Jurnal Informatika: Jurnal pengembangan IT

Vol. 11, No. 2, 2026
ISSN : 2477-5126, DOI:10.30591/jpit.v11i2.10241

Model Hybrid PSO-SMOTE, Correlation Feature Selection dan Random Forest untuk Deteksi Penyakit Jantung

Nur Dila Yuanti [1], Taghfirul Azhima Yoga Siswa [2] ,Wawan Joko Pranoto [3]
Program Studi Teknik Informatika, Universitas Muhammadiyah Kalimantan Timur, Jl. Ir. H. Juanda No.15, Sidodadi, Kec. Samarinda Ulu, Kota Samarinda, Kalimantan Timur 75124, Indonesia
nrdilay21@gmail.com, tay758@umkt.ac.id, wjp337@umkt.ac.id

Info Artikel

Riwayat Artikel:

Received 2026-01-25
Revised 2026-07-25
Accepted 2026-08-13

Abstract *– Heart disease is one of the leading causes of death worldwide, with approximately 17.8 million deaths reported in 2021. In Indonesia, the number of cases reached an estimated 15.5 million in 2022, highlighting the need for accurate early detection. This study aims to improve heart disease classification by integrating Correlation Feature Selection (CFS), Synthetic Minority Oversampling Technique (SMOTE), Particle Swarm Optimization (PSO), and Random Forest. CFS was applied to remove less relevant features before classification, SMOTE addressed class imbalance, and PSO optimized the Random Forest hyperparameters. The dataset consisted of 1,025 records from four heart disease datasets, which were cleaned into 302 unique instances. CFS eliminated the fasting blood sugar (fbs) attribute due to its very weak correlation with the target variable. Evaluation using 10-fold cross-validation showed that the baseline Random Forest achieved an accuracy of 82.80%, precision of 83.62%, recall of 86.07%, and f1-score of 84.46%, while Random Forest–PSO with SMOTE produced the best performance, achieving 85.11% accuracy, 84.62% precision, 89.74% recall, and 86.77% f1-score. These findings indicate that PSO provides the greatest performance improvement, while CFS simplifies the feature set before classification. The proposed hybrid model contributes by integrating feature selection, data balancing, and hyperparameter optimization into a single classification framework for more effective heart disease prediction.*

Keywords: *Correlation, Heart Disease, Particle Swarm Optimization, Random Forest, SMOTE.*

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

Abstrak **–** Penyakit jantung merupakan salah satu penyebab utama kematian di dunia, dengan sekitar 17,8 juta kematian dilaporkan pada tahun 2021. Di Indonesia, jumlah kasus diperkirakan mencapai 15,5 juta pada tahun 2022, sehingga diperlukan metode deteksi dini yang akurat. Penelitian ini bertujuan meningkatkan klasifikasi penyakit jantung dengan mengintegrasikan *Correlation Feature Selection (CFS), Synthetic Minority Oversampling Technique (SMOTE), Particle Swarm Optimization (PSO),* dan *Random Forest*. *CFS* diterapkan untuk menghilangkan fitur yang kurang relevan sebelum proses klasifikasi, *SMOTE* digunakan untuk mengatasi ketidakseimbangan kelas, sedangkan *PSO* digunakan untuk mengoptimalkan hiperparameter *Random Forest*. Dataset terdiri atas 1.025 data dari empat dataset penyakit jantung yang setelah proses pembersihan menghasilkan 302 data unik. Hasil seleksi fitur menggunakan *CFS* menghilangkan atribut *fasting blood sugar (fbs)* karena memiliki korelasi yang sangat lemah terhadap variabel target. Evaluasi menggunakan *10-Fold Cross Validation* menunjukkan bahwa Random Forest menghasilkan akurasi 82,80%, *precision* 83,62%, *recall* 86,07%, dan *f1-score* 84,46%, sedangkan kombinasi *Random Forest–PSO* dengan *SMOTE* memberikan performa terbaik dengan akurasi 85,11%, *precision* 84,62%, *recall* 89,74%, dan *f1-score* 86,77%. Hasil penelitian menunjukkan bahwa *PSO* memberikan peningkatan performa terbesar, sedangkan *CFS* menyederhanakan himpunan fitur sebelum proses klasifikasi. Model *hybrid* yang diusulkan memberikan kontribusi melalui integrasi seleksi fitur, penyeimbangan data, dan optimasi hiperparameter dalam satu alur klasifikasi sehingga menghasilkan prediksi penyakit jantung yang lebih efektif.

Kata Kunci*: Correlation,* Penyakit Jantung, *Particle Swarm Optimization, Random Forest, SMOTE.*

I. PENDAHULUAN

Penyakit jantung adalah salah satu masalah kesehatan paling serius di Indonesia, dengan jumlah kasus yang diperkirakan mencapai 15,5 juta pada tahun 2022. Menurut data dari Riset Kesehatan (Riskesdas), prevalensi meningkat dari 0,5% pada tahun 2013 menjadi 1,5% pada tahun 2018. Dari tahun 2000 hingga 2016, prevalensi penyakit jantung pada orang di bawah usia 40 tahun juga terus meningkat dengan laju sekitar 2% per tahun. Secara global, *Institute for Health Metrics and Evaluation (IHME)* melaporkan 9,14 juta kematian akibat penyakit jantung pada tahun 2019, sementara *World Health Organization (WHO)* melaporkan bahwa pada tahun 2021, penyakit ini menyebabkan 17,8 juta kematian per tahun, atau tiga kali lipat dari total semua kematian di seluruh dunia, dengan 21,2 juta penderita, mayoritasnya laki-laki [1]. Peningkatan prevalensi penyakit jantung menunjukkan perlunya pengembangan sistem diagnosis dan prediksi yang lebih akurat [2].

Pada kasus data medis khususnya penyakit jantung, *machine learning* telah banyak dimanfaatkan sebagai pendekatan analisis [3]. *Machine learning* adalah cabang kecerdasan buatan yang memungkinkan sistem komputer untuk belajar dari data historis, menganalisis data, dan membuat prediksi tanpa perlu diprogram secara eksplisit untuk setiap tugas [4]. Beberapa algoritma yang pernah digunakan dalam penelitian di bidang data medis penyakit jantung di antaranya *C4.5*, *K-Nearest Neighbor (KNN)*, *Support Vector Machine (SVM)*, *Artificial Neural Network (ANN)*, dan *Random Forest* ([5] ,[6], [7], [8]).

Random Forest merupakan algoritma *ensemble learning* yang membangun banyak *decision tree* dari subset data secara acak dan menggabungkan hasil prediksinya untuk meningkatkan akurasi serta mengurangi risiko *overfitting* [9], [10]. Meskipun demikian, penerapannya pada klasifikasi penyakit jantung masih belum optimal dengan akurasi sebesar 78,8% [11]. Sebaliknya, pada berbagai klasifikasi data medis lainnya, *Random Forest* mampu mencapai akurasi 95% pada kasus stroke [12], 98% pada diabetes[13], dan 91,76% pada gagal jantung [14]. Perbedaan tersebut menunjukkan bahwa karakteristik data penyakit jantung memerlukan pendekatan yang lebih optimal. Salah satu pendekatan yang telah dilakukan adalah menggabungkan dataset *Cleveland, Hungarian, Switzerland, Long Beach,* dan *Statlog*, guna mengatasi perbedaan karakteristik data antar sumber (*inter-dataset discrepancy*). Pendekatan tersebut mampu menghasilkan akurasi 100% pada dataset tunggal dan 96% pada dataset gabungan sehingga meningkatkan kemampuan generalisasi model [15]. Namun, penelitian lain menunjukkan bahwa kualitas fitur juga memengaruhi performa klasifikasi sehingga diperlukan feature selection untuk menghilangkan fitur yang kurang relevan [16].

Dataset medis umumnya mengandung fitur yang kurang relevan sehingga diperlukan teknik *feature selection* untuk memilih atribut yang paling berkontribusi terhadap proses klasifikasi [17]. Salah satu metode yang banyak digunakan adalah *Correlation Feature Selection (CFS)*, yang memilih fitur berdasarkan korelasinya terhadap kelas dan hubungan antarfitur [18]. Pada penelitian ini, *CFS* mengidentifikasi atribut *fasting blood sugar (fbs)* sebagai fitur dengan nilai korelasi terendah terhadap variabel target (−0,026826), sehingga dieliminasi dari proses klasifikasi. Sementara itu, atribut seperti *chest pain (cp), thalach*, *exang*, *oldpeak*, dan *ca* memiliki korelasi yang lebih tinggi sehingga dipertahankan sebagai fitur masukan model.

Selain pemilihan fitur, ketidakseimbangan distribusi kelas juga menjadi salah satu faktor yang memengaruhi performa klasifikasi penyakit jantung. Salah satu pendekatan yang banyak digunakan adalah *Synthetic Minority Oversampling Technique (SMOTE)*, yang mampu menyeimbangkan distribusi kelas dengan membangkitkan sampel sintetis pada kelas minoritas [19]. Penelitian sebelumnya menunjukkan bahwa penerapan *SMOTE* pada algoritma *K-Nearest Neighbor (KNN)* mampu meningkatkan akurasi klasifikasi hingga mencapai 90% [11]. Permasalahan ketidakseimbangan data tersebut umumnya diatasi melalui teknik *data balancing*, baik dengan pendekatan *oversampling* maupun *undersampling*, sehingga distribusi kelas menjadi lebih seimbang dan hasil klasifikasi menjadi lebih optimal [18], [19].

