DETEKSI PENYAKIT SERANGAN JANTUNG MENGGUNAKAN RANDOM FOREST DENGAN VARIASI *SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE* (SMOTE) DAN *PRINCIPAL COMPONENT ANALYSIS* (PCA)

SKRIPSI

Oleh :
MIFTAHUL ARIFIN
NIM. 220605110161

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026

DETEKSI PENYAKIT SERANGAN JANTUNG MENGGUNAKAN RANDOM FOREST DENGAN VARIASI *SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE* (SMOTE) DAN *PRINCIPAL COMPONENT ANALYSIS* (PCA)

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :
MIFTAHUL ARIFIN
NIM. 220605110161

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2026

HALAMAN PERSETUJUAN

DETEKSI PENYAKIT SERANGAN JANTUNG MENGGUNAKAN RANDOM FOREST DENGAN VARIASI *SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE* (SMOTE) DAN *PRINCIPAL COMPONENT ANALYSIS* (PCA)

SKRIPSI

Oleh :
MIFTAHUL ARIFIN
NIM. 220605110161

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 10 Desember 2025

Pembimbing I,	Pembimbing II,
Dr. Cahyo Crysdian, M.CS	Roro Inda Melani, MT, M.Sc
NIP. 19740424 200901 1 008	NIP. 19780925 200501 2 008

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Supriyono, M. Kom
NIP. 19841010 201903 1 012

HALAMAN PENGESAHAN

DETEKSI PENYAKIT SERANGAN JANTUNG MENGGUNAKAN RANDOM FOREST DENGAN VARIASI *SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE* (SMOTE) DAN *PRINCIPAL COMPONENT ANALYSIS* (PCA)

SKRIPSI

Oleh :
MIFTAHUL ARIFIN
NIM. 220605110161

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 03 Maret 2026

Susunan Dewan Penguji

Ketua Penguji	: Dr. M. Amin Hariyadi, M.T NIP. 19670118 200501 1 001	( )
Anggota Penguji I	: Okta Qomaruddin Aziz, M.Kom NIP. 19911019 201903 1 013	( )
Anggota Penguji II	: Dr. Cahyo Crysdian, M.CS NIP. 19780625 200801 2 006	( )
Anggota Penguji III	: Roro Inda Melani, MT, M.Sc NIP. 19780925 200501 2 008	( )

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Supriyono, M. Kom
NIP. 19841010 201903 1 012

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama	: Miftahul Arifin
NIM	: 220605110161
Fakultas / Program Studi	: Sains dan Teknologi / Teknik Informatika
Judul Skripsi	: Deteksi Penyakit Serangan Jantung Menggunakan Random Forest dengan Variasi *Synthetic Minority Oversampling Technique* dan *Principal Component Analysis* (PCA)

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 08 Oktober 2026

Yang membuat pernyataan,

Miftahul Arifin

NIM.220605110161

MOTTO

"Memperbaiki diri adalah awal dari setiap perubahan"

"Keseimbangan adalah seni menjaga arah antara usaha dan keyakinan agar perubahan itu benar-benar terwujud"

HALAMAN PERSEMBAHAN

Alhamdulillahi rabbil 'alamin puji syukur kepada Allah *Subḥānahu wa Ta'ālā* karena berkat rahmat dan petunjuk-Nya penulis dapat menyelesaikan skripsi ini. Shalawat serta salam selalu tercurahkan kepada Rasulullah *Ṣallallāhu 'Alaihi Wasallam* yang telah membawa kita dari zaman jahiliyah menuju *addinul Islam*. Skripsi ini penulis persembahkan kepada seluruh pihak yang telah berperan dan berjasa dalam perjalanan penelitian ini.

Pertama, penulis persembahkan skripsi ini kepada kedua orang tua tercinta. Terima kasih atas doa yang tidak pernah putus, dukungan yang selalu menguatkan, serta keyakinan dan kepercayaan yang diberikan kepada penulis dalam setiap langkah untuk menggapai cita-cita. Penulis juga mempersembahkan kepada kakak-kakak tercinta yang senantiasa memberikan motivasi, semangat, dan dukungan selama proses penyusunan skripsi ini.

Terakhir, penulis juga mempersembahkan kepada sahabat-sahabat seperjuangan skripsi dan teman-teman angkatan 2022 yang telah memberikan dukungan, semangat, motivasi, serta menemani dalam setiap proses, baik dalam suka maupun duka, hingga skripsi ini dapat terselesaikan.

Semoga segala kebaikan dan bantuan yang telah diberikan mendapatkan balasan terbaik dari Allah *Subḥānahu wa Ta'ālā*. *Aamiin*.

KATA PENGANTAR

Segala puji hanya milik Allah *Subḥānahu wa Ta'ālā* atas segala nikmat dan kasih sayang-Nya yang telah memudahkan penulis untuk menyelesaikan skripsi yang berjudul "DETEKSI PENYAKIT SERANGAN JANTUNG MENGGUNAKAN RANDOM FOREST DENGAN VARIASI *SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE* (SMOTE) DAN *PRINCIPAL COMPONENT ANALYSIS* (PCA)". Semoga shalawat dan salam senantiasa terlimpah kepada Nabi Muhammad *Ṣallallāhu 'Alaihi Wasallam*. Dan semoga kita semua mendapat syafaatnya di hari kiamat nanti, *Aamiin*. Penulis mengucapkan rasa syukur dan terima kasih yang tak terhingga kepada semua pihak-pihak yang selalu memberikan bantuan dan motivasi kepada penulis untuk dapat menyelesaikan skripsi ini. Ucapan ini penulis sampaikan kepada:

1. Prof. Dr. Hj. Ilfi Nurdiana, M.Si., CAHRM., CRMP, selaku Rektor Universitas Islam Negeri Malang, yang telah memberikan kesempatan dan fasilitas untuk menempuh pendidikan di universitas ini.
2. Dr. Agus Mulyono, M.Kes, selaku Dekan Fakultas Saintek Universitas Islam Negeri Malang, atas segala dukungan dan arahan yang diberikan selama proses studi.
3. Dr. Cahyo Crysdian, M.Cs. selaku dosen pembimbing I dan Roro Inda Melani, MT, M.Sc. selaku dosen pembimbing II yang telah memberikan bantuan dan arahan kepada penulis, sehingga bisa menuntaskan skripsi ini.
4. Dr. M. Amin Hariyadi, M.T. selaku dosen ketua penguji dan Okta Qomaruddin Aziz, M.Kom selaku dosen penguji I yang telah menguji serta

memberikan masukan sehingga penulis dapat menuntaskan skripsi dengan baik.

5. Segenap Dosen, Admin, Laboran dan Mahasiswa Program Studi Teknik Informatika yang telah memberikan banyak dukungan dan bimbingan selama pengerjaan skripsi ini.
6. Orang tua serta saudara saya yang selalu memberikan dukungan dan motivasi untuk terus berusaha, dan doa yang tak putus-putusnya selalu disampaikan agar dapat menuntaskan skripsi ini dengan lancar dan baik.

Malang, 9 Maret 2026

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

ABSTRAK

Arifin, Miftahul. 2026. **Deteksi Penyakit Serangan Jantung Menggunakan Random Forest Dengan Variasi *Synthetic Minority Oversampling Technique* (SMOTE) dan *Principal Component Analysis* (PCA)**. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Cahyo Crsydian, M.Cs (II) Roro Inda Melani, MT, M.Sc.

Kata Kunci: Random Forest, SMOTE, PCA, Serangan Jantung.

Penelitian ini bertujuan mengevaluasi efektivitas algoritma Random Forest melalui empat skenario pemodelan (standar, dengan PCA, dengan SMOTE, serta kombinasi keduanya) untuk mendeteksi serangan jantung pada populasi Indonesia menggunakan dataset 158.355 pasien. Metode penelitian mencakup prapemrosesan variabel kategorikal, penyeimbangan kelas dengan SMOTE, dan reduksi dimensi dengan PCA. Hasil menunjukkan bahwa Skenario Model 3 (Random Forest dengan SMOTE) memberikan performa paling unggul dan stabil. Dengan teknik One-Hot Encoding, model ini mencapai akurasi 80,87% dan skor ROC-AUC 89,56%. Sebaliknya, penerapan PCA menurunkan performa karena menghilangkan informasi penting pada fitur medis yang bersifat independen. Strategi penyeimbangan data merupakan faktor paling krusial dalam meningkatkan sensitivitas deteksi dini serangan jantung dibandingkan reduksi dimensi fitur.

ABSTRACT

Arifin, Miftahul. 2026. **Heart Attack Detection Using Random Forest With Variations Of Synthetic Minority Oversampling Technique (SMOTE) and Principal Component Analysis (PCA).** Thesis. Department of Informatics Engineering, Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University Malang. Advisors: (I) Dr. Cahyo Crysdian, M.Cs (II) Roro Inda Melani, MT, M.Sc

This research evaluates the Random Forest algorithm across four scenarios (standard, PCA, SMOTE, and hybrid) to optimize heart attack Detection in the Indonesian population using a dataset of 158,355 entries. The methodology includes categorical encoding, SMOTE for class balancing, and PCA for dimension reduction. Results indicate that Model Scenario 3 (Random Forest with SMOTE) provides the most superior and stable performance. Utilizing One-Hot Encoding, this model achieved 80.87% accuracy and an 89.56% ROC-AUC score. Conversely, PCA reduced performance by losing critical independent medical features. Data balancing is more vital than feature dimension reduction for enhancing early heart attack Detection sensitivity.

Keywords: Random Forest, SMOTE, PCA, Heart Attack.

البحث مستخلص

أريفين، مفتاحول. ٢٠٢٦. **الكشف عن أمراض النوبات القلبية باستخدام تقنية Random Forest مع تقنية إعادة أخذ العينات الاصطناعية للأقليات (SMOTE) والتحليل المكون الرئيسي (PCA)**. أطروحة. قسم هندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية، مالانج. المشرفون (I) :د. كاهيو كرسيديان، M.Cs (II) رورو إندا ميلاني، MT، .M.Sc

الكلمات المفتاحية: Random Forest، SMOTE، PCA، النوبة القلبية.

تهدف هذه الدراسة إلى تقييم فعالية خوارزمية الغابة العشوائية من خلال أربعة سيناريوهات للنمذجة) قياسي، مع PCA ، مع SMOTE، ومزيج من الاثنين (للكشف عن النوبات القلبية في السكان الإندونيسيين باستخدام مجموعة بيانات تضم ١٥٨,٣٥٥ مريضًا. تضمنت طريقة البحث المعالجة المسبقة للمتغيرات الفئوية، وموازنة الفئات باستخدام SMOTE ، وتقليل الأبعاد باستخدام PCA. أظهرت النتائج أن سيناريو النموذج ٣ (Random Forest مع SMOTE) قدم الأداء الأفضل والأكثر استقرارًا. باستخدام تقنية الترميز One-Hot Encoding ، حقق هذا النموذج دقة بنسبة ٨٠,٨٧٪. ودرجة ROC-AUC بنسبة ٨٩,٥٦٪. على العكس من ذلك، أدى تطبيق PCA إلى انخفاض الأداء لأنه أزال معلومات مهمة عن السمات الطبية المستقلة. كانت استراتيجية موازنة البيانات هي العامل الأكثر أهمية في تحسين حساسية الكشف المبكر عن النوبات القلبية مقارنة بتقليل أبعاد السمات.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Penyakit jantung merupakan salah satu penyebab utama kematian di dunia maupun di Indonesia. Menurut Organisasi Kesehatan Dunia (WHO, 2025), penyakit kardiovaskular bertanggung jawab atas sekitar 17,9 juta kematian setiap tahun secara global, yang mewakili 31% dari seluruh kematian di dunia. Di Indonesia, data terbaru dari Survei Kesehatan Indonesia (SKI) 2023 menunjukkan prevalensi penyakit jantung berdasarkan diagnosis tenaga kesehatan sebesar 0,85% pada seluruh kelompok umur, sedikit meningkat dibanding Riskesdas 2018 yang mencatat 1,5% pada populasi dewasa (SKI, 2023). Peningkatan ini berkorelasi dengan perubahan gaya hidup, urbanisasi, dan meningkatnya faktor risiko seperti hipertensi, diabetes, obesitas, merokok, dan riwayat keluarga (Kemenkes RI, 2021).

Meski penyakit jantung memerlukan diagnosis dini yang akurat untuk meningkatkan peluang kesembuhan, ketersediaan tenaga medis spesialis dan fasilitas kesehatan di Indonesia masih belum merata. Kementerian Kesehatan menetapkan target rasio dokter spesialis jantung dan pembuluh darah sebesar 1 per 100.000 penduduk, namun realisasinya masih di bawah target tersebut, sehingga akses layanan kardiovaskular di banyak kabupaten/kota belum optimal (Kemenkes RI, 2023). WHO menegaskan pentingnya peningkatan jumlah dan pemerataan tenaga kesehatan untuk mencapai *Universal Health Coverage* (UHC), dengan target global minimal 4,45 tenaga kesehatan (dokter, perawat, dan bidan) per 1.000 penduduk untuk menjamin akses layanan yang merata (WHO, 2024).

