368, 2023, doi: 10.25126/jtiik.2023107467.

[19] I. F. Ashari and M. C. Untoro, "Hyperparameter Tuning Feature Selection with Genetic Algorithm and Gaussian Naïve Bayes for Diabetes Disease Prediction," *J. Telemat.*, vol. 17, no. 1, pp. 17–23, 2022, doi: 10.61769/telematika.v17i1.488.

[20] A. Nugroho and Y. Religia, "Analisis Optimasi Algoritma Klasifikasi Naive Bayes menggunakan Genetic Algorithm dan Bagging," *J. RESTI*, vol. 5, no. 3, pp. 504–510, 2021, doi: 10.29207/resti.v5i3.3067.

[21] R. Febryani and T. Arifin, "Optimasi Naïve Bayes Menggunakan Particle Swarm Optimization Untuk Tingkat Keberhasilan Cryotherapy Pada Penyakit Kutil," *J. Responsif Ris. Sains dan Inform.*, vol. 3, no. 2, pp. 174–183, 2021, doi: 10.51977/jti.v3i2.431.

[22] F. Fajrizal, T. Taslim, S. Handayani, D. Toresa, and L. Lisnawita, "Naïve Bayes Alpha Parameter Optimization with Ant Colony for Clinical Text Classification," *Digit. Zo. J. Teknol. Inf. dan Komun.*, vol. 16, no. 1, pp. 47–57, 2025, doi: 10.31849/rqfd9v12.

[23] A. K. Saputra, W. Susanty, and A. S. Kurniawan, "Optimasi Naïve Bayes Classifier Menggunakan Particle Swarm Optimization Pada Klasifikasi Data Penerima Bantuan Sosial," *Expert J. Manaj. Sist. Inf. dan Teknol.*, vol. 14, no. 2, p. 80, 2024, doi: 10.36448/expert.v14i2.3951.

[24] D. Weiskhy Steven, "Komparasi Algoritma Klasifikasi svm-pso Dan C4.5-pso dalam Prediksi Penyakit Jantung," *Inform. J. Inform. Manaj. dan Komput.*, vol. 13, no. 2, pp. 31–41, 2021, doi: 10.36723/juri.v13i2.301.

[25] S. Sumarna, S. Sartini, W. E. Pangesti, R. Suryadithia, and V. Riyanto, "Decision Tree Optimization in Heart Failure Diagnostics: a Particle Swarm Optimization Approach," *J. Tek. Inform.*, vol. 5, no. 3, pp. 739–746, 2024, doi: 10.52436/1.jutif.2024.5.3.1815.

[26] Y. Ayuningtyas and I. Made Suartana, "Klasifikasi Penyakit Stroke Menggunakan Support Vector Machine (SVM) dan Particle Swarm Optimization (PSO)," *JINACS (Journal Informatics Comput. Sci.*, vol. 04, no. 04, pp. 452–457, 2023, doi: 10.33884/jif.v9i02.3755.

[27] D. Susilowati, S. Sutrisno, and M. Yunus, "Penerapan Particle Swarm Optimization Untuk Meningkatkan Kinerja Algoritma K-Nearest Neighbor Dalam Klasifikasi Penyakit Diabetes," *J-REMI J. Rekam Med. dan Inf. Kesehat.*, vol. 4, no. 3, pp. 176–184, 2023, doi: 10.25047/j-remi.v4i3.3980.

[28] O. Pahlevi, A. Amrin, and Y. Handrianto, "Optimasi Algoritma Naïve Bayes Berbasis Particle Swarm Optimization Untuk Klasifikasi Status Stunting," *Comput. Sci.*, vol. 4, no. 1, pp. 37–43, 2024, doi: 10.31294/coscience.v4i1.2963.

[29] M. A. Wiratama and W. M. Pradnya, "Optimasi Algoritma Data Mining Menggunakan Backward Elimination untuk Klasifikasi Penyakit Diabetes," *J. Nas. Pendidik. Tek. Inform.*, vol. 11, no. 1, pp. 1–12, 2022, doi: 10.23887/janapati.v11i1.45282.

[30] Y. Miftahuddin and M. M. Faturrahman, "Penerapan Data Standardization dan Multilayer Perceptron pada Identifikasi Website Phishing," *MIND*, vol. 7, no. 2, pp. 111–123, 2022, doi: https://doi.org/10.26760/mindjournal.v7i2.111-123 |.