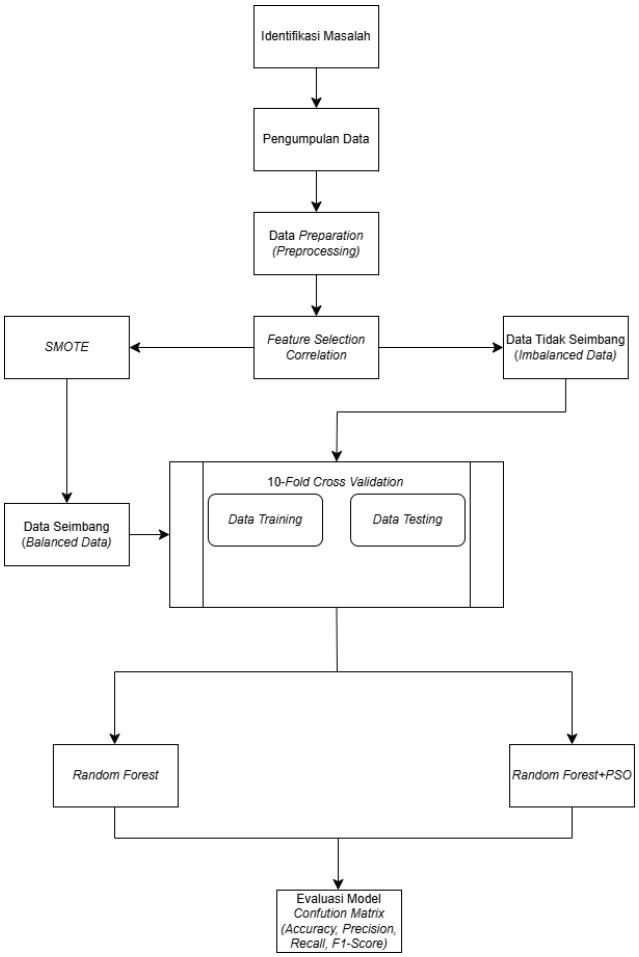
Efektivitas *SMOTE* juga telah dibuktikan pada berbagai algoritma klasifikasi, termasuk *Random Forest*. Penerapan *SMOTE* yang dikombinasikan dengan *Random Forest* terbukti mampu meningkatkan kinerja klasifikasi pada berbagai kasus. Pada data stunting di Kota Samarinda, kombinasi tersebut meningkatkan akurasi dari 98,83% menjadi 99,77% [21]. Penelitian lain juga menunjukkan bahwa penerapan *Particle Swarm Optimization-Random Forest* (*PSO-RF*) dengan dukungan *SMOTE* pada tahap prapemrosesan mampu menghasilkan akurasi hingga 99,7% pada beberapa dataset penyakit kronis, termasuk penyakit jantung, diabetes, dan kanker [22]. Temuan tersebut menunjukkan bahwa penanganan ketidakseimbangan data menggunakan *SMOTE* berkontribusi terhadap peningkatan performa model. Meskipun demikian, penanganan ketidakseimbangan data saja belum cukup karena performa *Random Forest* juga dipengaruhi oleh pemilihan hiperparameter. Oleh karena itu, diperlukan metode optimasi hiperparameter untuk memperoleh kombinasi parameter terbaik sehingga performa klasifikasi dapat ditingkatkan.

Salah satu metode optimasi hiperparameter yang banyak digunakan adalah *Particle Swarm Optimization* (*PSO*), yaitu algoritma optimasi berbasis populasi yang meniru perilaku kawanan burung atau ikan dalam mencari solusi terbaik secara iteratif sehingga mampu memperoleh kombinasi hiperparameter yang optimal [23]. Penelitian sebelumnya menunjukkan bahwa optimasi *Random Forest* menggunakan *PSO* pada klasifikasi penyakit jantung mampu menghasilkan akurasi sebesar 88,52%, *precision* 86,11%, *recall* 91,18%, *f1-score* 90,14%, serta nilai *AUC-ROC* sebesar 93,40% [24]. Hasil tersebut menunjukkan bahwa optimasi hiperparameter menggunakan *PSO* mampu meningkatkan kemampuan prediksi *Random Forest* sehingga layak dipadukan dengan teknik *feature selection* dan *data balancing* dalam klasifikasi penyakit jantung.

Meskipun berbagai penelitian telah menunjukkan bahwa *CFS*, *SMOTE*, maupun *PSO* mampu meningkatkan performa klasifikasi, sebagian besar masih menerapkan metode-metode tersebut secara terpisah atau hanya mengombinasikan satu hingga dua pendekatan. Penggunaan *CFS*, *SMOTE*, *PSO*, dan *Random Forest* secara terpadu pada klasifikasi penyakit jantung masih jarang diterapkan. Selain itu, penelitian terdahulu umumnya hanya berfokus pada peningkatan performa model akhir, sehingga pengaruh setiap tahapan, mulai dari seleksi fitur, penyeimbangan data, hingga optimasi hiperparameter terhadap hasil klasifikasi belum banyak dibahas.

Penelitian ini menggunakan kombinasi *Correlation Feature Selection (CFS), Synthetic Minority Oversampling Technique (SMOTE)*, *Particle Swarm Optimization (PSO),* dan *Random Forest* dalam satu alur klasifikasi penyakit jantung. Evaluasi dilakukan melalui empat skenario eksperimen, yaitu *Random Forest*, *Random Forest–SMOTE*, *Random Forest–PSO*, dan *Random Forest–PSO* dengan *SMOTE* setelah penerapan *CFS*, untuk menganalisis kontribusi setiap tahapan terhadap performa model. Penelitian ini bertujuan menghasilkan model klasifikasi penyakit jantung yang lebih akurat melalui peningkatan nilai *accuracy*, *precision*, *recall*, dan *f1-score*.

II. METODE



Gambar 1. Prosedur Penelitian

Gambar 1 Prosedur penelitian diawali dengan identifikasi masalah dan pengumpulan data dari *dataset* penyakit jantung. Selanjutnya dilakukan tahap *data preparation* (*preprocessing*) untuk menyiapkan data sebelum proses klasifikasi. Pada tahap ini diterapkan *Correlation Feature Selection* (CFS) untuk memilih atribut yang paling relevan serta menghilangkan fitur yang kurang berkontribusi terhadap proses klasifikasi. Setelah itu, ketidakseimbangan distribusi kelas ditangani menggunakan metode *Synthetic Minority Oversampling Technique* (SMOTE) sehingga diperoleh distribusi data yang lebih seimbang. Selanjutnya, data dibagi menggunakan metode *10-Fold Cross Validation* menjadi data *training* dan data *testing*. Data *training* digunakan untuk membangun dua model, yaitu *Random Forest* standar dan *Random Forest* yang dioptimasi menggunakan *Particle Swarm Optimization* (PSO). Pada tahap akhir, kinerja kedua model dievaluasi menggunakan *Confusion Matrix* dengan metrik *accuracy*, *precision, recall, dan f1-score.*

A. Pengumpulan Data

Penelitian ini menggunakan data medis yang diperoleh dari empat sumber, yaitu *Cleveland*, *Hungary*, *Switzerland*, dan *Long Beach V*, yang tersedia melalui situs *Kaggle* dan *Zenodo*. Dataset tersebut terdiri atas 1.025 data pasien dengan 14 atribut medis yang berkaitan dengan kondisi kesehatan jantung.

TABEL 1
ATRIBUT DATA PENYAKIT JANTUNG

No	Atribut	Tipe Data	Penjelasan
1	*Age*	*Numeric*	Umur pasien
2	*Sex*	*Category*	Jenis kelamin pasien (1 = Laki-laki, 0 = Perempuan)
3	*Cp*	*Category*	Tipe nyeri dada (0 = *Asymptomatic* yaitu tanpa gejala, 1 = *Atypical Angina* yaitu nyeri dengan gejala yang tidak khas, 2 = *Non-Anginal* yaitu bukan jantung, 3 = *Typical Angina* yaitu nyeri penyakit jantung)
4	*RestBP*	*Numeric*	Tekanan darah pasien ketika dalam keadaan istirahat (mmHg)
5	*Chol*	*Numeric*	Kadar kolesterol dalam darah pasien (mg/dl)
6	*Fbs*	*Category*	Kadar Gula darah pasien > 120 mg/dl (1 = Ya, 0 = Tidak)
7	*Restecg*	*Category*	Hasil rekam jantung istirahat (0 = Normal, 1 = *ST-T Abnormal* yaitu kondisi ketika terjadi inversi gelombang T dan/atau segmen ST yang mengalami peningkatan maupun penurunan lebih dari 0,5 mV, 2 = *LVH* yaitu keadaan dimana ventricular kiri mengalami hipertropi)
8	*Thalach*	*Numeric*	Detak jantung maksimum tercatat
9	*Exang*	*Category*	Keadaan dimana pasien akan mengalami nyeri dada apabila berolah raga (1 = Ya, 0 = Tidak)
10	*Oldpeak*	*Numeric*	Penurunan ST akibat olahraga (mm)
11	*Slope*	*Category*	Kemiringan segmen *ST* (0 = *Upsloping* yaitu garis *ST* menanjak ke atas, biasanya normal), 1 = *Flat* yaitu garis *ST* datar, bisa menjadi tanda awal adanya masalah, 2 = *Downsloping* yaitu garis *ST* menurun, sering dikaitkan dengan penyakit jantung).
12	*Ca*	*Category*	Jumlah pembuluh darah utama yang terlihat pada *fluoroskopi* (0–3)
13	*Thal*	*Category*	Kelainan darah *Thalassemia* (1 = Normal, 2 = *Fixed Defect* yaitu ada kerusakan permanen pada aliran darah., 3 = *Reversible Defect* yaitu ada kelainan, tetapi sifatnya bisa pulih atau berubah.)
14	*Target*	*Category*	Diagnosis (1 = Penyakit Jantung, 0 = Normal)