Di era digital saat ini, teknologi *machine learning* memiliki potensi besar untuk mengatasi keterbatasan sumber daya medis melalui sistem deteksi otomatis yang dapat diakses secara luas. Namun, implementasi teknologi ini dalam konteks penyakit jantung di Indonesia masih menghadapi sejumlah tantangan teknis. Salah satu permasalahan utama pada dataset medis adalah ketidakseimbangan kelas (*class imbalance*), di mana jumlah data pasien dengan kondisi positif jauh lebih sedikit dibandingkan pasien dengan kondisi negatif, yang dapat menyebabkan model cenderung bias terhadap kelas mayoritas. Permasalahan lainnya adalah dimensi data yang tinggi (*high dimensionality*) dengan banyak fitur yang berpotensi menyebabkan model menjadi kompleks, lambat, dan rentan terhadap *overfitting* (Amshi *et al*., 2023; Aubaidan *et al*., 2025).

Pemilihan algoritma Random Forest didasarkan pada kebutuhan untuk mengatasi secara efektif tantangan data spesifik yang dihadapi, terutama dalam dataset berskala besar seperti pada penelitian "Heart Attack Prediction in Indonesia" yang memiliki 27 variabel prediktor dan 158.355 *records*. Terdapat dua keunggulan utama yang menjadikan Random Forest pilihan ideal. Pertama, ketahanan terhadap *overfitting* dan kemampuan dalam menangani data kompleks. Dalam domain klinis, hubungan antar-faktor risiko sering kali bersifat non-linier dan kompleks, sehingga model sederhana seperti Logistic Regression sering gagal menangkap sinyal penting.

Meskipun Decision Tree tunggal dapat menangani non-linearitas, model tersebut rentan terhadap *overfitting* akibat sensitif terhadap variasi kecil pada data pelatihan. Random Forest mengatasi masalah ini melalui mekanisme *Bootstrap*

Aggregating (*Bagging*) dengan membangun banyak *tree* keputusan yang tidak berkorelasi, dan keputusan akhir diambil melalui *majority voting*. Pendekatan ini menurunkan varians model secara signifikan dan menghasilkan model yang lebih robust serta mampu melakukan generalisasi dengan baik pada data baru.

Kedua, kinerja superior Random Forest dalam pipa optimasi data menjadikannya sangat efektif ketika diintegrasikan dengan variasi seperti SMOTE dan PCA. Dalam konteks *class imbalance*, Random Forest terbukti meningkatkan *Recall* dan *F1-Score* secara drastis setelah dilakukan penyeimbangan data menggunakan SMOTE-ENN, dari hanya 12,58% menjadi 93,85%, peningkatan sebesar 81,27 poin persentase (Wei & Shi, 2025). Selain itu, dalam menangani *high dimensionality*, Random Forest secara intrinsik tangguh menghadapi data berdimensi tinggi, dan ketika dikombinasikan dengan *Principal Component Analysis (*PCA*)* yang mereduksi dimensi tanpa kehilangan informasi penting, akurasi model meningkat hingga 97,91% dengan *F1-Score* sebesar 97,88% (Wei & Shi, 2025).

Secara komparatif, Random Forest menawarkan keseimbangan optimal antara kinerja, kecepatan, dan stabilitas, dengan kebutuhan penyetelan parameter yang relatif minimal dibandingkan model *boosting* seperti *Gradient Boosted Trees* (GBM) dan XGBoost yang lebih kompleks serta rentan terhadap *overfitting*. Lebih jauh lagi, Random Forest menyediakan fitur *importance* bawaan dan dapat diintegrasikan dengan metode *Explainable AI* (XAI*)* seperti SHAP, yang krusial untuk interpretasi klinis dan membangun kepercayaan dokter terhadap hasil deteksi.

Sejumlah penelitian sebelumnya telah mengeksplorasi berbagai algoritma *machine learning* untuk deteksi penyakit jantung. Perbandingan antara Logistic Regression, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), dan Random Forest menggunakan *k-Fold cross -validation*, dan menemukan bahwa Random Forest memberikan akurasi tertinggi pada dataset Cleveland *Heart Disease* (Rimal *et al*., 2025). Komparasi berbagai model seperti Decision Tree, Random Forest, Support Vector Machine (SVM), Logistic Regression, Adaptive Boosting, dan K-Nearest Neighbors menunjukkan bahwa kombinasi teknik *feature engineering* dengan model berbasis *ensemble* mampu meningkatkan akurasi deteksi (Bouqentar *et al*., 2024).

Beberapa penelitian juga mulai mengintegrasikan variasi lanjutan untuk meningkatkan performa model. Priyanto *et al* (2025) menggabungkan PCA dan *K-Means* SMOTE-ENN dengan Random Forest pada data medis berdimensi tinggi, dan berhasil mencapai akurasi 97,56% serta AUC 97,73% pada dataset penyakit jantung peningkatan signifikan dibanding XGBoost (+14,73% AUC). Aswani *et al* (2025) merancang kerangka kerja deteksi CAD yang mengombinasikan PCA, SMOTE, dan *hyperparameter tuning* menggunakan GridSearchCV, serta menambahkan analisis interpretabilitas dengan SHAP untuk mengidentifikasi fitur penting. Wei & Shi (2025) menunjukkan bahwa penerapan SMOTE-ENN diikuti PCA mampu meningkatkan akurasi dari 85,26% menjadi 97,91% dan *F1-Score* menjadi 97,88% pada dataset Framingham.

Meskipun demikian, Penelitian tersebut masih dilakukan pada dataset internasional seperti *Cleveland*, *Pima Indians Diabetes*, dan *Framingham*. Belum

ditemukan studi yang secara khusus mengombinasikan SMOTE, PCA, dan Random Forest untuk data penyakit jantung berbasis populasi Indonesia. Celah ini menjadi peluang untuk mengevaluasi dampak kombinasi ketiga teknik tersebut terhadap akurasi, sensitivitas (*Recall*), dan kemampuan generalisasi model pada data dengan ketidakseimbangan kelas dan dimensi fitur yang tinggi.

Kondisi tersebut memperkuat urgensi untuk mengembangkan solusi teknologi deteksi yang mampu menjembatani kesenjangan antara kebutuhan diagnosis dini penyakit jantung dan keterbatasan sumber daya medis. Model *machine learning* seperti Random Forest terbukti efektif dalam mengidentifikasi faktor risiko secara cepat dan akurat, sehingga memungkinkan deteksi dini penyakit kronis pada populasi luas (Talebi Moghaddam *et al*., 2024). Algoritma ini juga konsisten menunjukkan performa unggul dibandingkan model lain pada dataset penyakit kardiovaskular berdimensi tinggi dan tetap robust terhadap *overfitting* (Hossain *et al*., 2024).

Untuk mengatasi ketidakseimbangan kelas yang umum terjadi pada data medis, teknik SMOTE telah terbukti efektif dalam meningkatkan kinerja model klasifikasi dengan menghasilkan distribusi data yang lebih seimbang dan mengurangi bias terhadap kelas mayoritas (Hairani *et al*., 2024; Zhang *et al*., 2024). Selain itu, penggunaan PCA mampu mereduksi dimensi data secara signifikan sambil mempertahankan sebagian besar informasi penting, sehingga mengurangi kompleksitas model, mempercepat pelatihan, dan menurunkan risiko *overfitting* (Mehrabinezhad *et al*., 2024).

Dalam Islam, upaya mendeteksi dan mencegah penyakit merupakan bentuk ikhtiar ilmiah yang sejalan dengan perintah Allah untuk memperhatikan dan mengenali tanda-tanda yang ada. Konsep deteksi ini selaras dengan firman Allah dalam QS. Muhammad ayat 30:

وَلَوْ نَشَآءُ لَاَرَيْنٰكَهُمْ فَلَعَرَفْتَهُمْ بِسِيْمٰهُمْ ۗوَلَتَعْرِفَنَّهُمْ فِيْ لَحْنِ الْقَوْلِۗ وَاللّٰهُ يَعْلَمُ اَعْمَالَكُمْ

"*Seandainya Kami berkehendak, niscaya Kami menunjukkan mereka kepadamu (Nabi Muhammad) sehingga engkau benar-benar dapat mengenali mereka melalui tanda-tandanya. Engkau pun benar-benar akan mengenali mereka melalui nada bicaranya. Allah mengetahui segala amal perbuatanmu.*" *(QS. Muhammad: 30)*

Menurut Tafsir Ibnu Katsir, ayat ini menjelaskan adanya ketentuan *sunnatullah* berupa tanda-tanda tertentu yang dapat dijadikan dasar bagi manusia untuk mendeteksi, mengenali, dan membedakan suatu keadaan. Kata *sīmāhum* ("tanda-tandanya") dipahami sebagai karakteristik atau ciri khusus yang termanifestasi, sehingga manusia dapat mengidentifikasi realitas yang tersembunyi melalui indikator-indikator yang tampak secara lahiriah. Tafsir ini menekankan bahwa proses pengenalan terhadap suatu kondis termasuk dalam konteks diagnosis medis berlandaskan pada pengamatan tanda-tanda objektif yang telah polanya telah ditetapkan dalam alam semesta.

Prinsip ini relevan dengan penelitian deteksi penyakit jantung, karena data medis ibarat "tanda-tanda" (*simāh*) yang harus dianalisis secara cermat agar kondisi pasien dapat dikenali dengan tepat. Algoritma *machine learning* seperti Random Forest, SMOTE, dan PCA berfungsi sebagai alat bantu untuk mengenali pola pada data dan "mendeteksi" risiko penyakit sejak dini, sehingga tindakan pencegahan dapat dilakukan secara lebih efektif dan terarah. Dengan demikian, proses deteksi

dalam penelitian ini merupakan upaya ilmiah yang sejalan dengan prinsip Qur'ani dalam memahami suatu keadaan melalui tanda-tanda yang Allah tampakkan.

Penelitian ini mengusulkan penggunaan algoritma Random Forest yang dikombinasikan dengan *Synthetic Minority Oversampling Technique* (SMOTE), serta *Principal Component Analysis* (PCA) berdasarkan data penyakit jantung berbasis populasi Indonesia. Pendekatan ini diharapkan menjadi kontribusi ilmiah dengan mengintegrasikan *balancing* data dan reduksi dimensi secara simultan sebelum pelatihan model Random Forest, yang belum banyak dieksplorasi dalam konteks deteksi penyakit jantung di Indonesia. Kombinasi ketiga metode ini diharapkan dapat menghasilkan model deteksi yang lebih akurat, *robust*, dan efisien secara komputasi.

1.2 Pernyataan Masalah

Model Random Forest sering mengalami penurunan performa saat menghadapi data medis yang tidak seimbang dan berdimensi tinggi. Oleh karena itu, diperlukan analisis pengaruh penerapan SMOTE dan PCA, baik secara terpisah maupun digabungkan, terhadap kinerja Random Forest berdasarkan metrik akurasi, presisi, *Recall*, *F1-Score*, dan ROC-AUC.

1.3 Batasan Masalah

Penelitian ini menggunakan dataset "Heart Attack Prediction in Indonesia" dari platform Kaggle yang berisi 158.355 *record* dengan 27 variabel prediktor dan 1 variabel target.

1.4 Tujuan Penelitian

Untuk membandingkan hasil evaluasi model pada empat skenario (Random Forest saja, Random Forest dengan SMOTE , Random Forest dengan PCA, dan kombinasi Random Forest dengan SMOTE dan PCA) berdasarkan metrik akurasi, presisi, *Recall*, *F1-Score*, dan ROC-AUC.

1.5 Manfaat Penelitian

Untuk institusi kesehatan, Tersedianya model berbasis web yang dapat diimplementasikan sebagai alat bantu diagnosis awal, sehingga membantu tenaga kesehatan indonesia untuk mempercepat *screening* penyakit jantung, khususnya di daerah dengan keterbatasan tenaga medis spesialis. Dengan demikian, dapat mendukung pengambilan keputusan klinis berbasis data dan meningkatkan pemerataan layanan kesehatan kardiovaskular.

BAB II

STUDI PUSTAKA

Priyanto *et al* (2025) meneliti *health data classification* dengan mengintegrasikan PCA dan K-Means SMOTE-ENN pada model Random Forest untuk memperbaiki performa pada data medis yang tinggi dimensinya dan tidak seimbang. Pada dataset *Pima Indians Diabetes*, metode gabungan ini menghasilkan akurasi 98,41% dan nilai AUC 98,33%. Sedangkan pada dataset *Heart Disease*, akurasinya mencapai 97,56% dan AUC 97,73%. Dibandingkan dengan metode sebelumnya seperti SMOTE dan *Stacking Ensemble* pada dataset Diabetes, serta XGBoost pada dataset *Heart Disease*, metode yang diusulkan menunjukkan peningkatan signifikan, yakni sekitar 2,91% terhadap metode SMOTE/*Stacking* di Diabetes, dan peningkatan akurasi +14,73% AUC dibanding XGBoost di dataset *Heart Disease*.