[31] D. B. Dinova and B. Prasetiyo, "Implementasi Random Forest dalam Klasifikasi Kanker Paru- Paru," *JOINTER*, vol. 05, no. 225, pp. 27–31, 2024, doi: 10.53682/jointer.v5i01.331.

[32] Y. Priantama and T. A. Y. Siswa, "Optimasi correlation-based feature selection untuk perbaikan akurasi random forest classifier dalam prediksi performa akademik mahasiswa," *JIKO*, vol. 6, no. 2, pp. 251–260, 2022, doi: 10.26798/jiko.v6i2.651.

[33] N. S. Sani, M. I. Esa, and B. A. Musawi, "SS symmetry Feature Selection," *Symmetry (Basel).*, vol. 15, no. 1, pp. 1–21, 2023, doi: 10.3390/sym15010123.

[34] S. Samasil, Y. Yuyun, and H. Hazriani, "Klasifikasi Mahasiswa Berpotensi Drop Out Menggunakan Algoritma Naive Bayes dan Decision Tree," *J. Ilm. Ilmu Komput.*, vol. 8, no. 2, pp. 108–114, 2022, doi: 10.35329/jiik.v8i2.242.

[35] R. R. R. Arisandi, B. Warsito, and A. R. Hakim, "Aplikasi naïve bayes classifier (nbc) pada klasifikasi status gizi balita stunting dengan pengujian k-fold cross validation 1,2,3," *J. GAUSSIAN*, vol. 11, no. 1, pp. 130–139, 2022, doi: 10.14710/j.gauss.v11i1.33991.

[36] R. N. Irawan, K. M. Hindrayani, and M. Idhom, "Penerapan Cross Validation sebagai Analisis Sentimen Pelayanan Publik Kereta Api Lokal Daop 8 Menggunakan Metode Multinomial Naïve Bayes," *G-Tech J. Teknol. Terap.*, vol. 8, no. 2, pp. 954–963, 2024, doi: 10.33379/gtech.v8i2.4117.

[37] H. Kurniawan, A. Rahim, T. Azhima, and Y. Siswa, "Implementasi Algoritma Gaussian Naïve Bayes Dalam Klasifikasi Status Gizi Pada Balita," *BITS*, vol. 6, no. 2, pp. 627–635, 2024, doi: 10.47065/ bits.v6i2.5493.

[38] M. A. Dwiputra, I. G. P. S. W. Wijaya, and R. Dwiyansaputra, "Perancangan Mesin Klasifikasi Menggunakan Particle Swarm Optimization," *J-COSINE (Journal Comput. Sci. Informatics Eng.*, vol. 8, no. 2, pp. 146–154, 2024, doi: 10.29303/jcosine.v8i2.614.

[39] N. Saurina, K. T. Aqila, A. S. Azizah, and U. Pudjianto, "Jurnal Informatika : Jurnal pengembangan IT Klasifikasi Pemanfaatan Ekstrak Bunga Sepatu Menggunakan Naïve Bayes dan Word Cloud," *J. Pengemb. IT*, vol. 11, no. 2, pp. 324–332, 2026, doi: 10.30591/jpit.v11i2.10045.

[40] F. Alghifari and D. Juardi, "Penerapan Data Mining Pada Penjualan Makanan Dan Minuman Menggunakan Metode Algoritma Naïve Bayes," *J. Ilm. Inform.*, vol. 9, no. 2, pp. 75–81, 2021, doi: 10.33884/jif.v9i02.3755.

[41] Pratama M. Duta, Gustriansyah Rendra, and Purnamasari Evi, "Klasifikasi Penyakit Daun Pisang Menggunakan Convolutional Neural Network (CNN)," *J. Teknol. Terpadu*, vol. 10, no. 1, pp. 1–6, 2024, doi: 10.54914/jtt.v10i1.1167.

[42] R. L. Atimi and E. E. Pratama, "Implementasi Model Klasifikasi Sentimen Pada Review Produk Lazada Indonesia," *J. Sains dan Inform.*, vol. 8, no. 1, pp. 88–96, 2022, doi: 10.34128/jsi.v8i1.419.

[43] C. A. Salsabila, F. Yulianto, and T. A. Y. Siswa, "Implementasi Metode Naive Bayes Untuk Klasifikasi Kecelakaan Lalu Lintas Di Kota Samarinda," *J. Inform. dan Tek. Elektro Terap.*, vol. 13, no. 1, 2025, doi: 10.23960/jitet.v13i1.5890.