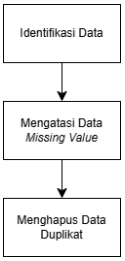
Tabel 1 menampilkan 13 atribut dan 1 label yang digunakan dalam penelitian terkait data penyakit jantung. Atribut-atribut tersebut mencakup data demografis, hasil pemeriksaan medis, hingga kondisi klinis pasien. Atribut yang digunakan meliputi usia, jenis kelamin, tekanan darah, kadar kolesterol, gula darah puasa, hasil elektrokardiogram (*EKG*), denyut jantung maksimum, serta kondisi angina saat berolahraga. Selain itu, dataset juga mencakup atribut depresi segmen ST, kemiringan segmen ST, jumlah pembuluh darah besar yang terdeteksi, kondisi thalassemia, serta label target yang menunjukkan status diagnosis penyakit jantung. Atribut-atribut tersebut digunakan sebagai faktor yang memengaruhi penyakit jantung dalam penelitian ini,

B. Data Pre-Processing

Pada tahap pra-pemrosesan, dataset yang terdiri atas 1.025 data dengan 14 atribut, yaitu 526 data positif dan 499 data negatif, dipersiapkan agar sesuai untuk proses deteksi penyakit jantung. Untuk mengurangi bias akibat ketidakseimbangan kelas, diterapkan *Synthetic Minority Oversampling Technique* (*SMOTE*). Tahapan pra-pemrosesan meliputi *data selection*, *data cleaning*, dan *data transformation* melalui standarisasi, diikuti dengan proses *balancing* menggunakan *SMOTE*. Selanjutnya, diterapkan *10-Fold Cross Validation* untuk memastikan stabilitas dan keandalan model sebelum data digunakan pada pemodelan *Random Forest* yang dioptimasi dengan *Particle Swarm Optimization* (*PSO*).

1) Data Cleaning

Langkah awal penelitian adalah pembersihan data (*data cleaning*) yang mencakup penanganan nilai hilang (*missing value*) dan penghapusan data duplikat agar data layak digunakan dalam analisis [25].



Gambar 2. Diagram Data Cleaning

Gambar 2 menunjukkan proses *data cleaning* yang meliputi identifikasi data, penanganan nilai yang hilang (*missing value*), penghapusan data duplikat.

2) Data Standarisasi

Pada tahap ini digunakan *Standard Scaler* untuk menormalkan fitur agar memiliki nilai rata-rata 0 dan simpangan baku 1, sehingga skala antar atribut menjadi seimbang dan mengurangi potensi bias pada model. [26]. Persamaan *standardization* [27].

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

Keterangan:

x : Data yang diinput.
μ : Rata-rata data.
σ : Standar deviasi data.

3) Feature Selection Correlation

Analisis korelasi dilakukan menggunakan *heatmap* untuk melihat hubungan antar atribut dengan nilai korelasi antara −1 hingga 1, sehingga membantu mengidentifikasi fitur yang relevan agar model lebih efisien dan akurat. [28]. Persamaan *correlation* [29].

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

Keterangan:

$\mathrm{Corr}_{x,y}$: 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).

4) Data Balancing dengan SMOTE

Tahap ini menerapkan *Synthetic Minority Oversampling Technique* (*SMOTE*) untuk menyeimbangkan distribusi kelas agar model tidak bias terhadap kelas mayoritas dan mampu meningkatkan kinerja klasifikasi pada kelas minoritas [30].

$$x_{new} = x_i + \lambda \cdot (x_k - x_i) \quad (3)$$

Keterangan:

x_i : Vektor sampel minoritas asli
x_k : Salah satu dari *k*-tetangga terdekat (minoritas) dari x_i
λ : Nilai acak di interval [0,1]
x_{new} : Sampel sintetik yang dibangkitkan di antara x_i dan x_k secara linier

C. 10-Fold Cross Validation

Dataset dibagi menjadi data latih dan data uji, di mana data latih digunakan untuk membangun model dan data uji untuk mengevaluasi kinerjanya. Pada penelitian ini digunakan metode *10-fold cross-validation* dengan membagi data latih ke dalam sepuluh bagian yang digunakan secara bergantian untuk proses pelatihan dan validasi [31]. Pemilihan nilai k = 10 didasarkan pada studi sebelumnya yang mengevaluasi beberapa nilai *k*, yaitu k = 3, 5, 7, 10, 15, *dan* 20, serta menyimpulkan bahwa k = 10 merupakan nilai yang umum digunakan dan mampu memberikan estimasi performa yang stabil serta mengurangi risiko *overfitting* [32].

TABEL 2
ILUSTRASI PROSES K-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 menjelaskan proses *10-Fold Cross Validation*. Dataset dibagi menjadi beberapa bagian dengan ukuran yang sama, di mana setiap batang atau kolom mewakili satu lipatan (K=10). Sebagai aturan umum, satu lipatan digunakan sebagai data uji (ditandai dengan peringatan abu-abu), sementara lipatan lainnya digunakan sebagai data latih. Proses ini akan diulang sebanyak 10 kali, dan hasil evaluasi dari setiap langkah dibandingkan untuk mendapatkan gambaran kinerja model yang lebih akurat.

D. Random Forest

Random Forest adalah jenis pohon keputusan yang dibangun dari sampel aritmatika tetapi memiliki kemampuan untuk mengenali simpul yang berbeda. Model ini bekerja dengan menggunakan subset fitur tertentu di setiap titik, kemudian menentukan batas terbaik untuk digunakan saat menganalisis data. *Random Forest* dipilih karena memiliki kemampuan menangani data berdimensi tinggi, lebih tahan terhadap overfitting dibanding decision tree tunggal, serta mampu memberikan performa yang baik pada data medis dengan karakteristik fitur yang beragam [33].

$$Entropy\ (Y) = -\Sigma_i p\ (c|Y) log^2 p(c|Y) \quad (4)$$

Keterangan:
Y : Himpunan Kasus
P(c|Y) : Proporsi nilai Y terhadap kelas c.

Kemudian menghitung *Information Gain*(Y,a), untuk menentukan atribut yang paling berpengaruh dalam proses klasifikasi.

$$Gain(Y) = -\sum_v evalues(a) \frac{|Y_v|}{|Y_a|} Entropy\ (Y_v) \quad (5)$$

Keterangan:
Values(a) : Nilai yang mungkin dalam himpunan kasus a.
Y_v : Subkelas dari Y dengan kelas v yang berhubungan dengan kelas a.
Y_a : Semua nilai yang sesuai dengan a.

E. Random Forest+PSO

Particle Swarm Optimization (PSO) merupakan algoritma optimasi berbasis populasi yang terinspirasi dari perilaku kawanan burung atau ikan dalam mencari solusi terbaik secara iteratif [23]. Pada penelitian ini, *PSO* mengoptimasi hiperparameter *Random Forest*, yaitu *n_estimators* dan *max_depth*, menggunakan 15 partikel, 10 iterasi, *inertia weight* 0,7, serta nilai *c1* dan *c2* masing-masing sebesar 1,5. Konfigurasi tersebut dipilih untuk menjaga keseimbangan antara eksplorasi dan eksploitasi selama proses optimasi.

$$V_k^{(i+1)} = V_k^{(i)} + c_1 r_1 \left(pbest_{k,i} - X_k^{(i)}\right) + c_2 r_2 \left(gbest_i - X_k^{(i)}\right) \quad (6)$$

Keterangan:
$V_k^{(i+1)}$: Kecepatan partikel ke-*k* pada iterasi ke (i+1).
$V_k^{(i)}$: Kecepatan partikel-*k* pada iterasi ke-i.
$X_k^{(i)}$: Posisi Partikel-*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.
$r_2 r_2$: Bilangan acak dalam rentang 0-1 yang digunakan untuk memastikan pencarian tetap terjaga.

F. Evaluasi

Confusion Matrix adalah alat evaluasi yang digunakan sebagai alat evaluasi untuk membandingkan hasil prediksi model dengan label aktual. Melalui matriks ini, performa klasifikasi dapat dianalisis secara lebih rinci berdasarkan jumlah prediksi yang benar dan salah pada setiap kelas [34].

TABEL 3
CONFUSION MATRIX

	Prediksi Positif	**Prediksi Negatif**
Aktual Positif	*True Positive (TP)*	*False Negative (FN)*
Aktual Negatif	*False Positive (FP)*	*True Negative (TN)*

Tabel 3 *True Positive* (TP) menunjukkan jumlah data positif yang berhasil diklasifikasikan dengan benar, sedangkan *True Negative* (TN) merepresentasikan data negatif yang juga diklasifikasikan secara tepat. *False Positive* (FP) merujuk pada data negatif yang salah diklasifikasikan sebagai positif, sementara *False Negative* (FN) merupakan data positif yang keliru diklasifikasikan sebagai negatif. Nilai-nilai tersebut selanjutnya digunakan untuk menghitung metrik evaluasi seperti *accuracy*, *precision*, *recall*, dan *F1-score* guna menilai kekuatan serta kelemahan model dalam mengklasifikasikan setiap kelas.