Aswani *et al* (2025) mengajukan kerangka kerja deteksi *Coronary Artery Disease* (CAD) yang menggabungkan beberapa teknik: reduksi dimensi menggunakan PCA, penanganan ketidakseimbangan kelas dengan SMOTE, dan optimasi *hyperparameter* lewat GridSearchCV (parameter seperti jumlah estimator, kedalaman maksimum, dan minimal sampel split). Untuk interpretabilitas model, digunakan SHAP (*SHAPley Additive exPlanations*) untuk mengidentifikasi kontribusi fitur-fitur penting dalam deteksi CAD. Artikel melaporkan bahwa *framework* ini mengatasi kekurangan dari model sebelumnya terutama dalam hal redundansi fitur, ketidakseimbangan kelas, dan kurangnya transparansi model.

Nasution *et al* (2025) membandingkan performa tiga algoritma: Random Forest, Support Vector Machine (SVM), dan Logistic Regression (LR) dalam memdeteksi penyakit jantung menggunakan dataset UCI *Heart Disease*. Tanpa teknik seleksi fitur, Random Forest unggul dengan akurasi 89,7%, diikuti SVM dan LR. Ketika menggunakan teknik seleksi fitur seperti *Mutual Information* dan *Chi-Square*, performa tidak selalu meningkat signifikan, terkadang performa Random Forest turun saat menggunakan fitur seleksi, menunjukkan bahwa fitur awal sudah punya relevansi yang tinggi. Artikel menekankan bahwa Random Forest konsisten tampil baik dalam berbagai kondisi.

Dede Kurniadi *et al* (2025) menggunakan dataset Kaggle yang berjumlah 1.000 *record* dengan 14 variabel, di mana target klasifikasinya adalah penyakit jantung. Penelitian ini melakukan penyeimbangan kelas dengan SMOTE serta melakukan seleksi fitur berdasarkan konsultasi dengan dokter spesialis jantung untuk menjaga relevansi klinis. Model-yang diuji meliputi Naïve Bayes, Random Forest, dan kombinasi keduanya melalui *Ensemble Voting Classifier*. Hasil terbaik dicapai oleh metode ensemble, dengan akurasi 98,28%, presisi 98,41%, *Recall* 98,41%, dan *F1-Score* 98,41%, menunjukkan peningkatan dibanding algoritma individual.

Lamir *et al* (2025) menggunakan dataset *Heart-Disease* yang terdiri dari 303 sampel dan 14 fitur. Penelitian ini melakukan *preprocessing* , menggunakan klasifikasi dengan tiga algoritma: Logistic Regression, K-Nearest Neighbors (KNN), dan Random Forest. Untuk meningkatkan kinerja model, digunakan *hyperparameter* tuning melalui GridSearchCV dan RandomizedSearchCV. Hasil

pengujian menunjukkan bahwa Random Forest unggul dengan akurasi 91% dan *F1-Score* sebesar 0,89. Evaluasi juga mencakup metrik seperti *precision*, *Recall*, serta Confusion Matrix.

Wei & Shi (2025) menggunakan dataset *Framingham* dari Kaggle dengan 4.238 sampel dan 16 variabel (15 fitur + 1 target) untuk memdeteksi risiko penyakit arteri koroner (*Coronary Heart Disease*) dalam 10 tahun ke depan. Tahapan *preprocessing* mencakup pengisian *missing value* (mode untuk variabel kategorikal dan *mean* untuk numerik) pada variabel seperti pendidikan, BMI, glukosa, dll.; kemudian normalisasi/standardisasi data agar fitur-fitur berada dalam skala yang sama. Untuk menangani ketidakseimbangan kelas, digunakan metode SMOTE-ENN, dan setelah itu dilakukan reduksi dimensi menggunakan PCA. Model-model yang dibandingkan adalah Decision Tree, K-Nearest Neighbors, Support Vector Machine, XGBoost, dan Random Forest; semua dievaluasi menggunakan metrik *accuracy*, *precision*, *Recall*, *F1-Score*, dan AUC. Hasilnya: akurasi awal sebelum *balancing* ~85,26% dengan *F1-Score* rendah; setelah SMOTE-ENN naik menjadi sekitar 92,16% akurasi dan 93,85% *F1-Score*; dan setelah ditambah PCA akurasi menjadi 97,91% dan *F1-Score* 97,88%.

Tabel 2. 1 Relevansi terhadap penelitian terdahulu

No	Peneliti (Tahun)	Dataset / Kasus	Metode yang Digunakan	Hasil / Temuan Utama	Relevansi terhadap Penelitian Ini
1	Priyanto *et al*. (2025)	*Pima Indians Diabetes & Heart Disease*	PCA + K-*Mean*s SMOTE-ENN + Random Forest	Akurasi 98,41% (Diabetes) dan 97,56% (*Heart Disease*); peningkatan signifikan terhadap SMOTE/*Stacking* & XGBoost.	Membuktikan efektivitas integrasi reduksi dimensi dan balancing dalam meningkatkan performa model medis berdimensi tinggi.
2	Aswani *et al*. (2025)	*Coronary Artery Disease* (CAD)	PCA + SMOTE + GridSearchCV + SHAP	*Framework* meningkatkan akurasi dan interpretabilitas model CAD. SHAP menjelaskan fitur penting.	Menunjukkan pentingnya interpretabilitas model serta kombinasi reduksi dimensi dan *balancing* data.
3	Nasution *et al*. (2025)	*UCI Heart Disease*	Random Forest, SVM, *Logistic Regression* dengan & tanpa seleksi fitur	Random Forest unggul (akurasi 89,7%), bahkan tanpa seleksi fitur.	Mengonfirmasi stabilitas dan keandalan Random Forest pada klasifikasi medis tanpa perlu banyak prapemrosesan fitur.
4	Dede Kurniadi *et al*. (2025)	Kaggle *Heart Disease* (1.000 data, 14 fitur)	SMOTE + *Feature Selection* (dokter ahli) + *Ensemble Voting* (Naïve Bayes + Random Forest)	Akurasi 98,28%, Presisi/*Recall*/F1 = 98,41%.	Menunjukkan kombinasi *ensemble* meningkatkan performa dan mempertahankan relevansi klinis.

No	Peneliti (Tahun)	Dataset / Kasus	Metode yang Digunakan	Hasil / Temuan Utama	Relevansi terhadap Penelitian Ini
5	Lamir *et al*. (2025)	*UCI Heart Disease* (303 data, 14 fitur)	*Logistic Regression*, KNN, Random Forest + GridSearchCV	Random Forest terbaik dengan akurasi 91%, *F1-Score* 0,89.	Menunjukkan bahwa optimasi *hyperparameter* berpengaruh besar terhadap peningkatan performa model.
6	Wei & Shi (2025)	*Framingham Heart Study* (4.238 sampel, 16 variabel)	SMOTE-ENN + PCA + (DT, KNN, SVM, XGBoost, Random Forest)	Akurasi naik dari 85,26% → 97,91%; *F1-Score* naik dari 93,85% → 97,88%.	Menunjukkan kombinasi SMOTE-ENN dan PCA efektif mengatasi ketidakseimbangan kelas dan dimensi tinggi pada data medis.

Penelitian ini memiliki relevansi yang erat dengan penelitian terdahulu, baik dari sisi domain, metode, maupun tujuan yang ingin dicapai. Seluruh penelitian pada Tabel 2.1 bergerak dalam bidang klasifikasi penyakit medis berbasis *machine learning*, yang menegaskan bahwa topik ini memiliki urgensi tinggi dan terus berkembang.

Dari sisi metode, penelitian ini mengadopsi pendekatan yang telah terbukti efektif, yaitu SMOTE-ENN untuk menangani ketidakseimbangan kelas, PCA untuk reduksi dimensi, serta Random Forest sebagai algoritma klasifikasi utama. Ketiga pendekatan tersebut secara konsisten menghasilkan performa terbaik pada penelitian-penelitian sebelumnya. Penelitian ini mengintegrasikan ketiganya dalam satu *pipeline* yang utuh, sebagai upaya menghasilkan model klasifikasi penyakit yang akurat dan dapat diinterpretasikan secara klinis.

BAB III

DESAIN DAN IMPLEMENTASI SISTEM

3.1 Pengumpulan Dataset

Tahap berikutnya adalah mengumpulkan dataset yang akan digunakan, yaitu dataset "Heart Attack Prediction in Indonesia" dari platform Kaggle yang terdiri dari 158.355 *record* dengan 28 atribut yang bersumber dari data populasi lokal, berbeda dengan dataset internasional seperti Framingham Heart Study atau Cleveland Heart Disease Dataset yang sering digunakan dalam studi global. Perbedaan antara dataset nasional dan internasional terletak pada beberapa aspek utama sebagai berikut:

Tabel 3. 1 Perbedaan dataset nasional dan Internasional

Aspek	Dataset Internasional	Dataset Nasional (Indonesia)
Karakteristik Populasi	Populasi berasal dari negara maju (AS, Eropa), dengan gaya hidup, pola makan, dan sistem kesehatan yang berbeda.	Populasi Indonesia dengan karakteristik tropis, variasi etnis, pola makan tinggi karbohidrat, dan aktivitas fisik berbeda.
Distribusi Faktor Risiko	Faktor dominan: kolesterol tinggi, obesitas, dan usia lanjut.	Faktor dominan: hipertensi, stres, dan pola tidur tidak teratur.
Konteks Sosial & Lingkungan	Paparan polusi dan stres kerja relatif stabil dan terkontrol.	Lingkungan sosial dan paparan polusi bervariasi antar wilayah (urban vs rural).

Tabel 3.1 tersebut menunjukkan perbedaan karakteristik utama antara dataset internasional dan dataset nasional (Indonesia). Dataset internasional umumnya merepresentasikan populasi negara maju dengan pola hidup dan sistem

kesehatan yang lebih stabil, sehingga faktor risiko dominan cenderung berkaitan dengan kolesterol tinggi, obesitas, dan usia lanjut.

Sebaliknya, dataset nasional Indonesia mencerminkan kondisi populasi dengan karakteristik tropis, variasi etnis yang tinggi, serta perbedaan pola makan dan aktivitas fisik. Faktor risiko yang lebih dominan pada dataset nasional meliputi hipertensi, tingkat stres, dan pola tidur yang tidak teratur, dengan pengaruh lingkungan sosial dan paparan polusi yang bervariasi antar wilayah.

Perbedaan ini menunjukkan bahwa model prediksi penyakit jantung perlu mempertimbangkan konteks populasi dan lingkungan agar hasil deteksi lebih relevan dan akurat. Berikut merupakan detail fitur dataset "Heart Attack prediction in indonesia" :

Tabel 3. 2 *Detail* Fitur Dataset

No	Nama Variabel	Kategori	Tipe Data	Deskripsi	Nilai / Rentang
1	*age*	*Demographics*	Integer	Usia individu	25–90 tahun
2	*gender*	*Demographics*	String	Jenis kelamin	*Male / Female*
3	*region*	*Demographics*	String	Wilayah tempat tinggal	*Urban / Rural*
4	*income_level*	*Demographics*	String	Tingkat ekonomi	*Low / Middle / High*
5	*hypertension*	*Clinical*	Integer	Riwayat tekanan darah tinggi	1 = Ya, 0 = Tidak
6	*diabetes*	*Clinical*	Integer	Riwayat diabetes	1 = Ya, 0 = Tidak
7	*cholesterol_level*	*Clinical*	Integer	Total kolesterol (mg/dL)	100–400
8	*obesity*	*Clinical*	Integer	BMI > 30	1 = Ya, 0 = Tidak

No	Nama Variabel	Kategori	Tipe Data	Deskripsi	Nilai / Rentang
9	*waist_circumference*	*Clinical*	Integer	Lingkar pinggang (cm)	60–150
10	*family_history*	*Clinical*	Integer	Riwayat keluarga penyakit jantung	1 = Ya, 0 = Tidak
11	*smoking_status*	*Lifestyle*	String	Kebiasaan merokok	*Never / Past / Current*
12	*alcohol_consumption*	*Lifestyle*	String	Konsumsi alkohol	*None / Moderate / High*
13	*physical_activity*	*Lifestyle*	String	Aktivitas fisik	*Low / Moderate / High*
14	*dietary_habits*	*Lifestyle*	String	Pola makan	*Healthy / Unhealthy*
15	*air_pollution_exposure*	*Environmental*	String	Paparan polusi udara	*Low / Moderate / High*
16	*stress_level*	*Environmental*	String	Tingkat stres	*Low / Moderate / High*
17	*sleep_hours*	*Environmental*	Float	Rata-rata jam tidur per hari	3–9 jam
18	*blood_pressure_systolic*	*Medical*	Integer	Tekanan darah sistolik (mmHg)	90–200
19	*blood_pressure_diastolic*	*Medical*	Integer	Tekanan darah diastolik (mmHg)	60–120
20	*fasting_blood_sugar*	*Medical*	Integer	Gula darah puasa (mg/dL)	70–300
21	*cholesterol_hdl*	*Medical*	Integer	Kolesterol HDL (mg/dL)	30–100

No	Nama Variabel	Kategori	Tipe Data	Deskripsi	Nilai / Rentang
22	*cholesterol_ldl*	*Medical*	Integer	Kolesterol LDL (mg/dL)	70–250
23	*triglycerides*	*Medical*	Integer	Trigliserida (mg/dL)	50–500
24	*EKG_results*	*Medical*	String	Hasil EKG	Normal / Abnormal
25	*previous_heart_disease*	*Medical*	Integer	Riwayat penyakit	1 = Ya, 0 = Tidak
26	*medication_usage*	*Medical*	Integer	Konsumsi obat jantung saat ini	1 = Ya, 0 = Tidak
27	*participated_in_free_screening*	*Health System*	Integer	Ikut program skrining gratis	1 = Ya, 0 = Tidak
28	*heart_attack*	*Target*	Integer	Terjadi serangan jantung	1 = Ya, 0 = Tidak

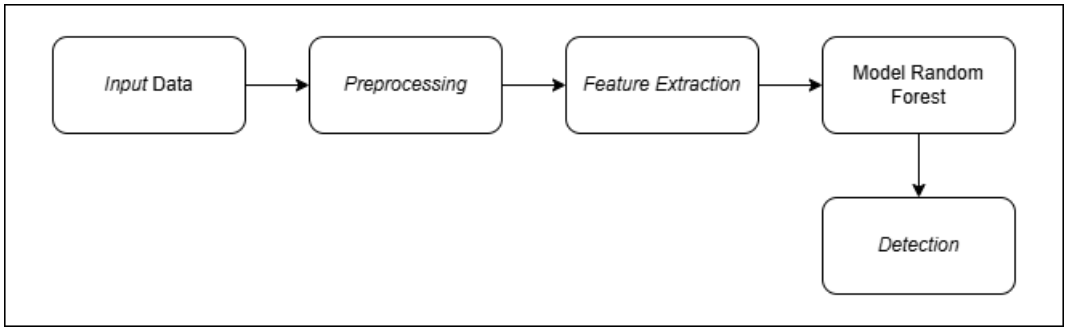
Pada tabel 3.2 terdapat 28 atribut dalam dataset, namun untuk prediktornya hanya menggunakan 27 atribut karena atribut *heart_attack* adalah atribut yang digunakan untuk mendiagnosa kesehatan pasien untuk dideteksi menggunakan Random Forest dengan variasi SMOTE dan PCA.

Tabel 3. 3 *Sample Dataset*

age	***gender***	***region***	***income_l***	***hyperten***	***diabetes***	***cholester***	***obesity***	***waist_cir***	***family_h***	***smoking***	***alcohol_consump***	***physical activity***	***dietary_habits***	***air_pollution_exp***	***stress_level***	***sleep_hours***	***blood_pr***	***blood_pr***	***fasting***	***cholester***	***cholester***	***triglyceri***	***EKG_results***	***previous***	***medicati***	***particina***	***heart_att***
60	*Male*	*Rural*	*Middle*	0	1	211	0	83	0	*Never*	*None*	*High*	*Unhealthy*	*Moderate*	*Moderate*	5,9706E+16	113	62	173	48	121	101	Normal	0	0	0	0
53	*Female*	*Urban*	*Low*	0	0	208	0	106	1	*Past*	*None*	*Moderate*	*Healthy*	*High*	*High*	5,64381E+15	132	76	70	58	83	138	Normal	1	0	1	0
52	*Male*	*Urban*	*Middle*	0	0	231	1	81	1	*Past*	*None*	*Low*	*Unhealthy*	*Low*	*Moderate*	8,24909E+15	131	71	129	34	148	191	Normal	0	1	1	0
25	*Female*	*Urban*	*Middle*	0	0	220	0	93	0	*Current*	*Moderate*	*High*	*Unhealthy*	*High*	*High*	4,87221E+15	121	77	85	32	115	147	Normal	0	1	1	1
64	*Female*	*Rural*	*Middle*	0	0	163	0	77	0	*Current*	*None*	*Low*	*Healthy*	*Moderate*	*High*	6,90039E+15	123	90	147	59	77	112	Normal	0	0	0	1
56	*Male*	*Rural*	*Middle*	1	0	196	1	111	0	*Past*	*None*	*Moderate*	*Unhealthy*	*Low*	*Low*	6,84366E+16	128	83	77	35	105	156	Normal	0	0	0	1
51	*Male*	*Urban*	*Middle*	1	0	132	0	95	0	*Never*	*None*	*Moderate*	*Unhealthy*	*Moderate*	*Moderate*	7,44694E+15	149	81	121	56	106	168	Abnormal	0	0	0	1

3.2 Desain Sistem

Sistem deteksi penyakit jantung pada penelitian ini dirancang dengan arsitektur horizontal, yaitu menggambarkan alur pemrosesan data secara berurutan dari *Input* hingga menghasilkan output deteksi. Arsitektur ini dipilih karena sesuai dengan sifat alur pemodelan *machine learning* yang membutuhkan tahap-tahap sistematis agar hasil deteksi yang diperoleh valid dan reliabel. Tahapan desain sistem Gambar 3.1:



Gambar 3. 1 Desain Sistem

Gambar 3.1 di atas menunjukkan alur proses deteksi serangan jantung. Data mentah terlebih dahulu melalui tahap *preprocessing* untuk membersihkan data dan melakukan *encoding* pada variabel kategorikal. Selanjutnya, dilakukan *feature extraction* untuk mendapatkan fitur yang paling relevan. Fitur yang telah diproses kemudian dimasukkan ke dalam model Random Forest untuk melakukan proses klasifikasi, sehingga menghasilkan *output* berupa hasil deteksi serangan jantung.

3.3 *Preprocessing*

Implementasi *prepocessing* pada penelitian ini dilakukan menggunakan dua metode pengolahan variabel kategorikal, yaitu Ordinal Encoding

dan One-Hot Encoding, yang kemudian dibandingkan hasilnya. Kedua metode ini diterapkan karena penelitian membutuhkan seluruh variabel dalam bentuk numerik, mengingat algoritma balancing SMOTE bekerja berdasarkan perhitungan jarak Euclidean antar titik data untuk menghasilkan sampel sintetis (Chawla et al., 2002).

Pada metode Ordinal Encoding, setiap kategori diubah menjadi nilai numerik berurutan (0, 1, 2, …) sehingga tidak menambah jumlah fitur. Namun metode ini dapat menimbulkan urutan semu pada variabel nominal. Sebaliknya, metode One-Hot Encoding mengubah setiap kategori menjadi vektor biner sehingga tidak menimbulkan urutan palsu, namun menyebabkan penambahan jumlah fitur.

3.3.1 Ordinal Encoding

Tahap Ordinal Encoding dilakukan untuk mengubah variabel kategorikal menjadi bentuk numerik yang memiliki urutan logis (*ordered categories*). Metode ini dipilih karena beberapa fitur pada dataset, seperti *physical_activity*, *income_level*, dan *stress_level*, memiliki urutan tingkatan yang jelas (misalnya *Low* < *Medium* < *High*). Teknik ini mengubah kategori menjadi nilai numerik 0, 1, dan 2 agar model dapat memahami hubungan tingkatannya.

Selain itu, penggunaan SMOTE (*Synthetic Minority Oversampling Technique*) dalam penelitian ini mengharuskan seluruh variabel berbentuk numerik, karena algoritma SMOTE bekerja dengan menghitung jarak *Euclidean* antar titik data untuk menghasilkan sampel sintetis (Chawla *et al*., 2002). Dengan demikian, Ordinal Encoding digunakan agar hubungan urutan antar kategori tetap terjaga, menghindari hilangnya informasi seperti pada One-Hot Encoding, serta menjaga

efisiensi dimensi dataset (Micheal, 2023). Misalkan sebuah fitur kategorikal X_i memiliki k kategori berurutan, maka setiap kategori direpresentasikan dengan bilangan bulat:

$$X_i = \begin{cases} 0, & \text{jika kategori rendah (}\textit{Low / Never}\text{)} \\ 1, & \text{jika kategori sedang (}\textit{Medium / Past / Moderate}\text{)} \\ 2, & \text{jika kategori tinggi (}\textit{High / Current}\text{)} \end{cases}$$

Dataset yang digunakan dalam penelitian ini terdiri atas 27 atribut, dengan 11 fitur kategorikal yang perlu dikonversi menjadi bentuk numerik pada tahap ini agar dapat diproses oleh algoritma pembelajaran mesin.

Tabel 3. 4 Transformasi Sebelum dan Sesudah Ordinal Encoding

Variabel	Kategori	Nilai
gender	*Female, Male*	0, 1
region	*Rural, Urban*	0, 1
income_level	*Low, Middle, High*	0, 1, 2
smoking_status	*Never, Past, Current*	0, 1, 2
alcohol_consumption	*None, Moderate, High*	0, 1, 2
physical_activity	*Low, Moderate, High*	0, 1, 2
dietary_habits	*Healthy, Unhealthy*	0, 1
air_pollution_exposure	*Low, Moderate, High*	0, 1, 2
stress_level	*Low, Moderate, High*	0, 1, 2
sleep_hours	*Kurang, Cukup, Berlebih*	0, 1, 2
EKG_results	*Normal, Abnormal*	0, 1

Tabel 3.4 menunjukkan proses transformasi variabel kategorikal menjadi nilai numerik menggunakan Ordinal Encoding. Setiap kategori pada variabel seperti *gender*, *region*, *income level*, dan lainnya diberi urutan nilai tertentu (0, 1, 2). Transformasi ini dilakukan agar data kategorikal dapat diproses oleh algoritma Random Forest pada tahap pembelajaran model.

3.3.2 One-Hot Encoding

Tahap One-Hot Encoding dilakukan sebagai metode alternatif untuk mengubah variabel kategorikal menjadi bentuk numerik tanpa memberikan hubungan urutan semu antar kategori. Berbeda dengan Ordinal Encoding, metode ini merepresentasikan setiap kategori sebagai vektor biner (0 atau 1), sehingga setiap kategori berdiri secara independen dan tidak memiliki hubungan hierarkis. Metode One-Hot Encoding dipilih karena sebagian fitur kategorikal dalam dataset bersifat nominal, seperti *gender*, *region*, dan *EKG_results*, yang tidak memiliki tingkatan nilai. Dengan pendekatan ini, model tidak mengasumsikan adanya urutan antar kategori, sehingga dapat menghindari bias akibat interpretasi urutan numerik yang tidak semestinya (Kuhn & Johnson, 2019).

Namun demikian, penerapan One-Hot Encoding menyebabkan peningkatan jumlah fitur (*dimensionality expansion*), karena setiap kategori direpresentasikan sebagai kolom baru. Kondisi ini berpotensi meningkatkan kompleksitas model dan kebutuhan komputasi, terutama ketika dikombinasikan dengan algoritma oversampling seperti SMOTE. Meskipun demikian, seluruh variabel tetap harus dalam bentuk numerik karena SMOTE bekerja dengan menghitung jarak Euclidean antar sampel untuk menghasilkan data sintetis (Chawla *et al*., 2002).

Secara matematis, jika suatu variabel kategorikal X_i memiliki k kategori, maka One-Hot Encoding akan mentransformasikannya menjadi k variabel biner sebagai berikut:

$$X_i = x_{i1}, x_{i2}, \dots, x_{ik}, x_{ij} \in 0,1 \tag{3.1}$$

dengan nilai 1 menunjukkan keberadaan kategori tertentu dan 0 untuk kategori lainnya. Dataset yang digunakan dalam penelitian ini memiliki 11 variabel kategorikal, yang setelah proses One-Hot Encoding menghasilkan penambahan jumlah fitur sesuai dengan banyaknya kategori pada masing-masing variabel.

Tabel 3. 5 Transformasi Sebelum dan Sesudah One-Hot Encoding

Variabel	Kategori	Representasi One-Hot Encoding
gender	*Female, Male*	*gender_Female, gender_Male*
region	*Rural, Urban*	*region_Rural, region_Urban*
income_level	*Low, Middle, High*	*income_Low, income_Middle, income_High*
smoking_status	*Never, Past, Current*	*smoking_Never, smoking_Past, smoking_Current*
alcohol_consumption	*None, Moderate, High*	*alcohol_None, alcohol_Moderate, alcohol_High*
physical_activity	*Low, Moderate, High*	*activity_Low, activity_Moderate, activity_High*
dietary_habits	*Healthy, Unhealthy*	*diet_Healthy, diet_Unhealthy*
air_pollution_exposure	*Low, Moderate, High*	*pollution_Low, pollution_Moderate, pollution_High*
stress_level	*Low, Moderate, High*	*stress_Low, stress_Moderate, stress_High*
sleep_hours	*Kurang, Cukup, Berlebih*	*sleep_Kurang, sleep_Cukup, sleep_Berlebih*
EKG_results	*Normal, Abnormal*	*EKG_Normal, EKG_Abnormal*

Tabel 3.5 menunjukkan proses transformasi variabel kategorikal menjadi representasi biner menggunakan One-Hot Encoding. Setiap kategori direpresentasikan sebagai fitur terpisah dengan nilai 0 atau 1. Pendekatan ini memastikan tidak adanya hubungan urutan antar kategori, namun berdampak pada meningkatnya dimensi dataset yang selanjutnya dianalisis pengaruhnya terhadap performa model Random Forest.