1) Accuracy

Accuracy bisa diartikan sebagai tingkat ketepatan suatu model dalam mengklasifikasikan data, yakni dengan membandingkan jumlah prediksi yang benar terhadap keseluruhan data yang ada [35].

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

2) Precision

Precision adalah ukuran yang digunakan untuk melihat seberapa tepat sebuah model dalam memberikan prediksi positif, yaitu dengan menilai berapa banyak dari hasil positif yang diprediksi benar-benar sesuai dengan kondisi sebenarnya [36].

$$Precision = \frac{TP}{TP + FP} \times 100\% \quad (8)$$

3) Recall

Recall adalah ukuran yang digunakan untuk melihat seberapa baik sebuah model dalam menemukan data positif yang sebenarnya, yaitu dengan membandingkan jumlah data positif yang berhasil terdeteksi dengan total data positif yang ada [37].

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (9)$$

4) F1-Score

F1-Score merupakan metrik yang digunakan untuk menilai keseimbangan antara *precision* dan recall, dengan cara menghitung rata-rata harmonis dari keduanya guna menggambarkan kinerja keseluruhan model klasifikasi [5].

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (10)$$

III. HASIL DAN PEMBAHASAN

A. Hasil

Bagian ini menyajikan hasil evaluasi model klasifikasi penyakit jantung menggunakan kombinasi *Correlation Feature Selection (CFS)*, *Synthetic Minority Oversampling Technique (SMOTE*), *Particle Swarm Optimization (PSO)*, dan *Random Forest*. Pengujian dilakukan pada empat skenario eksperimen, yaitu *Random Forest*, *Random Forest–SMOTE*, *Random Forest–PSO*, dan *Random Forest–PSO* dengan *SMOTE* setelah penerapan *CFS*. Kinerja model dievaluasi berdasarkan nilai *accuracy, precision, recall,* dan *F1-score* untuk menganalisis pengaruh setiap tahapan terhadap performa klasifikasi.

TABEL 4
DATASET PENYAKIT JANTUNG

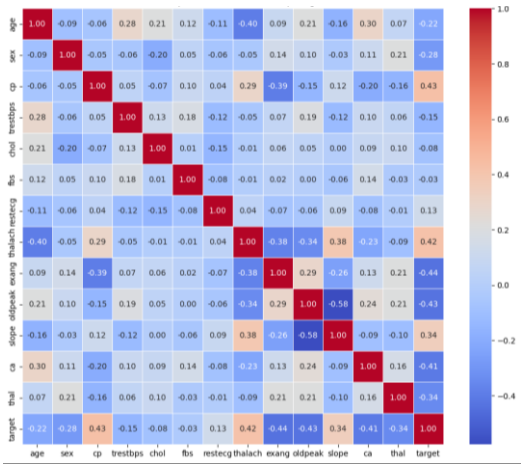
No	*Age*	*Sex*	*Cp*	*Trestbps*	*Chol*	*Fbs*	*Restecg*	*Thalach*	*Exang*	*Oldpeak*	*Slope*	*Ca*	*Thal*	*Target*
1	52	1	0	125	212	0	1	168	0	1	2	2	3	0
2	53	1	0	140	203	1	0	155	1	3,1	0	0	3	0
3	70	1	0	145	174	0	1	125	1	2,6	0	0	3	0
4	61	1	0	148	203	0	1	161	0	0	2	1	3	0
5	62	0	0	138	294	1	1	106	0	1,9	1	3	2	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1021	59	1	1	140	221	0	1	164	1	0	2	0	2	1
1022	60	1	0	125	258	0	0	141	1	2,8	1	1	3	0
1023	47	1	0	110	275	0	0	118	1	1	1	1	2	0
1024	50	0	0	110	254	0	0	159	0	0	2	0	2	1
1025	54	1	0	120	188	0	1	106	0	1,4	1	1	3	0

Tabel 4 menunjukkan dataset penyakit jantung yang digunakan dalam penelitian ini. Dataset terdiri atas 13 atribut sebagai variabel masukan, yaitu *Age*, *Sex*, *Chest Pain Type*, *TrestBPS*, *Cholesterol*, *FastingBS*, *Resting Electrocardiographic Results*, *Thalach*, *Exang*, *Oldpeak*, *Slope*, *Ca*, dan *Thalassemia*, serta satu atribut target yang menunjukkan hasil diagnosis penyakit jantung, dengan nilai 1 menyatakan pasien menderita penyakit jantung dan nilai 0 menyatakan pasien tidak menderita penyakit jantung. Sebelum proses klasifikasi dilakukan, dataset terlebih dahulu melalui tahap *preprocessing* untuk memastikan kualitas data.

TABEL 5
DATA CLEANING

Initial Dataset Size	Duplicate Records	Final Records After Cleaning
1025	723	302

Tabel 5 menyajikan hasil proses *deduplication* pada tahap *preprocessing*. Dari 1.025 data awal, sebanyak 723 data duplikat dieliminasi sehingga menghasilkan 302 data unik yang digunakan dalam proses klasifikasi. Selanjutnya, data distandardisasi menggunakan *StandardScaler* agar setiap atribut memiliki skala yang sama. Hasil standardisasi menghasilkan nilai positif dan negatif yang menunjukkan posisi data terhadap nilai rata-ratanya.



Gambar 3. *Correlation Heatmap*

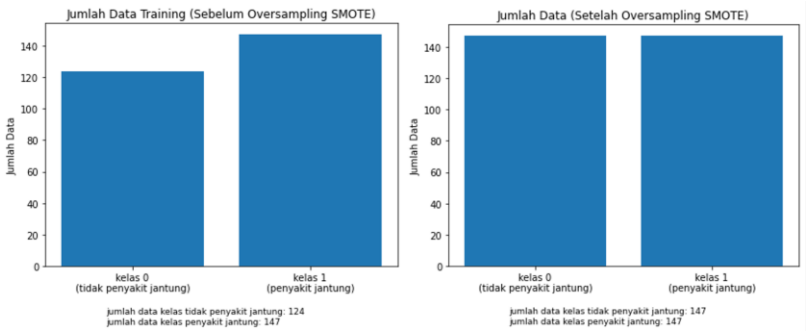
Gambar 3 menunjukkan hasil analisis *correlation heatmap* antaratribut terhadap variabel target. Atribut *cp* (0,432080) dan *thalach* (0,419955) memiliki korelasi positif tertinggi, sedangkan *exang* (-0,435601), *oldpeak* (-0,429146), dan *ca* (-0,408992) menunjukkan korelasi negatif yang relatif kuat. Sebaliknya, atribut *fbs* memiliki nilai korelasi terendah (-0,026826), sehingga dieliminasi pada tahap *Correlation Feature Selection (CFS)*, sementara atribut lainnya dipertahankan sebagai masukan pada proses klasifikasi.

TABEL 6
DATASET CORRELATION FEATURE SELECTION

No	*Age*	*Sex*	*Cp*	*Trestbps*	*Chol*	*Restecg*	*Thalach*	*Exang*	*Oldpeak*	*Slope*	*Ca*	*Thal*	*Target*
1	-0.267966	1	0	-0.376556	-0.667728	1	0.806035	0	-0.037124	2	2	3	0
2	-0.157260	1	0	0.478910	-0.841918	0	0.237495	1	1.773958	0	0	3	0
3	1.724733	1	0	0.764066	-1.403197	1	-1.074521	1	1.342748	0	0	3	0
4	0.728383	1	0	0.935159	-0.841918	1	0.499898	0	-0.899544	2	1	3	0
5	0.839089	0	0	0.364848	0.919336	1	-1.905464	0	0.739054	1	3	2	0

Tabel 6 menunjukkan pemilihan fitur dilakukan berdasarkan perbandingan nilai korelasi *Pearson* tanpa menggunakan *threshold* tertentu. Atribut *fasting blood sugar* (*fbs*) memiliki nilai korelasi paling rendah sehingga dieliminasi. Penghapusan atribut *fbs* menyederhanakan data dari 14 menjadi 13 atribut tanpa menurunkan performa model sehingga proses pelatihan menjadi lebih efisien. Selanjutnya, dataset yang terdiri atas 302 sampel dievaluasi menggunakan metode *10-Fold Cross Validation*. Karena jumlah sampel tidak dapat dibagi secara merata ke dalam sepuluh *fold*, jumlah data *training* berkisar antara 271–272 sampel, sedangkan data *testing* terdiri atas 30–31 sampel pada setiap *fold*.

Permasalahan ketidakseimbangan kelas ditangani menggunakan metode *Synthetic Minority Over-sampling Technique (SMOTE)*. Sebelum penerapan *SMOTE*, distribusi kelas belum seimbang, dengan rata-rata sekitar 124 data pada kelas 0 dan 147 data pada kelas 1.



Gambar 4. Visualisasi Sebelum dan Sesudah Menggunakan *SMOTE*

Gambar 4 menunjukkan perbandingan distribusi data *training* sebelum dan sesudah penerapan *SMOTE*, di mana sebelum *SMOTE* jumlah data pada kelas 0 (tidak penyakit jantung) dan kelas 1 (penyakit jantung) masing-masing sebesar 124 data dan 147 data, sedangkan setelah *SMOTE* jumlah data pada kedua kelas menjadi seimbang, yaitu 147 data pada setiap kelas. Hasil evaluasi *Random Forest* dari setiap *fold* kemudian dianalisis untuk memperoleh nilai kinerja model.