3.3.3 StandardScaler

Tahap berikutnya adalah menyamakan skala antar fitur numerik agar tidak ada variabel yang mendominasi perhitungan model. Metode yang digunakan adalah standardisasi, yang mengubah setiap fitur menjadi distribusi dengan rata-rata nol dan simpangan baku satu.

Untuk setiap fitur X_i, hasil standarisasi Z_i dihitung sebagai:

$$Z_i = \frac{X_i - \mu_i}{\sigma_i} \tag{3.2}$$

Keterangan :

X_i: nilai fitur ke-i,
μ_i: nilai rata-rata (*mean*) dari fitur ke-i,
σ_i: simpangan baku (*standard deviation*) dari fitur ke-i.

Setelah proses Ordinal Encoding atau One Hot Encoding dilakukan, tahap selanjutnya adalah standardisasi fitur numerik menggunakan StandardScaler. Standardisasi ini bertujuan untuk menyamakan skala seluruh variabel numerik, mengingat setiap fitur memiliki rentang nilai yang berbeda-beda, seperti *cholesterol_level*, *waist_circumference*, *blood_pressure*, dan *triglycerides*. Perbedaan skala tersebut dapat menyebabkan beberapa fitur mendominasi proses pembelajaran model, sehingga penyetaraan skala menjadi langkah penting sebelum masuk ke proses PCA dan pelatihan Random Forest.

Berdasarkan hasil implementasi StandardScaler, seluruh fitur numerik telah berhasil ditransformasikan sehingga memiliki skala yang seragam, dengan nilai rata-rata mendekati 0 dan simpangan baku mendekati 1. Proses ini memastikan bahwa setiap fitur memberikan kontribusi yang seimbang selama proses

pembelajaran model serta menghindari bias akibat perbedaan skala nilai. Berikut implementasinya :

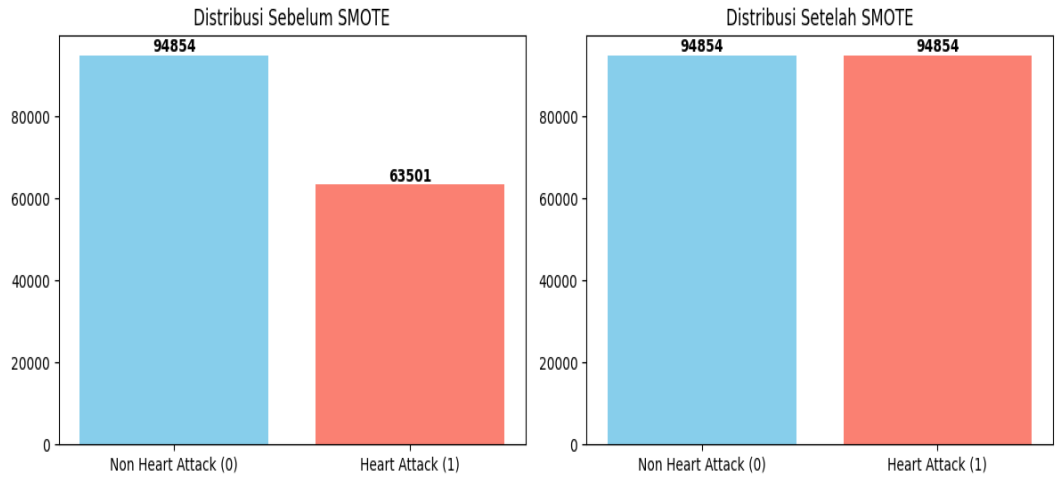
Tabel 3. 6 Sampel Data Hasil Stndardisasi

participa	-1.22443	0.816701	0.816701	-1.22443
medicati	0.998302	-1.00169	-1.00169	-1.00169
previous	-0.55138	1.813619	-0.55138	-0.55138
EKG_res	-2.00996	0.497518	0.497518	0.497518
triglyceri	0.432802	-0.21861	-2.03034	0.290306
cholester	0.010199	-0.07582	-0.84995	-1.05066
cholestar	1.956216	0.952907	-0.25106	-1.55537
fastina	0.238371	-0.26029	0.024661	-1.15074
blood_pr	-0.54968	-0.44986	-0.3487	0.747979
blood_pr	-0.90046	-0.16684	-1.3673	0.833551
sleep_ho	-0.14957	-1.7102	-0.21099	0.172535
stress_le	-0.22968	0.917704	0.917704	0.917704
air_pollu	-0.38507	-1.66554	-1.66554	0.895388
dietary	-1.22241	0.818048	0.818048	-1.22241
physical	1.071011	1.071011	-0.26417	-0.26417
alcohol	0.332779	0.332779	-3.00497	0.332779
smoking	1.636051	-1.10518	-0.19144	-0.19144
family_h	1.522246	-0.65649	1.522246	1.522246
waist_cir	1.117215	-0.27747	-1.30832	-0.3381
obesity	1.633933	-0.61201	-0.61201	1.633933
cholestar	0.755919	0.780762	-0.8837	-1.07678
diabetes	-0.53587	-0.53587	1.866115	-0.53587
hypertens	-0.71045	1.407555	-0.71045	-0.71045
income_l	-0.42124	0.979838	-0.42124	-0.42124
region	0.734178	0.734178	0.734178	-1.36205
gender	-1.0396	0.961899	-1.0396	0.961899
age	0.59004	-0.24326	0.756699	0.50671

Tabel 3.6 menunjukkan hasil standardisasi menggunakan StandardScaler pada seluruh fitur numerik dataset.

3.4 *Synthetic Minority Oversampling Technnique*

Dataset “Heart Attack Prediction in Indonesia” mengalami ketidakseimbangan kelas (*class imbalance*). Untuk mengatasi hal ini, pada Skenario 3 dan Skenario 4 diterapkan teknik SMOTE (*Synthetic Minority Over-sampling Technique*) yang membangkitkan sampel sintetis untuk kelas minoritas melalui interpolasi berbasis K-Nearest Neighbors yang dikembangkan oleh Chawla *et al.*, pada tahun 2002.



Gambar 3. 2 Distribusi Data Sebelum dan Sesudah SMOTE

Gambar 3.2 menunjukkan distribusi kelas sebelum dan sesudah dilakukan SMOTE. Distribusi sebelum *resampling* menunjukkan bahwa kelas 0 (*Non Heart Attack*) berjumlah 94854 data, sedangkan kelas 1 (*Heart Attack*) hanya berjumlah 63501 data. Ketidakseimbangan ini dapat menyebabkan model cenderung bias terhadap kelas mayoritas. Sedangkan distribusi setelah dilakukan SMOTE menunjukkan bahwa kedua kelas memiliki jumlah data yang seimbang, masing-masing 94854 data untuk kelas 0 (*Non Heart Attack*) dan 94854 data untuk kelas 1 (*Heart Attack*). Berikut langkah-langkah dalam menerapkan teknik SMOTE :

a. Mengidentifikasi kelas minoritas berdasarkan label target (*heart_attack* = 1).
b. Menentukan jumlah tetangga terdekat (k), umumnya menggunakan $k = 5$.
c. Untuk setiap sampel minoritas x_i, dipilih salah satu dari k-tetangga terdekatnya x_{nn}.

kemudian dibentuk data sintetis x_{new} dengan rumus:

$$x_{new} = x_i + \lambda(x_{nn} - x_i) \tag{3.3}$$

Keterangan :

λ : bilangan acak dalam rentang [0,1],
x_i : sampel minoritas yang dipilih,
x_{nn} : salah satu tetangga terdekat dari x_i.

d. Menggabungkan sampel sintetis dengan data asli untuk membentuk dataset seimbang antara kelas mayoritas dan minoritas.

3.5 *Feature Extraction*

Tahap *feature extraction* bertujuan menghasilkan representasi data yang lebih efisien dan informatif agar proses pelatihan model menjadi optimal. Pada penelitian ini digunakan metode *Principal Component Analysis* (PCA) sebagai teknik reduksi dimensi untuk menghilangkan korelasi antar fitur dan mengurangi redundansi informasi. PCA diterapkan pada Skenario 2 (Random Forest + PCA) dan Skenario 4 (Random Forest + SMOTE + PCA).

Beberapa penelitian terbaru menunjukkan bahwa penerapan PCA dapat meningkatkan performa model klasifikasi penyakit jantung dengan mengurangi redundansi fitur dan mempercepat waktu komputasi. Misalnya, Wei & Shi (2025) melaporkan peningkatan akurasi sebesar 5,75% setelah penambahan PCA pada model Random Forest yang telah diseimbangkan dengan SMOTE-ENN, sementara Priyanto *et al.,* (2025) juga menemukan bahwa kombinasi PCA dengan teknik *balancing* meningkatkan AUC hingga 97,73%.

Secara konsep, PCA bekerja dengan mencari kombinasi linear dari fitur-fitur asli yang mampu menjelaskan variasi terbesar pada data. Kombinasi ini

disebut komponen utama (*principal components*) yang saling ortogonal (tidak berkorelasi satu sama lain). Dengan demikian, PCA dapat menyederhanakan struktur data tanpa banyak kehilangan informasi penting. Secara matematis, langkah kerja PCA meliputi:

a. Setiap fitur distandardisasi agar memiliki *mean* = 0 dan *standard deviation* = 1.

$$z = \sigma_x - \bar{x} \tag{3.4}$$

Keterangan :

x : nilai asli dari suatu fitur
$\bar{x}$: rata-rata dari fitur x
σ_x : standar deviasi dari fitur
z : nilai hasil normalisasi

b. Mengukur hubungan antar fitur

$$\Sigma = \frac{1}{n-1}(Z^T Z) \tag{3.5}$$

Keterangan :

Σ : matriks kovarians dari data yang telah dinormalisasi
Z^T : transpose dari matriks Z
Z : data hasil normalisasi
n : jumlah sampel data

c. Matriks kovarians diuraikan menjadi vektor eigen (arah komponen utama) dan nilai eigen (besarnya variansi yang dijelaskan):

$$\Sigma v_i = \lambda_i v_i \tag{3.6}$$

Keterangan:
v_i : vektor eigen ke-i
λ_i : nilai eigen ke-i

d. Komponen dipilih berdasarkan kontribusi kumulatif variansi, dengan ambang ≥95%.

$$K = \min\left\{k : \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{p} \lambda_i} \geq 0.95\right\} \tag{3.7}$$

Keterangan :

λ_i : nilai eigen ke-i dari matriks kovarians.
p : jumlah total fitur awal
σ_x : jumlah komponen utama minimum yang mempertahankan ≥95\% keragaman data

e. Data hasil standardisasi dikalikan dengan matriks vektor eigen terpilih

$$Z = XW \tag{3.8}$$

Keterangan :

Z : matriks data asli
X : matriks vektor eigen (bobot) dari komponen utama
W : data hasil transformasi ke ruang komponen utama

Berikut merupakan langkah kerja PCA pada dua fitur dengan 5 sampel data:

Tabel 3. 7 Sampel data *Waist Circumference* dan *Cholesterol Level*

Sampel	***Waist Circumference***	***Cholesterol Level***
1	83	211
2	106	208
3	82	202
4	81	231
5	93	220

a. Setiap fitur distandardisasi agar memiliki *mean* = 0 dan *standard deviation* = 1

Hitung *mean* dan standar deviasi:

$$\bar{w} = 89.0, \qquad s_w = 10.0499; \quad \bar{c} = 214.4, \qquad s_c = 11.9746$$

Hitung *z-score*:

$$z_w = \frac{x_w - 89.0}{10.0499}, \qquad z_c = \frac{x_c - 214.4}{11.9746}$$

Hasil (dibulatkan 4 desimal):

$$Z = \begin{bmatrix} -0.5970 & -0.2843 \\ 1.6916 & -0.5348 \\ -0.7960 & -1.0329 \\ -0.8965 & 1.3828 \\ 0.5979 & 0.4663 \end{bmatrix}$$

b. Mengukur hubungan antar fitur

Matriks kovarians:

$$\Sigma = \frac{1}{4} Z^T Z = \begin{bmatrix} 1 & 0.628 \\ 0.628 & 1 \end{bmatrix}$$

c. Matriks kovarians diuraikan menjadi vektor eigen dan nilai eigen

$$\lambda_1 = 1.628, \lambda_2 = 0.372$$

$$v_1 = \frac{1}{\sqrt{2}}(1,1), v_2 = \frac{1}{\sqrt{2}}(1,-1)$$

d . Komponen dipilih berdasarkan kontribusi kumulatif variansi, dengan ambang ≥95%.

Explained Variance:

$$\text{PC1} = \frac{1.628}{2} = 0.814 = 81.4\%, \quad \text{PC2} = \frac{0.372}{2} = 18.6\%$$

Kumulatif = 100% → diperlukan 2 PC untuk ≥95% variansi.

Transformasi data ke PC:

$$\text{PC1} = \frac{1}{\sqrt{2}}(z_w + z_c), \quad \text{PC2} = \frac{1}{\sqrt{2}}(z_w - z_c)$$

Contoh perhitungan baris pertama:

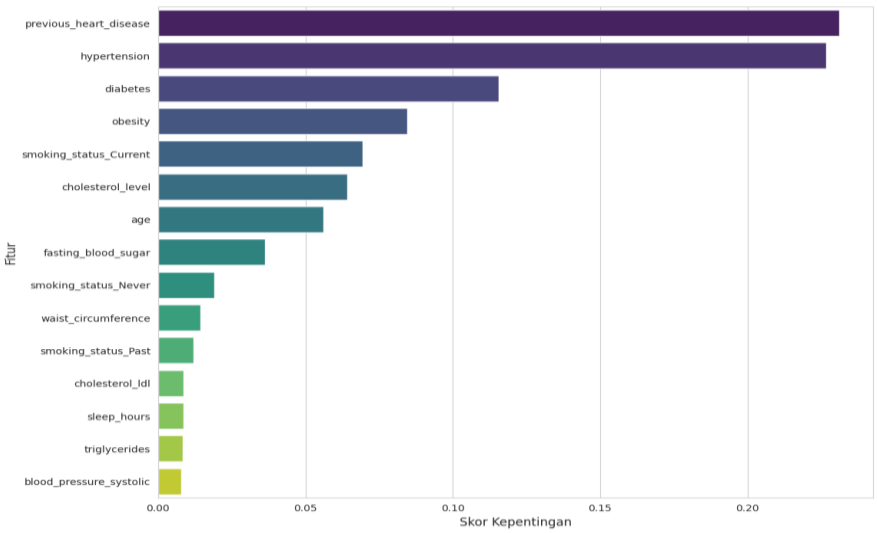
$$\text{PC1}_{(1)} = \frac{(-0.5970) + (-0.2843)}{\sqrt{2}} = -0.6234$$

$$\text{PC2}_{(1)} = \frac{(-0.5970) - (-0.2843)}{\sqrt{2}} = -0.2212$$

Interpretasi:

1. PC1 menggambarkan kenaikan bersama *waist* & *cholesterol* → faktor metabolik utama.
2. PC2 menggambarkan perbedaan relatif antar keduanya.

Berdasarkan hasil perhitungan manual tersebut, PCA pada dua fitur yang berkorelasi tinggi menghasilkan dua komponen utama (PC1 dan PC2) dengan total variansi 100%. Komponen PC1 menjelaskan 81,4% variansi data yang menggambarkan faktor metabolik pasien.



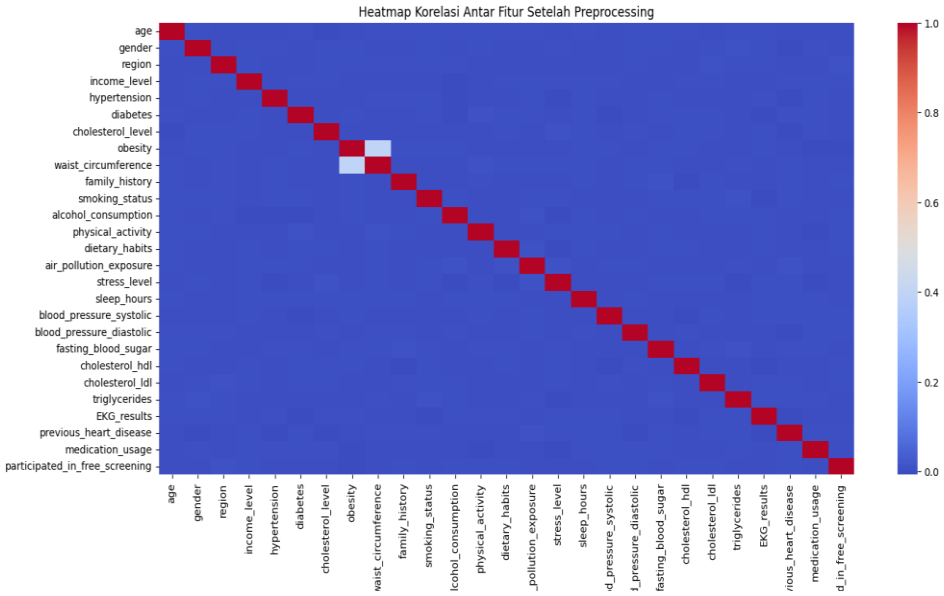
Gambar 3. 3 Top 15 Fitur Paling Berpengaruh dalam Deteksi (Sumber: Kaggle Notebook *Heart Attack Risk Classification* – Random Forest, Fadly Kurniawan)

Berdasarkan hasil *feature importance* pada gambar 3.3 dari model Random Forest, terlihat bahwa fitur *previous_heart_disease* dan *hypertension* memiliki skor kepentingan tertinggi. Hal ini menunjukkan bahwa riwayat penyakit jantung dan kondisi hipertensi merupakan faktor yang paling berpengaruh dalam prediksi serangan jantung pada penelitian ini.

Fitur lain seperti *diabetes*, *obesity*, dan *smoking_status_Current* juga menunjukkan kontribusi yang cukup besar terhadap model, sehingga dapat dikategorikan sebagai faktor risiko penting. Sementara itu, variabel *cholesterol_level*, *age*, dan *fasting_blood_sugar* memiliki pengaruh menengah dalam proses prediksi.

Adapun fitur dengan skor kepentingan rendah, seperti *sleep_hours*, *triglycerides*, dan *blood_pressure_systolic*, memberikan kontribusi yang relatif kecil terhadap hasil prediksi model. Secara keseluruhan, hasil ini menunjukkan

bahwa Random Forest mampu mengidentifikasi faktor-faktor risiko utama penyakit jantung secara konsisten dan menjadi dasar dalam pemilihan fitur dominan pada tahap analisis selanjutnya.



Gambar 3. 4 *Heatmap* Korelasi Antar Fitur

Berdasarkan *heatmap* korelasi pada Gambar 3.4, dapat dilihat bahwa sebagian besar fitur pada dataset memiliki nilai korelasi yang sangat rendah (mendekati 0). Hal ini menunjukkan bahwa:

1. fitur-fitur pada dataset tidak memiliki hubungan linear yang kuat,
2. tidak terjadi multikolinearitas yang signifikan,
3. pola hubungan antar variabel bersifat lemah atau tidak linear.

Berikut hasil 5 korelasi tertinggi juga memperkuat temuan tersebut:

Tabel 3. 8 Korelasi Antar Fitur

Pasangan Fitur	Nilai Korelasi	Interpretasi
waist_circumference ↔ obesity	0.395	Hubungan logis: lingkar pinggang lebih besar umumnya menandakan obesitas.
triglycerides ↔ *fasting_blood_sugar*	0.006	Korelasi sangat lemah, hampir tidak linear.
cholesterol_hdl ↔ *EKG_results*	0.007	Tidak ada hubungan linear yang berarti.
alcohol_consumption ↔ *air_pollution_exposure*	0.006	Tidak relevan secara klinis, cenderung noise.
stress_level ↔ *alcohol_consumption*	0.006	Hubungan sangat kecil, tidak signifikan.

Dari tabel 3.8 terlihat bahwa hanya satu pasangan fitur yang memiliki korelasi moderat, yaitu *waist circumference* dan *obesity*. Semua pasangan lainnya memiliki korelasi sangat kecil (<0.01). Dengan demikian, PCA tidak dapat mengurangi banyak fitur melalui penggabungan variansi fitur berkorelasi, karena hampir semua fitur bersifat independen satu sama lain.

3.6 Model Random Forest

Random Forest merupakan algoritma pembelajaran *ensemble* yang diperkenalkan oleh Breiman (2001) sebagai pengembangan dari metode *bootstrap aggregating* (*bagging*) dan *random decision forests*. Algoritma ini menggabungkan deteksi dari sejumlah *tree* keputusan yang dilatih menggunakan subset data yang berbeda, sehingga menghasilkan model yang lebih akurat, stabil, dan tahan terhadap *overfitting* (Ibrahim, 2023). Kinerja Random Forest sangat dipengaruhi oleh penentuan beberapa *hyperparameter* utama yang mengontrol kompleksitas, variasi,

dan stabilitas model. Tabel berikut menjelaskan beberapa *hyperparameter* penting yang digunakan dalam algoritma Random Forest.

Tabel 3. 9 *Tuning hyperparameter*

Parameter	Nilai	Deskripsi
n_estimators (J)	200, 400, 600, 800	Jumlah *tree* yang akan dibangun dalam *forest*. Lebih banyak *tree* umumnya lebih baik hingga titik konvergensi, tetapi menambah waktu komputasi secara linier.
max_features (m)	3, 5, "*sqrt*"	Jumlah fitur yang dipertimbangkan secara acak untuk setiap *split.*
max_depth	10, 20, *None*	Kedalaman maksimum setiap pohon. Membatasi kedalaman membantu mengontrol *overfitting*.
min_samples_split	2, 5, 10	Jumlah minimum sampel yang diperlukan untuk membelah *node*.
min_samples_leaf	1, 2, 4	Setiap *leaf node* minimal memiliki dua sampel. Digunakan untuk menjaga granularitas deteksi tanpa kehilangan detail informasi.

algoritma Random Forest bekerja melalui beberapa tahapan berikut:

a. Dari dataset berukuran n, algoritma membentuk beberapa subset melalui *sampling with replacement*. Setiap subset berukuran sama dengan dataset asli, namun sebagian sampel bisa terambil lebih dari sekali, sementara lainnya tidak terambil. Berdasarkan teori *bootstrap* Hartshorn tahun 2023, sekitar 63,2% data digunakan sebagai *Training data* dan 36,8% sisanya menjadi *Out-of-Bag* (OOB) data yang berfungsi sebagai *internal testing* tanpa memerlukan dataset validasi tambahan.

b. Dalam pembangunan setiap *tree*, hanya sebagian fitur yang dipilih secara acak di tiap *node*, bukan seluruhnya. Teknik ini meningkatkan *diversity* antar *tree* dan mengurangi risiko *overfitting*. Misalnya, dari 27 variabel prediktor, hanya $\sqrt{27} \approx 5$ fitur acak yang digunakan di setiap node (contoh: *age*, *hypertension*,

cholesterol_level, *smoking_status*, *stress_level*). Untuk pemisahan data di setiap node didasarkan pada nilai *Gini Impurity*, dengan rumus:

$$Gini(s_i) = 1 - \sum_{j=1}^{c} p_i^2 \tag{3.7}$$

Keterangan:
s_i : menunjukkan node/kelas
c : proporsi kelas ke-i dalam node
p_i^2 : jumlah total kelas

c. Titik split terbaik ditentukan berdasarkan nilai Gini Split, yaitu nilai rata-rata tertimbang dari impurity dua subset hasil split:

$$Gini_{split} = \sum_{i=1}^{k} \left(\frac{n_i}{n} \times Gini(s_i) \right) \tag{3.8}$$

Keterangan:
n_i : jumlah data pada subset ke-i
n : total data pada node
$Gini(s_i)$: nilai gini dari subset ke-i

d. Proses pembentukan node berlangsung secara rekursif hingga data dalam node homogen ($Gini = 0$) atau tidak ada fitur kategorikal yang dapat meningkatkan kemurnian, serta tree mencapai kedalaman maksimum (*max_depth*). Karena sebagian besar atribut bersifat kategorikal (misalnya *gender*, *region*, *income_level*, *smoking_status*, *dietary_habits*, *EKG_results*), pemisahan dilakukan dengan membandingkan kombinasi kategori, seperti {"*Male*"} vs {"*Female*"}, {"*Low*", "*Middle*"} vs {"*High*"}, atau {"*Never*", "*Past*"} vs {"*Current*"}. Kombinasi dengan nilai *Gini Split* terkecil dipilih sebagai *split* terbaik..

e. Setelah semua *tree* terbentuk, deteksi terhadap data uji dilakukan dengan mengambil suara dari masing-masing *tree* yang nantinya dibandingkan dengan label aslinya, lalu hasil akhir ditentukan bedasarkan kelas yang paling banyak dipilih (mayoritas). Rumus *majority voting* dapat dituliskan sebagai berikut:

$$\hat{y} = \arg\max_{c \in C} \sum_{j=1}^{T} I\big(h_j(x) = c\big) \tag{3.9}$$

Keterangan:

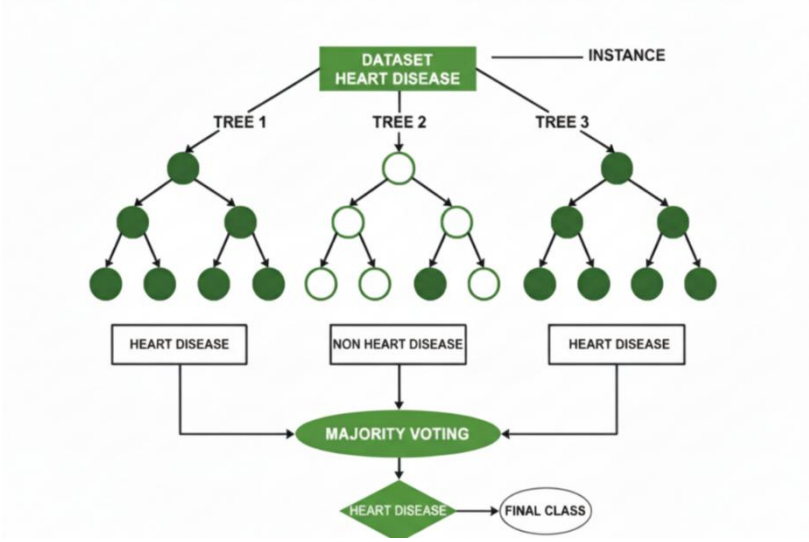
$\hat{y}$: kelas akhir yang dipilih oleh Random Forest
$\mathcal{C}$: himpunan seluruh kelas (misalnya {*Non Heart Attack*, *Heart Attack*})
T : jumlah total tree dalam Random Forest
$h_j(x)$: prediksi tree ke-j untuk sampel input x
$I(\cdot)$: fungsi indikator, bernilai 1 jika kondisi benar, dan 0 jika salah

f. Selanjutnya adalah mengevaluasi menggunakan *Out-of-Bag* (OOB) *Accuracy* untuk mengukur kinerja model tanpa dataset uji eksternal.

$$OOB_{\text{Accuracy}} = \frac{1}{N} \sum_{i=1}^{N} I\big(\widehat{y_{\text{OOB},i}} = y_i\big) \tag{3.10}$$

Keterangan :

N : Jumlah total sampel dalam dataset.
i : Indeks untuk setiap sampel data (dari 1 sampai (N)).
$\widehat{y_{\text{OOB},i}}$: Deteksi model untuk sampel (i) berdasarkan hanya data uji.
y_i : Label aktual (ground truth) untuk sampel ke-(i).
$I(.)$: Fungsi indikator: bernilai 1 jika deteksi benar, dan 0 jika salah.