TABEL 7
MODEL RADOM FOREST

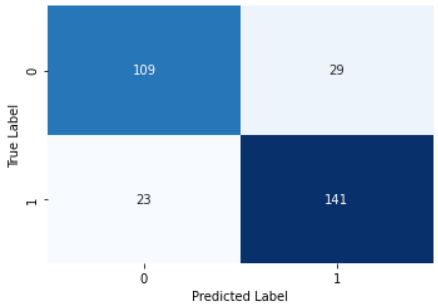
Fold	*Accuracy*	*Precision*	*Recall*	*F1-Score*
1	80.65%	86.67%	76.47%	81.25%
2	80.65%	86.67%	76.47%	81.25%
3	86.67%	87.50%	87.50%	87.50%
4	73.33%	68.18%	93.75%	78.95%
5	80.00%	77.78%	87.50%	82.35%
6	83.33%	86.67%	81.25%	83.87%
7	83.33%	86.67%	81.25%	83.87%
8	90.00%	84.21%	100.00%	91.43%
9	93.33%	94.12%	94.12%	94.12%
10	76.67%	77.78%	82.35%	80.00%

Tabel 7 menunjukkan hasil pemodelan algoritma *Random Forest* tanpa penerapan *PSO* dan tanpa teknik *SMOTE* menggunakan skema *10-fold cross validation*. Nilai *accuracy* pada setiap *fold* menunjukkan perbedaan hasil. *Accuracy* tertinggi diperoleh pada *fold* ke-9 sebesar 93,33%, sedangkan *accuracy* terendah terjadi pada *fold* ke-4 sebesar 73,33%. Nilai *precision*, *recall*, dan *f1-score* pada sebagian besar *fold* menunjukkan hasil yang cukup baik, meskipun masih terdapat perbedaan antar *fold*.

TABEL 8
HASIL RATA-RATA RANDOM FOREST

Rata-rata			
Accuracy	*Precision*	*Recall*	*F1-Score*
82.80%	83.62%	86.07%	84.46%

Tabel 8 menunjukkan nilai rata-rata kinerja model *Random Forest* tanpa *SMOTE* dari seluruh *fold* pengujian. Nilai yang diperoleh meliputi *accuracy* sebesar 82,80%, *precision* 83,62%, *recall* 86,07%, dan *F1-score* 84,46%.



Gambar 5. *Confusion Matrix* model *Random Forest*

Gambar 5 menunjukkan hasil *confusion matrix* dari model *Random Forest*. Berdasarkan visualisasi tersebut, model berhasil mengklasifikasikan 141 data positif dan 109 data negatif dengan benar. Selain itu, terdapat 29 data negatif yang salah diklasifikasikan sebagai positif serta 23 data positif yang diprediksi sebagai negatif. Nilai-nilai ini selanjutnya digunakan dalam perhitungan *accuracy, precision*, *recall*, *f1-score* dengan menggunakan rumus.

$$Accuracy = \frac{141 + 109}{141 + 109 + 29 + 23} \times 100\% = 82{,}78\%$$

$$Precision = \frac{141}{141 + 29} \times 100\% = 82{,}94\%$$

$$Recall = \frac{141}{141 + 23} \times 100\% = 85{,}98\%$$

$$F1 - Score = 2 \times \frac{0{,}8294 \times 0{,}8598}{0{,}8294 + 0{,}8598} \times 100\% = 84{,}43\%$$

Berdasarkan perhitungan *confusion matrix*, diperoleh nilai *accuracy* sebesar 82,78%, *precision* 82,94%, *recall* 85,98%, dan *f1-score* 84,43%.

TABEL 9
PARAMETER TERBAIK HASIL OPTIMASI *PSO*

Parameter	**Nilai Terbaik**
n_estimators	111
max_depth	3
Best F1-score	0.8635

Tabel 9 menunjukkan parameter terbaik yang diperoleh dari hasil optimasi menggunakan PSO. Berdasarkan hasil tersebut, PSO berhasil menemukan kombinasi *n_estimators* sebesar 111 dan *max_depth* sebesar 3, dengan nilai *f1-score* terbaik sebesar 0.8635, nilai ini dipilih karena *f1-score* paling menggambarkan keseimbangan antara *precision* dan *recall* pada data medis yang tidak seimbang. Hasil evaluasi dari masing-masing *fold* kemudian dianalisis untuk menentukan kinerja model.

TABEL 10
MODEL *FOLD RANDOM FOREST+PSO*

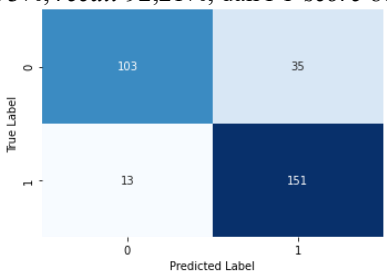
Fold	*Accuracy*	*Precision*	*Recall*	*F1-Score*
1	83,87%	87,50%	82,35%	84,85%
2	83,87%	87,50%	82,35%	84,85%
3	90,00%	84,21%	100,00%	91,43%
4	73,33%	66,67%	100,00%	80,00%
5	83,33%	78,95%	93,75%	85,71%
6	83,33%	86,67%	81,25%	83,87%
7	93,33%	88,89%	100,00%	94,12%
8	86,67%	80,00%	100,00%	88,89%
9	83,33%	80,00%	94,12%	86,49%
10	80,00%	78,95%	88,24%	83,33%

Tabel 10 menunjukkan hasil pengujian model *Random Forest* yang dioptimasi menggunakan *PSO*. Nilai *accuracy* tertinggi diperoleh pada *fold* ke-7 sebesar 93,33%, sedangkan *accuracy* terendah terjadi pada *fold* ke-4 sebesar 73,33%. Nilai *precision*, *recall*, dan *f1-score* menunjukkan hasil yang cukup baik pada sebagian besar *fold*. Nilai *recall* mencapai 100,00% pada beberapa *fold.*

TABEL 11
HASIL RATA-RATA *RANDOM FOREST+PSO*

Rata-rata			
Accuracy	*Precision*	*Recall*	*F1-Score*
84,11%	81,93%	92,21%	86,35%

Tabel 11 menunjukkan nilai rata-rata kinerja model *Random Forest–PSO* tanpa *SMOTE*. Nilai yang diperoleh meliputi *accuracy* 84,11%, *precision* 81,93%, *recall* 92,21%, dan *F1-score* 86,35%.



Gambar 6. *Confusion Matrix Random Forest+PSO*

Gambar 6 menampilkan *confusion matrix* dari model *Random Forest–PSO* tanpa penerapan *SMOTE*. Berdasarkan hasil tersebut, model berhasil mengklasifikasikan 151 data pasien penyakit jantung sebagai positif *(True Positive)* dan 103 data pasien sehat sebagai negatif *(True Negative)* dengan benar. Namun, masih terdapat 35 data pasien sehat yang salah diprediksi sebagai penyakit jantung *(False Positive)* serta 13 data pasien penyakit jantung yang keliru diprediksi sebagai sehat *(False Negative)*. Nilai-nilai tersebut selanjutnya digunakan sebagai dasar dalam perhitungan metrik evaluasi, yaitu *accuracy*, *precision*, *recall*, dan *f1-score* menggunakan rumus yang telah ditentukan.

$$Accuracy = \frac{151 + 103}{151 + 103 + 35 + 13} \times 100\% = 84{,}11\%$$

$$Precision = \frac{151}{151 + 35} \times 100\% = 81{,}18\%$$

$$Recall = \frac{151}{151 + 13} \times 100\% = 92{,}07\%$$

$$F1 - Score = 2 \times \frac{0{,}8118 \times 0{,}9207}{0{,}8118 + 0{,}9207} \times 100\% = 86{,}29\%$$

Berdasarkan hasil perhitungan *confusion matrix*, diperoleh nilai *accuracy* sebesar 84,11%, *precision* 81,18%, *recall* 92,07%, dan *f1-score* 86,29%.

TABEL 12
MODEL *RANDOM FOREST+ SMOTE*

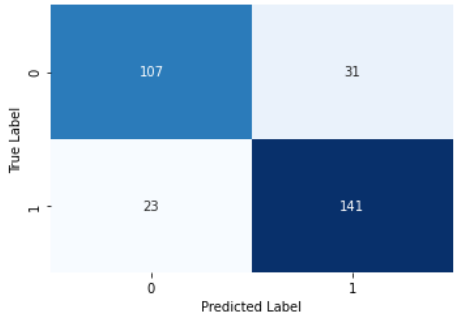
Fold	*Accuracy*	*Precision*	*Recall*	*F1-Score*
1	80.65%	86.67%	76.47%	81.25%
2	83.87%	92.86%	76.47%	83.87%
3	86.67%	87.50%	87.50%	87.50%
4	70.00%	65.22%	93.75%	76.92%
5	80.00%	77.78%	87.50%	82.35%
6	76.67%	80.00%	75.00%	77.42%
7	86.67%	87.50%	87.50%	87.50%
8	86.67%	83.33%	93.75%	88.24%
9	86.67%	84.21%	94.12%	88.89%
10	83.33%	83.33%	88.24%	85.71%

Tabel 12 menunjukkan hasil pemodelan *Random Forest* dengan penerapan *SMOTE*. Nilai *accuracy* tertinggi sebesar 86,67% dan terendah 70,00%. Nilai *precision*, *recall*, dan *f1-score* pada sebagian besar *fold* menunjukkan hasil yang cukup baik, meskipun terdapat perbedaan pada beberapa *fold*.