Gambar 3. 5 Alur deteksi penyakit jantung menggunakan Random Forest.

Gambar 3.5 menunjukkan alur kerja algoritma Random Forest pada kasus klasifikasi penyakit jantung. Dataset awal dibagi menjadi beberapa subset menggunakan teknik *bootstrap sampling*, di mana setiap subset digunakan untuk membangun satu *decision tree* (*Tree* 1, *Tree* 2, *Tree* 3). Setiap *tree* dilatih secara independen dengan subset fitur yang dipilih secara acak di setiap node, sehingga masing-masing *Tree* menghasilkan keputusan klasifikasi yang mungkin berbeda. Pada contoh ini, *Tree* 1 memdeteksi kelas *Heart Disease*, sedangkan *Tree* 2 dan *Tree* 3 mendeteksi kelas *Non Heart Disease*.

Hasil deteksi dari seluruh *Tree* kemudian digabungkan menggunakan mekanisme *majority voting*, di mana kelas dengan suara terbanyak akan dipilih sebagai hasil akhir. Pada kasus ini, dua dari tiga *Tree* mendeteksi *Heart Disease*, sehingga sistem menetapkan kelas akhir (*Final Class*) sebagai *Heart Disease*. Proses ini memungkinkan Random Forest menghasilkan deteksi yang lebih stabil

dan mengurangi risiko *overfitting* dibandingkan hanya menggunakan satu *Tree* keputusan.

3.7 *Detection*

Output sistem berupa klasifikasi biner, yaitu:

a. 0 → Tidak terkena penyakit jantung.
b. 1 → Terkena penyakit jantung.

3.8 Implementasi Sistem Berbasis Web

Aplikasi deteksi serangan jantung ini diimplementasikan dalam bentuk web interaktif berbasis streamlit. Aplikasi dirancang agar pengguna dapat melakukan dua jenis interaksi utama, yaitu:

a. Mengunggah dataset riwayat pasien (CSV) untuk melakukan pelatihan ulang (*model retraining*) secara otomatis.
b. Meng *Input* data pasien secara manual melalui form interaktif per kolom, misalnya usia, tekanan darah, kadar kolesterol, status merokok, dan sebagainya.

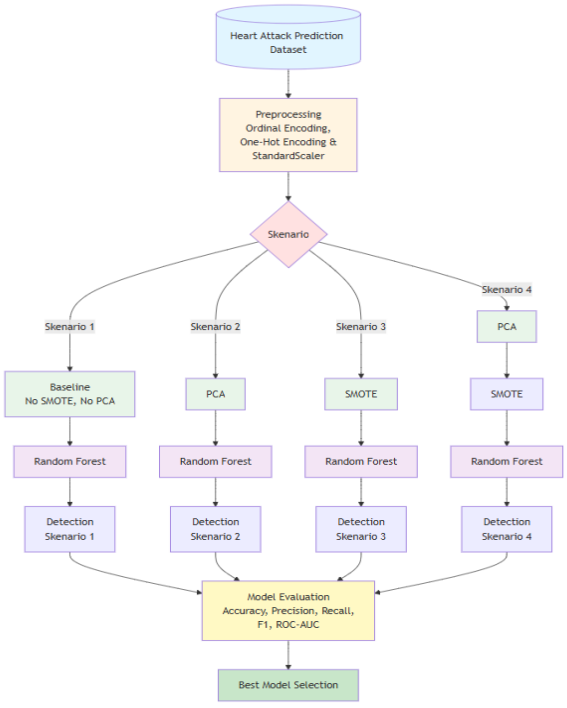
Setelah data dimasukkan, sistem akan menjalankan tahapan *preprocessing* , *balancing* data (SMOTE), dan reduksi dimensi (PCA) sesuai *pipeline* yang telah dijelaskan sebelumnya. Model kemudian dilatih ulang menggunakan data terbaru, dan hasil deteksi ditampilkan langsung pada halaman web berupa status pasien terkena serangan jantung atau tidak.

BAB IV

UJI COBA DAN PEMBAHASAN

4.1 Skenario Uji Coba

Pada penelitian ini, dilakukan perbandingan dua teknik *preprocessing* untuk variabel kategorikal, yaitu Ordinal Encoding dan One-Hot Encoding. Kedua teknik ini diuji secara terpisah pada setiap skenario untuk mengetahui pengaruhnya terhadap performa model deteksi serangan jantung. Pemilihan teknik *encoding* yang tepat sangat krusial karena dapat mempengaruhi bagaimana algoritma Random Forest menginterpretasikan hubungan antar fitur kategorikal dalam proses pembentukan Decision Tree (Zhu *et al*., 2024).



Gambar 4. 1 *Training proccess pipline* 4 skenario

Bagian ini juga menjelaskan prosedur pengujian yang dilakukan berdasarkan empat skenario model. Setiap skenario diuji menggunakan dataset "Heart Attack Prediction in Indonesia" dengan jumlah 158.355 *record* dan 27 fitur prediktor.

a. Skenario 1 (Random Forest saja) merupakan kondisi kontrol di mana algoritma Random Forest diterapkan langsung pada data yang hanya melalui *preprocessing* dasar, seperti *encoding* pada variabel kategorikal. Skenario ini bertujuan untuk menetapkan *baseline per*form*ance* yang akan menjadi acuan perbandingan bagi skenario lainnya. Penggunaan *baseline* tanpa teknik *preprocessing* lanjutan merupakan praktik standar dalam evaluasi model *machine learning* untuk memastikan validitas perbandingan dan menghindari bias hasil evaluasi (Ali *et al*., 2025; Wang *et al*., 2023).
b. Skenario 2 (Random Forest dengan PCA) Skenario ini mengimplementasikan *Principal Component Analysis* (PCA) untuk reduksi dimensi sebelum pelatihan Random Forest. PCA mentransformasikan fitur asli menjadi komponen utama yang saling ortogonal, dengan mempertahankan 95% varians dari data asli sehingga informasi yang relevan tetap terjaga.
c. Skenario 3 (Random Forest dengan SMOTE) menerapkan *Synthetic Minority Oversampling Technique* (SMOTE) sebelum pelatihan Random Forest untuk mengatasi ketidakseimbangan kelas. SMOTE bekerja dengan membangkitkan sampel sintetis dari kelas minoritas melalui interpolasi berbasis K-Nearest Neighbors, sehingga distribusi kelas menjadi lebih seimbang dan model tidak terlalu bias terhadap kelas mayoritas.

d. Skenario 4 (Random Forest dengan PCA dan SMOTE) mengombinasikan dua teknik variasi yakni SMOTE untuk menyeimbangkan distribusi kelas dan PCA untuk mereduksi dimensi sebelum pelatihan Random Forest. Urutan PCA terlebih dahulu, diikuti SMOTE, Sebuah studi menunjukkan bahwa setelah penerapan PCA, penggunaan SMOTE dapat membantu "mengompensasi" kehilangan informasi akibat reduksi fitur (Naseriparsa & Kashani, 2013). Kombinasi ini terbukti efektif dalam meningkatkan *Recall* dan AUC pada deteksi penyakit jantung, dengan peningkatan akurasi hingga >5% dibandingkan *baseline* (Priyanto *et al*., 2025; Wei & Shi, 2025).

4.1.1 Menghitung Kinerja Sistem

Penelitian ini menerapkan *K-Fold Cross -Validation* untuk memperoleh estimasi performa yang lebih reliabel. Dataset dibagi menjadi K lipatan (*Fold*) berukuran hampir sama; pada setiap iterasi, satu lipatan digunakan sebagai *test set* sementara K−1 lipatan lainnya berperan sebagai *Training set*. Proses ini diulang sebanyak K kali hingga seluruh data minimal sekali menjadi data uji, kemudian skor dari setiap iterasi dirata-ratakan untuk menghasilkan estimasi performa yang lebih stabil. Pada penelitian ini digunakan K=5, dipilih untuk menjaga keseimbangan antara bias, varians, dan efisiensi komputasi.

Selain itu, dievaluasi pula *Out-of-Bag* (OOB) *score* yang secara inheren melekat pada algoritma Random Forest. OOB *score* diperoleh dari deteksi terhadap data yang tidak digunakan untuk membangun setiap *tree* (*bootstrap sample*), sehingga dapat berfungsi sebagai validasi internal tanpa memerlukan data uji tambahan. Kombinasi *K-Fold Cross -Validation* dan OOB *score* memastikan proses

validasi menjadi lebih menyeluruh, sehingga performa model yang diperoleh dapat diandalkan sebelum diimplementasikan pada data nyata.

Setelah sistem dibangun, dilakukan evaluasi dengan menghitung akurasi, presisi, *F1-Score*, *Recall*, ROC-AUC menggunakan persamaan yang telah ditentukan. Confusion Matrix memiliki struktur sebagai berikut:

Tabel 4. 1 Confusion Matrix pada Klasifikasi Biner

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

Tabel 4.1 adalah klasifikasi dari Confusion Matrix. *True Positive* (TP) berarti model klasifikasi dengan benar memberi label jumlah tupel positif. *True Negative* (TN) berarti bahwa model klasifikasi dengan benar memberi label jumlah tupel negatif. *False Positive* (FP) berarti bahwa model klasifikasi memberi label yang salah untuk jumlah tupel negatif. *False negative* (FN) menunjukkan bahwa model klasifikasi memberi label yang salah untuk jumlah tupel positif.

Akurasi mengukur proporsi deteksi yang benar (baik positif maupun negatif) dibandingkan dengan keseluruhan data. Rumus untuk menghitung akurasi dapat dilihat pada Persamaan 4.1

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.1}$$

Presisi menunjukkan seberapa banyak deteksi positif yang benar-benar positif. Rumus untuk menghitung presisi dapat dilihat pada Persamaan 4.2.

$$Precision = \frac{TP}{TP + FP} \tag{4.2}$$

Dalam konteks medis, *False Positive* disebut juga *false alarm*, yaitu kondisi di mana model salah mendeteksi seseorang seolah-olah memiliki penyakit padahal tidak. Misalnya, model memdeteksi seseorang "terkena serangan jantung", padahal hasil pemeriksaan medis sebenarnya normal.

Recall mengukur proporsi data positif yang berhasil dideteksi dengan benar oleh model. Rumus untuk menghitung *Recall* dapat dilihat pada persamaan 4.3.

$$Recall = \frac{TP}{TP + FN} \tag{4.3}$$

Dalam konteks medis, *False negative* disebut juga *missed diagnosis*, yaitu ketika model gagal mengenali pasien yang sebenarnya menderita penyakit. Misalnya, pasien sebenarnya terkena serangan jantung, tetapi model memdeteksi "tidak terkena serangan jantung".

F1-Score adalah rata-rata harmonis antara presisi dan *Recall*. Rumus untuk menghitung *F1-Score* dapat dilihat pada Persamaan 4.4.

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4.4}$$

Harmonic mean dari *precision* dan *Recall*, memberikan keseimbangan antara keduanya. *Harmonic mean* adalah jenis rata-rata yang lebih sensitif terhadap nilai yang kecil. Jika salah satu nilai (*precision* atau *Recall*) rendah, maka hasil rata-ratanya akan turun drastis. *Harmonic mean* digunakan untuk menjaga keseimbangan antara *precision* dan *Recall*. Dalam konteks medis, hal ini sangat penting agar model tidak hanya fokus mendeteksi pasien yang sakit, tetapi juga meminimalkan kesalahan diagnosis pada pasien sehat. Dengan demikian, *F1-Score*

memberikan penilaian yang lebih realistis terhadap kinerja model dalam sistem deteksi penyakit jantung.