TABEL 13
HASIL RATA-RATA *RANDOM FOREST+SMOTE*

Rata-rata			
Accuracy	*Precision*	*Recall*	*F1-Score*
82.12%	82.84%	86.03%	83.97%

Tabel 13 menyajikan nilai rata-rata kinerja model *Random Forest* dengan penerapan *SMOTE* berdasarkan seluruh *fold* pengujian. Nilai yang diperoleh meliputi *accuracy* sebesar 82,12%, *precision* 82,84%, *recall* 86,03%, dan *F1-score* 83,97%.



Gambar 7. *Confusion Matrix Random Forest+SMOTE*

Gambar 7 menunjukkan *confusion matrix* dari model *Random Forest* dengan penerapan *SMOTE*. Berdasarkan hasil tersebut, model berhasil mengklasifikasikan 141 pasien penyakit jantung sebagai positif *(True Positive)* dan 107 pasien sehat sebagai negatif *(True Negative)* dengan benar. Namun, masih terdapat 31 pasien sehat yang salah diprediksi sebagai penyakit jantung *(False Positive)* serta 23 pasien penyakit jantung yang keliru diprediksi sebagai sehat *(False Negative)*. Nilai-nilai tersebut selanjutnya digunakan sebagai dasar dalam perhitungan metrik evaluasi, yaitu *accuracy*, *precision*, *recall*, dan *f1-score* sesuai dengan rumus yang telah ditetapkan.

$$Accuracy = \frac{141 + 107}{141 + 107 + 31 + 23} \times 100\% = 82{,}12\%$$

$$Precision = \frac{141}{141 + 31} \times 100\% = 81{,}98\%$$

$$Recall = \frac{141}{141 + 23} \times 100\% = 85{,}98\%$$

$$F1 - Score = 2 \times \frac{0{,}8198 \times 0{,}8598}{0{,}8198 + 0{,}8598} \times 100\% = 83{,}93\%$$

Berdasarkan perhitungan *confusion matrix*, diperoleh nilai *accuracy* 82,12%, *precision* 81,98%, *recall* 85,98%, dan *f1-score* 83,93%.

TABEL 14
HASIL RATA-RATA *RANDOM FOREST+PSO+SMOTE*

Parameter	Nilai Terbaik
n_estimators	218
max_depth	4
Best F1-score	0.8677

Tabel 14 menunjukkan parameter terbaik yang diperoleh dari hasil optimasi menggunakan PSO. Berdasarkan hasil tersebut, PSO berhasil menemukan kombinasi *n_estimators* sebesar 218 dan *max_depth* sebesar 4 dengan nilai *f1-score* terbaik sebesar 0.8677. Hasil evaluasi dari masing-masing *fold* kemudian dianalisis untuk menentukan kinerja model.

TABEL 15
MODEL *RANDOM FOREST+PSO+SMOTE*

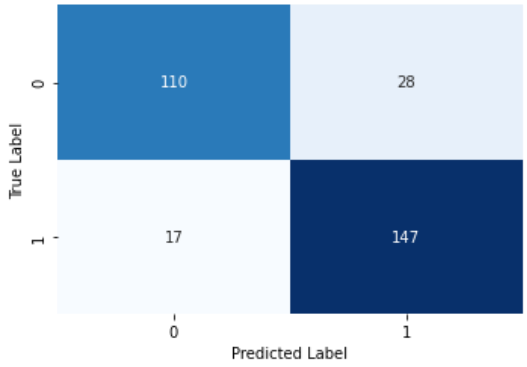
Fold	*Accuracy*	*Precision*	*Recall*	*F1-Score*
1	80,65%	86,67%	76,47%	81,25%
2	87,10%	93,33%	82,35%	87,50%
3	93,33%	88,89%	100,00%	94,12%
4	73,33%	68,18%	93,75%	78,95%
5	80,00%	77,78%	87,50%	82,35%
6	83,33%	86,67%	81,25%	83,87%
7	90,00%	88,24%	93,75%	90,91%
8	93,33%	88,89%	100,00%	94,12%
9	86,67%	84,21%	94,12%	88,89%
10	83,33%	83,33%	88,24%	85,71%

Tabel 15 menunjukkan hasil pemodelan *Random Forest–PSO* dengan penerapan *SMOTE* menggunakan sepuluh *fold* pengujian. Nilai *accuracy* tertinggi diperoleh sebesar 93,33% pada *fold* ke-3 dan ke-8, sedangkan *accuracy* terendah terjadi pada *fold* ke-4 sebesar 73,33%. Nilai *precision*, *recall*, dan *F1-score* pada sebagian besar *fold* menunjukkan hasil yang cukup baik, dengan beberapa *fold* mencapai nilai *recall* sebesar 100,00%.

TABEL 16
HASIL RATA-RATA DARI *RANDOM FOREST+PSO+SMOTE*

Rata-rata			
Accuracy	*Precision*	*Recall*	*F1-Score*
85,11%	84,62%	89,74%	86,77%

Tabel 16 menampilkan hasil rata-rata kinerja model *Random Forest–PSO* dengan penerapan *SMOTE* berdasarkan seluruh *fold* pengujian. Nilai yang diperoleh meliputi *accuracy* sebesar 85,11%, *precision* 84,62%, *recall* 89,74%, dan *F1-score* 86,77%.



Gambar 8. *Confusion Matrix Random Forest+PSO+SMOTE*

Gambar 8 menampilkan *confusion matrix* dari model *Random Forest–PSO* dengan penerapan *SMOTE*. Berdasarkan hasil tersebut, model berhasil mengklasifikasikan 147 pasien penyakit jantung sebagai positif *(True Positive)* dan 110 pasien sehat sebagai negatif *(True Negative)* dengan benar. Namun, masih terdapat 28 pasien sehat

yang salah diprediksi sebagai penyakit jantung *(False Positive)* serta 17 pasien penyakit jantung yang keliru diprediksi sebagai sehat *(False Negative)*. Nilai tersebut digunakan dalam perhitungan metrik evaluasi kinerja model.

$$Accuracy = \frac{147 + 110}{147 + 110 + 28 + 17} \times 100\% = 85{,}10\%$$

$$Precision = \frac{147}{147 + 28} \times 100\% = 84{,}00\%$$

$$Recall = \frac{147}{147 + 17} \times 100\% = 89{,}63\%$$

$$F1 - Score = 2 \times \frac{0{,}8400 \times 0{,}8963}{0{,}8400 + 0{,}8963} \times 100\% = 86{,}73\%$$

Perhitungan *confusion matrix* menghasilkan nilai *accuracy* 85,10%, *precision* 84,00%, *recall* 89,63%, dan *f1-score* 86,73%.

TABEL 17
HASIL PERBANDINGAN NILAI AKURASI RATA-RATA MODEL SEBELUM DITERAPKANNYA METODE SMOTE

Average Accuracy	*RF*	*RF+PSO*	Perubahan *RF* Ke *RF+PSO*
Rata-rata	82,80%	84,11%	+1,31%

Tabel 17 menunjukkan perbandingan nilai rata-rata *accuracy* antara model *Random Forest* dan *Random Forest–PSO* sebelum penerapan *SMOTE*. Hasil menunjukkan bahwa optimasi *PSO* meningkatkan nilai *accuracy* dari 82,80% menjadi 84,11%, dengan peningkatan sebesar 1,31%.

TABEL 18
PERBANDINGAN NILAI AKURASI RATA-RATA MODEL SESUDAH DITERAPKANNYA METODE SMOTE

Average Accuracy	*RF*	*RF+PSO*	Perubahan *RF* Ke *RF+PSO*
Rata-rata	82,12%	85,11%	+2,99%

Tabel 18 menunjukkan bahwa setelah penerapan *SMOTE*, nilai rata-rata *accuracy* model *Random Forest–PSO* meningkat dari 82,12% menjadi 85,11%, atau mengalami peningkatan sebesar 2,99% dibandingkan model *Random Forest*.

TABEL 19
PERBANDINGAN NILAI RATA-RATA MODEL RANDOM FOREST SEBELUM DITERAPKAN SMOTE

Average Accuracy	*RF*	*RF+SMOTE*	Perubahan *RF* Ke *RF+SMOTE*
Rata-rata	82,80%	82,12%	−0,82%

Tabel 19 menunjukkan bahwa penerapan *SMOTE* menurunkan nilai rata-rata *accuracy* model *Random Forest* dari 82,80% menjadi 82,12%, atau turun sebesar 0,82%.

TABEL 20
PERBANDINGAN NILAI RATA-RATA MODEL RANDOM FOREST-PSO SESUDAH DITERAPKAN SMOTE

Average Accuracy	*RF+PSO*	*RF+PSO+SMOTE*	Perubahan *RF+PSO* Ke *RF+PSO+SMOTE*
Rata-rata	84,11%	85,11%	+1,00%

Tabel 20 menunjukkan bahwa penerapan *SMOTE* pada model *Random Forest–PSO* meningkatkan nilai rata-rata *accuracy* dari 84,11% menjadi 85,11%, atau mengalami peningkatan sebesar 1,00%.

B. Pembahasan

Dataset yang digunakan pada penelitian ini merupakan data penyakit jantung yang dikompilasi dari empat sumber medis, yaitu *Cleveland*, *Hungary*, *Switzerland*, dan *Long Beach V Cleveland*, yang diperoleh melalui *platform Kaggle* dan *Zenodo*. Dataset awal terdiri dari 1.025 data pasien dengan total 14 atribut, yang mencakup 13 atribut sebagai variabel input dan satu atribut target sebagai label klasifikasi untuk menentukan kondisi pasien, yaitu menderita penyakit jantung atau tidak. Setelah dilakukan proses pembersihan data dan penghapusan duplikasi, jumlah data yang digunakan dalam tahap pemodelan berkurang menjadi 302 data. Permasalahan ketidakseimbangan kelas ditangani menggunakan metode *Synthetic Minority Over-sampling Technique (SMOTE).* Sebelum penerapan *SMOTE*,

distribusi kelas belum seimbang, dengan rata-rata sekitar 124 data pada kelas 0 dan 147 data pada kelas 1 pada setiap *fold*. Setelah *SMOTE* diterapkan, distribusi data antar kelas menjadi lebih seimbang dengan jumlah data berkisar antara 294 hingga 296 data pada setiap *fold*.

Evaluasi model dilakukan dengan *10-fold cross-validation*, di mana *SMOTE* hanya diterapkan pada data latih (*training set*) di setiap *fold*, sementara data uji tetap menggunakan data asli. Dengan demikian, jumlah total data asli tetap 302 baris. Proses evaluasi, termasuk pembentukan *confusion matrix*, dilakukan menggunakan data uji asli agar hasil pengujian valid dan bebas dari *data leakage*. Pendekatan ini memastikan model dievaluasi berdasarkan data yang merepresentasikan kondisi sebenarnya tanpa pengaruh data sintesis dari *SMOTE*. Penelitian ini menguji empat skenario, yaitu *Random Forest*, *Random Forest+SMOTE*, *Random Forest+PSO*, dan *Random Forest+PSO+SMOTE*.

TABEL 21
PERBANDINGAN MODEL

No	Model	*Accuracy*	*Precision*	*Recall*	*F1-Score*
1	*Random Forest*	82.80%	83.62%	86.07%	84.46%
2	*Random Forest+SMOTE*	82.12%	82.84%	86.03%	83.97%
3	*Random Forest+PSO*	84,11%	81,93%	92,21%	86,35%
4	*Random Forest+PSO+SMOTE*	85,11%	84,62%	89,74%	86,77%

Tabel 21 menampilkan perbandingan kinerja rata-rata dari empat skenario model yang diuji. Model *Random Forest* tanpa penerapan *SMOTE* menghasilkan nilai *accuracy* sebesar 82,80%, *precision* 83,62%, *recall* 86,07%, dan *f1-score* 84,46%. Setelah diterapkan *SMOTE*, nilai *accuracy* sedikit menurun menjadi 82,12% dan *f1-score* menjadi 83,97%, yang menunjukkan bahwa penerapan *SMOTE* tidak selalu menghasilkan peningkatan kinerja. Hal ini sejalan dengan penelitian sebelumnya yang melaporkan bahwa penggunaan *SMOTE* dapat menyebabkan penurunan nilai *accuracy*, dari 87,18% menjadi 70,55%, terutama ketika tingkat ketidakseimbangan kelas tidak bersifat ekstrem [38]. Kondisi tersebut menunjukkan bahwa *SMOTE* belum memberikan peningkatan yang signifikan pada dataset dengan distribusi kelas yang relatif seimbang.

Efektivitas *SMOTE* sendiri dipengaruhi oleh berbagai faktor, seperti tingkat ketidakseimbangan kelas, tumpang tindih antar kelas, serta keberadaan noise [39]. Selain itu, penerapan *SMOTE* tanpa pembersihan data yang memadai berpotensi menambah noise pada data latih [40]. Dalam kondisi tertentu, penggunaan *SMOTE* juga tidak selalu diperlukan apabila model klasifikasi yang digunakan telah memiliki kemampuan yang cukup kuat [41].

Penerapan *Particle Swarm Optimization (PSO)* menunjukkan peningkatan kinerja yang signifikan dibandingkan model dasar, dengan nilai *accuracy* sebesar 84,11%, *precision* 81,93%, *recall* 92,21%, dan *f1-score* 86,35%. Nilai *recall* yang tinggi menunjukkan bahwa optimasi parameter menggunakan *PSO* mampu meningkatkan kemampuan model dalam mendeteksi pasien yang menderita penyakit jantung. Kombinasi *Random Forest–PSO* dengan *SMOTE* menghasilkan kinerja terbaik dengan nilai *accuracy* 85,11%, *precision* 84,62%, *recall* 89,74%, dan *f1-score* 86,77%.

Secara keseluruhan, kombinasi *Random Forest–PSO* dengan *SMOTE* merupakan konfigurasi terbaik dalam mendeteksi penyakit jantung karena menghasilkan performa yang stabil dan seimbang pada seluruh metrik evaluasi. Meskipun demikian, kontribusi *SMOTE* terhadap model yang telah dioptimasi menggunakan *PSO* tergolong kecil, sehingga peningkatan kinerja lebih dominan dipengaruhi oleh proses optimasi parameter.

IV. SIMPULAN

Berdasarkan hasil penelitian, kombinasi Correlation Feature Selection (CFS), Synthetic Minority Oversampling Technique (SMOTE), Particle Swarm Optimization (PSO), dan Random Forest mampu meningkatkan performa klasifikasi penyakit jantung. Penerapan CFS berhasil menghilangkan atribut yang kurang relevan sehingga model dibangun menggunakan fitur yang lebih representatif. Hasil eksperimen menunjukkan bahwa optimasi hiperparameter menggunakan PSO memberikan peningkatan performa yang lebih signifikan dibandingkan penerapan SMOTE secara tunggal. Model terbaik diperoleh pada kombinasi Random Forest–PSO dengan SMOTE setelah penerapan CFS, dengan nilai accuracy sebesar 85,11%, precision 81,91%, recall 92,07%, dan f1-score 86,77%. Hasil tersebut menunjukkan bahwa kombinasi seleksi fitur, penyeimbangan data, dan optimasi hiperparameter mampu meningkatkan kemampuan model dalam mengklasifikasikan penyakit jantung. Selain itu, penggunaan 10-fold cross-validation menghasilkan proses evaluasi yang lebih stabil dan membantu mengurangi potensi bias pada pengujian model. Untuk penelitian selanjutnya, disarankan menggunakan dataset yang lebih besar dan beragam, mengevaluasi metode penyeimbangan data lain seperti *ADASYN*, *Borderline-SMOTE*, atau *Random Over-Sampling*, serta membandingkan PSO dengan metode optimasi lain, seperti *Genetic Algorithm* atau *Bayesian Optimization*. Selain itu, penelitian selanjutnya dapat membandingkan performa Random Forest dengan algoritma lain, seperti *XGBoost* atau *Deep Learning*, untuk memperoleh model klasifikasi penyakit jantung yang lebih optimal.

DAFTAR PUSTAKA

[1] K. K. R. Indonesia, "Kementerian Kesehatan 2024," 2024, *LMS Kementerian Kesehatan RI*. [Online]. Available: https://lms.kemkes.go.id/courses/35bff824-437e-4557-b37a-94b128c43333

[2] A. H. Anwar, "Sistematic Review Faktor Resiko Penyakit Jantung," *Indones. J. Heal. Res. Innov.*, vol. 02, no. 01, pp. 57–69, 2025, doi: https://doi.org/10.64094/fqanc998.

[3] G. A. Ansari, S. S. Bhat, M. D. Ansari, S. Ahmad, J. Nazeer, and A. E. M. Eljialy, "Performance Evaluation of Machine Learning Techniques (MLT) for Heart Disease Prediction," *Comput. Math. Methods Med.*, vol. 2023, no. Ml, 2023, doi: 10.1155/2023/8191261.

[4] V. Lampos, J. Mintz, and X. Qu, "An artificial intelligence approach for selecting effective teacher communication strategies in autism education," *npj Sci. Learn.*, vol. 6, no. 1, 2021, doi: 10.1038/s41539-021-00102-x.

[5] E. Erlin, Y. Desnelita, N. Nasution, L. Suryati, and F. Zoromi, "Dampak SMOTE terhadap Kinerja Random Forest Classifier berdasarkan Data Tidak seimbang," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 21, no. 3, pp. 677–690, 2022, doi: 10.30812/matrik.v21i3.1726.

[6] P. A. Jusia, A. Rahim, H. Yani, and J. Jasmir, "Improving Performance of KNN and C4.5 using Particle Swarm Optimization in Classification of Heart Diseases," *J. RESTI*, vol. 8, no. 3, pp. 333–339, 2024, doi: 10.29207/resti.v8i3.5710.

[7] N. Sureja, B. Chawda, and A. Vasant, "A novel salp swarm clustering algorithm for prediction of the heart diseases," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 25, no. 1, pp. 265–272, 2022, doi: 10.11591/ijeecs.v25.i1.pp265-272.

[8] A. Khaleel Faieq and M. M. Mijwil, "Prediction of heart diseases utilising support vector machine and artificial neural network," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 26, no. 1, pp. 374–380, 2022, doi: 10.11591/ijeecs.v26.i1.pp374-380.

[9] M. H. Musyaffa, T. H. Saragih, D. T. Nugrahadi, D. Kartini, and A. Farmadi, "Effectiveness of SMOTE in Enhancing Adult Autism Spectrum Disorder Diagnosis Predictive Performance With Missforest Imputation And Random Forest," *Indones. J. Electron. Electromed. Eng. Med. Informatics*, vol. 7, no. 2, pp. 270–280, 2025, doi: 10.35882/ijeeemi.v7i2.66.

[10] H. A. Salman, A. Kalakech, and A. Steiti, "Random Forest Algorithm Overview," *Babylonian J. Mach. Learn.*, vol. 2024, pp. 69–79, 2024, doi: 10.58496/bjml/2024/007.

[11] A. Samosir, M. S. Hasibuan, W. E. Justino, and T. Hariyono, "Komparasi Algoritma Random Forest, Naïve Bayes dan K- Nearest Neighbor Dalam klasifikasi Data Penyakit Jantung," *Pros. Semin. Nas. Darmajaya*, vol. 1, no. 0, pp. 214–222, 2021, [Online]. Available: https://jurnal.darmajaya.ac.id/index.php/PSND/article/view/2955

[12] A. P. Siregar, D. P. Purba, J. P. Pasaribu, and K. R. Bakara, "Implementasi Algoritma Random Forest Dalam Klasifikasi Diagnosis Penyakit Stroke," *J. Penelit. Rumpun Ilmu Tek.*, vol. 2, no. 4, pp. 155–164, 2023.

[13] A. Ghozali, H. Pratiwi, and S. S. Handajani, "Implementasi Data Mining Menggunakan Metode Random Forest Dan Support Vector Machine Dalam Klasifikasi Penyakit Diabetes," *Delta J. Ilm. Pendidik. Mat.*, vol. 11, no. 2, p. 147, 2023, doi: 10.31941/delta.v11i2.2686.

[14] Agung Khoeruddin, Fahri Andriansyah Sudrajat, Galuh Purnama, Iman Kuwangid, Kurnia Kurnia, and Ricky Firmansyah, "Optimasi Fitur Seleksi Random Forest Menggunakan GA Dalam Klasifikasi Data Penyakit Gagal Jantung," *J. Penelit. Teknol. Inf. dan Sains*, vol. 1, no. 2, pp. 01–09, 2023, doi: 10.54066/jptis.v1i2.323.

[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 Comput. Sci.*, vol. 10, pp. 1–51, 2024, doi: 10.7717/peerj-cs.1917.

[16] Z. Noroozi, A. Orooji, and L. Erfannia, "Analyzing the impact of feature selection methods on machine learning algorithms for heart disease prediction," *Sci. Rep.*, no. 0123456789, pp. 1–15, 2023, doi: 10.1038/s41598-023-49962-w.

[17] M. Salman, A. Nag, M. Mohisn, and S. Dev, "Healthcare Analytics Analyzing the impact of feature selection on the accuracy of heart disease prediction," *Healthc. Anal.*, vol. 2, no. February, p. 100060, 2022, doi: 10.1016/j.health.2022.100060.

[18] H. Optimization, K. Vishnu, V. Reddy, I. Elamvazuthi, A. A. Aziz, and S. Paramasivam, "applied sciences An Efficient Prediction System for Coronary Heart Disease Risk Using Selected Principal Components and," 2023.

[19] U. Hasanah, A. M. Soleh, and K. Sadik, "Effect of Random Under sampling, Oversampling, and SMOTE on the Performance of Cardiovascular Disease Prediction Models," *J. Mat. Stat. dan Komputasi*, vol. 21, no. 1, pp. 88–102, 2024, doi: 10.20956/j.v21i1.35552.

[20] U. Ungkawa and M. A. Rafi, "Data Balancing Techniques Using the PCA-KMeans and ADASYN for Possible Stroke Disease Cases," *J. Online Inform.*, vol. 9, no. 1, pp. 138–147, 2024, doi: 10.15575/join.v9i1.1293.

[21] A. A. Dhani, T. A. Y. Siswa, and W. J. Pranoto, "Perbaikan Akurasi Random Forest Dengan ANOVA Dan SMOTE Pada Klasifikasi Data Stunting," *Teknika*, vol. 13, no. 2, pp. 264–272, 2024, doi: 10.34148/teknika.v13i2.875.

[22] A. Singh, N. Prakash, and A. Jain, "Particle Swarm Optimization-Based Random Forest Framework for the Classification of Chronic Diseases," *IEEE Access*, vol. 11, no. December, pp. 133931–133946, 2023, doi: 10.1109/ACCESS.2023.3335314.

[23] R. Z. Zheng, Y. Zhang, and K. Yang, "A transfer learning-based particle swarm optimization algorithm for travelling salesman problem," *J. Comput. Des. Eng.*, vol. 9, no. 3, pp. 933–948, 2022, doi: 10.1093/jcde/qwac039.

[24] K. A. Barry, Y. Manzali, R. Flouchi, and M. Elfar, "Heart disease approach using modified random forest and particle swarm optimization," *IAES Int. J. Artif. Intell.*, vol. 14, no. 2, pp. 1242–1251, 2025, doi: 10.11591/ijai.v14.i2.pp1242-1251.

[25] T. P. Amal Dluha, Rizkianda Farma Saputra, Santoso, "Sentiment Analysis Aplication Ruang Guru Using Naive Bayes and Support Vector Machine," *JAISEN JournalJournal Adv. Inf. Syst. Eng.*, vol. 1, pp. 39–47, 2025, [Online]. Available: https://journal.unilak.ac.id/index.php/jaisen/home

[26] F. Aldi *et al.*, "STANDARDSCALER ' S POTENTIAL IN ENHANCING BREAST CANCER," vol. 5, no. 1, pp. 401–413, 2023.

[27] 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.

[28] S. Akinwamide, T. Fele, and O. A. Ojo, "Comparative evaluation of supervised machine learning algorithms for breast cancer prediction using the Wisconsin diagnostic dataset," vol. 24, no. 02, pp. 196–203, 2025.

[29] 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.

[30] V. Artanti, "Classification of Cardiovascular Diseases Using the K-Nearest Neighbors (KNN) Algorithm," *IEESE Int. J. Sci. Technol.*, vol. 13, no. 2, pp. 12–27, 2024, doi: https://doi.org/10.62411/tc.v23i2.10061.

[31] R. R. Adhitya, Wina Witanti, and Rezki Yuniarti, "Perbandingan Metode Cart Dan Naïve Bayes Untuk Klasifikasi Customer Churn," *INFOTECH J.*, vol. 9, no. 2, pp. 307–318, 2023, doi: 10.31949/infotech.v9i2.5641.

[32] I. K. Nti, "Performance of Machine Learning Algorithms with Different K Values in K-fold Cross- Validation," *I.J. Inf. Technol. Comput. Sci.*, no. December, pp. 61–71, 2021, doi: 10.5815/ijitcs.2021.06.05.

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

[34] S. Sathyanarayanan, "Confusion Matrix-Based Performance Evaluation Metrics," *African J. Biomed. Res.*, no. November, pp. 4023–4031, 2024, doi: 10.53555/ajbr.v27i4s.4345.

[35] D. Kurniadi, F. Nuraeni, N. Faturrohman, and A. Mulyani, "Klasifikasi Perputaran Karyawan Perusahaan Menggunakan Algoritma Random Forest dan Random Over-sampling," *Edu Komputika J.*, vol. 10, no. 2, pp. 93–103, 2024, doi: 10.15294/edukomputika.v10i2.73782.

[36] A. E. Saputra Rusdi, A. Rangkuti, N. Aris et., "Perbandingan ANN, Random Forest, dan XGBOOST Dalam Klasifikasi Antibiotik Dengan Penerapan Metode Sampling," vol. 12, no. 4, 2025.

[37] Z. Fan, B. Liu, and X. Yan, "Cardiovascular Disease Prediction Based on Machine Learning," pp. 404–411, 2024, doi: 10.5220/0012939000004508.

[38] W. J. P. Azwar Damari, Taghfirul Azhima Yoga Siswa, "IMPLEMENTATION OF THE PSO-SMOTE METHOD ON THE NAIVE BAYES ALGORITHM TO ADDRESS CLASS IMBALANCE IN LANDSLIDE DISASTER DATA PENERAPAN METODE PSO-SMOTE PADA ALGORITMA NAIVE BAYES UNTUK MENGATASI CLASS IMBALANCE," vol. 10, no. 1, pp. 332–343, 2025.

[39] M. Carvalho, A. J. Pinho, and S. Brás, "Resampling approaches to handle class imbalance : a review from a data perspective," *J. Big Data*, vol. 55, pp. 1–29, 2025, doi: 10.1186/s40537-025-01119-4.

[40] A. Arafa, N. El-fishawy, M. Badawy, and M. Radad, "RN-SMOTE : Reduced Noise SMOTE based on DBSCAN for enhancing imbalanced data classification," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 8, pp. 5059–5074, 2022, doi: 10.1016/j.jksuci.2022.06.005.

[41] A. Sakho and E. Malherbe, "Do we need rebalancing strategies ? A theoretical and empirical study around SMOTE and its variants," *arXiv Prepr.*, pp. 1–18, 2025, doi: https://arxiv.org/abs/2402.03819.