ROC-AUC mengukur kemampuan model dalam membedakan antara kelas positif dan negatif pada berbagai threshold. Rumus untuk menghitung ROC-AUC dapat dilihat pada Persamaan 4.5 dan 4.6.

$$TPR = \frac{TP}{TP + FN} \tag{4.5}$$

$$FPR = \frac{FP}{FP + TN} \tag{4.6}$$

4.2 Hasil Uji Coba

Untuk mengevaluasi pengaruh dari penanganan data yang tidak seimbang dan korelasi antar fitur terhadap performa model, dilakukan pengujian pada empat skenario berbeda. Pada setiap skenario, model Random Forest tidak langsung digunakan dengan parameter *default*. Untuk memperoleh performa terbaik dari model Random Forest pada setiap skenario, dilakukan proses *hyperparameter tuning* menggunakan RandomizedSearchCV. RandomizedSearchCV dipilih karena mampu mengeksplorasi ruang parameter lebih efisien, memberikan performa yang mendekati GridSearchCV, serta mendukung desain penelitian yang membutuhkan perbandingan skenario secara adil dengan parameter model yang konsisten. Selain itu juga digunakan untuk mencari kombinasi parameter terbaik dengan teknik *cross -validation* sebanyak 5 lipatan (*5-Fold cross -validation*). Adapun paramater yang disesuaikan seperti pada Tabel 3.6.

Melalui proses ini, diperoleh kombinasi parameter optimal untuk masing-masing skenario yang kemudian digunakan untuk pelatihan model dan evaluasi. Setelah model dilatih menggunakan parameter terbaik, dilakukan evaluasi performa menggunakan Confusion Matrix. Confusion Matrix memberikan gambaran seberapa baik model mampu mengklasifikasikan dua kelas pada dataset, yaitu kelas 0 (*Non Heart Attack*) dan kelas 1 (*Heart Attack*). Dari Confusion Matrix diperoleh metrik evaluasi seperti: akurasi, presisi, *Recall*, dan *F1-Score*. Berikut adalah hasil uji coba masing-masing skenario.

4.2.1 Uji Coba Model Dasar

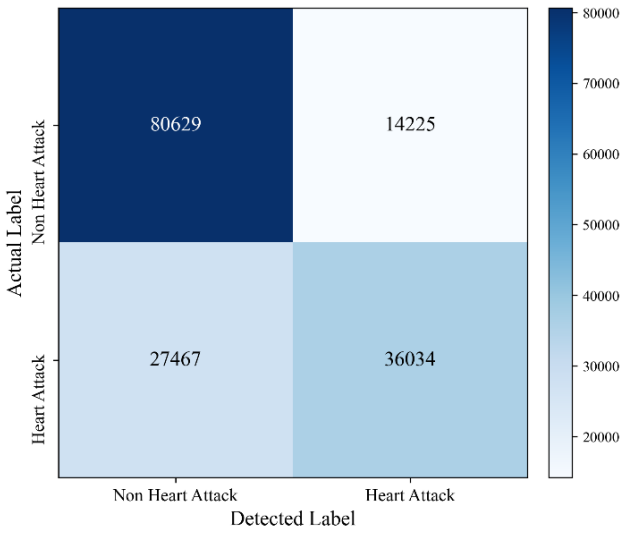
Pada skenario pertama, algoritma Random Forest digunakan sebagai model dasar tanpa penerapan teknik penyeimbangan data maupun reduksi dimensi. Model dibangun menggunakan hasil *preprocessing* awal berupa Ordinal Encoding untuk variabel kategorikal serta StandardScaler untuk menormalkan seluruh fitur numerik.

Proses penentuan parameter terbaik dilakukan menggunakan RandomizedSearchCV dengan 5-*Fold Cross -Validation* sebagai mekanisme evaluasi utama. Berdasarkan proses tuning tersebut, diperoleh konfigurasi hyperparameter terbaik, yaitu *n_estimators* = 600, *max_depth* = 10, *max_features* = *sqrt*, *min_samples_split* = 5, dan *min_samples_leaf* = 4. Konfigurasi ini menghasilkan nilai ROC-AUC rata-rata sebesar 0,8161 selama proses validasi silang 5-*Fold*. Hasil evaluasi lengkap per *Fold* ditunjukkan pada Tabel berikut:

Tabel 4. 2 Matriks Evaluasi 5-*Fold* Model Dasar Ordinal Encoding

Fold	**Akurasi**	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
Fold 1	73,63%	70,70%	56,85%	63,02%	81,35%
Fold 2	73,63%	72,47%	56,13%	63,26%	81,71%
Fold 3	73,41%	70,90%	56,67%	62,99%	81,47%
Fold 4	73,95%	72,97%	56,95%	63,97%	82,26%
Fold 5	73,26%	70,87%	56,30%	62,75%	81,25%
Mean	73,58%	71,58%	56,58%	63,20%	81,61%

Sebagai evaluasi tambahan, penelitian ini juga menggunakan metode *Out-of-Bag* (OOB) yang merupakan mekanisme evaluasi internal pada algoritma Random Forest. Berbeda dengan *cross -validation*, OOB memanfaatkan sampel data yang tidak terpilih pada proses bootstrap untuk melakukan evaluasi tanpa pembagian data secara eksplisit. Berdasarkan hasil prediksi OOB, diperoleh Confusion Matrix yang ditunjukkan pada Gambar 4.2.



Gambar 4. 2 Hasil Confusion Matrix (OOB) Model Dasar Ordinal Encoding

Berdasarkan Confusion Matrix OOB tersebut, model menghasilkan *True Positive* (TP) sebanyak 36034 data, *False Positive* (FP) sebanyak 14225 data, *True Negative* (TN) sebanyak 80629 data, dan *False negative* (FN) sebanyak 27467 data.

Nilai-nilai ini menunjukkan bahwa model mampu mengidentifikasi sebagian besar kasus pasien yang tidak mengalami serangan jantung dengan baik, namun masih terdapat kesalahan deteksi pada kelas pasien yang mengalami serangan jantung.

Berdasarkan evaluasi OOB, diperoleh nilai matrix evaluasi sebagaimana ditampilkan pada Tabel 4.3.

Tabel 4. 3 Matrix Evaluasi OOB Model Dasar Ordinal Encoding

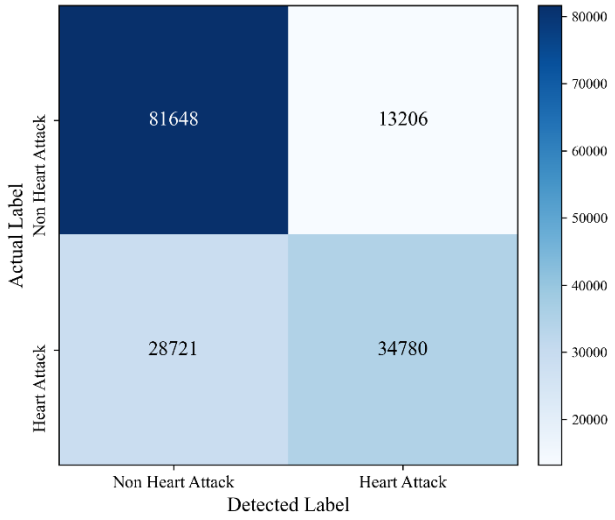
Akurasi	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
73.67%	71.69%	56.75%	63.30%	81.61%

Sebagai bagian dari uji coba awal, model dasar juga dibangun menggunakan teknik *preprocessing* lain, yaitu One-Hot Encoding (OHE). Proses tuning menghasilkan parameter terbaik pada *n_estimators* = 200, *max_depth* = 10, *max_features* = *sqrt*, *min_samples_split* = 10, dan *min_samples_leaf* = 4. Konfigurasi ini menghasilkan nilai ROC-AUC rata-rata sebesar 0,8161 selama proses validasi silang 5-*Fold*. Hasil evaluasi lengkap per *Fold* ditunjukkan pada Tabel berikut :

Tabel 4. 4 Matrix Evaluasi 5-*Fold* Model Dasar One-Hot Encoding

Fold	**Akurasi**	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
Fold 1	73,45%	71,94%	53,79%	61,55%	81,32%
Fold 2	73,41%	73,40%	53,72%	62,04%	81,69%
Fold 3	73,40%	72,05%	54,52%	62,07%	81,43%
Fold 4	74,05%	73,87%	55,85%	63,61%	82,23%
Fold 5	73,08%	71,15%	54,98%	62,03%	81,11%
Mean	73,48%	72,48%	54,57%	62,26%	81,55%

Sedangkan berdasarkan hasil deteksi evaluasi matriks OOB, diperoleh Confusion Matrix yang ditunjukkan pada Gambar 4.3.



Gambar 4. 3 Hasil Confusion Matrix (OOB) Model Dasar One-Hot Encoding

Pada Gambar 4.3 model menghasilkan *True Positive* (TP) = 34780, yaitu data pasien yang benar terdeteksi mengalami serangan jantung, serta *False Positive* (FP) = 13206, yaitu data pasien yang sebenarnya tidak mengalami serangan jantung namun terdeteksi sebagai positif. Selain itu, model menghasilkan *True Negative* (TN) = 81648, yaitu pasien yang benar terdeteksi tidak mengalami serangan jantung, serta *False negative* (FN) = 28721, yaitu pasien yang mengalami serangan jantung tetapi tidak terdeteksi oleh model. Berdasarkan Confusion Matrix tersebut diperoleh nilai matrik evaluasi performa model dasar menggunakan One-Hot Encoding.

Tabel 4. 5 Matrik Evaluasi (OOB) Skenario Model 1 One-Hot Encoding

Akurasi	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
73,52%	72,47%	54,77%	62,39%	81,54%

Dari tabel Matrix Evaluasi 4.3 di atas dapat diketahui skor akurasi, presisi, *Recall*, dan *F1-Score*, ROC-AUC yang didapat menggunakan model dasar.

4.2.2 Uji Coba Model dengan PCA

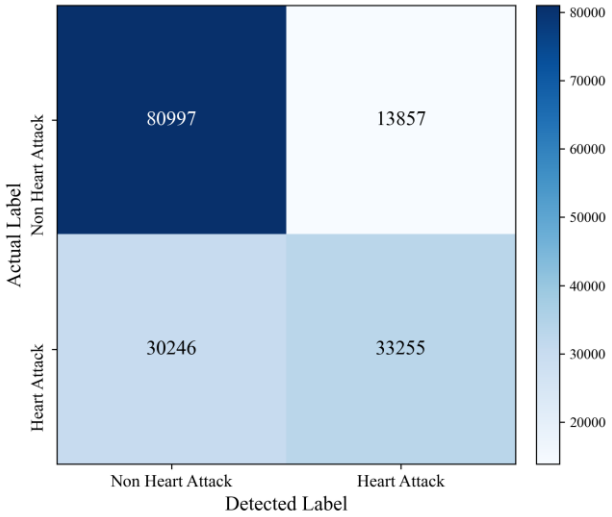
Pada skenario kedua, model Random Forest dikombinasikan dengan *Principal Component Analysis* (PCA) untuk mengurangi dimensi dan menghilangkan multikolinearitas antar fitur. Dari total 27 fitur awal, beberapa variabel menunjukkan korelasi yang cukup kuat, seperti *waist circumference* ↔ *obesity* (r = 0.395), sehingga PCA digunakan untuk mengekstraksi komponen yang lebih informatif dan bebas korelasi.

Proses tuning menggunakan RandomizedSearchCV dengan *5-Fold Cross - Validation* menghasilkan konfigurasi terbaik pada *n_estimators* = 600, *max_depth* = *None*, *max_features* = 3, *min_samples_split* = 10, dan *min_samples_leaf* = 4. Konfigurasi ini menghasilkan nilai ROC-AUC rata-rata sebesar 79,33% selama proses validasi silang 5-*Fold*. Hasil evaluasi lengkap per *Fold* ditunjukkan pada Tabel berikut:

Tabel 4. 6 Matriks Evaluasi 5-*Fold* Model 2 Ordinal Encoding

Fold	**Akurasi**	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
Fold 1	72,27%	71,10%	51,99%	60,07%	79,24%
Fold 2	72,15%	70,81%	51,98%	59,95%	79,35%
Fold 3	72,53%	71,38%	52,58%	60,56%	79,64%
Fold 4	71,90%	70,53%	51,42%	59,47%	79,06%
Fold 5	72,39%	71,32%	52,12%	60,22%	79,41%
Mean	72,25%	71,03%	52,02%	60,05%	79,34%

Sedangkan berdasarkan hasil deteksi evaluasi matriks OOB, diperoleh Confusion Matrix yang ditunjukkan pada Gambar 4.4.



Gambar 4. 4 Confusion Matrix Skenario Model 2 Ordinal Encoding

Pada gambar 4.4 model menghasilkan *True Positive* (TP) = 33255 data pasien terkena serangan jantung yang dideteksi sebagai pasien terkena serangan jantung (deteksi benar). *False Positive* (FP) = 13857 data pasien tidak terkena serangan jantung yang dideteksi sebagai pasien terkena serangan jantung (deteksi salah), *True Negative* (TN) = 80997 data pasien tidak terkena serangan jantung yang dideteksi sebagai pasien tidak terkena serangan jantung (deteksi benar), *False negative* (FN) = 30246 data pasien terkena serangan jantung yang dideteksi sebagai pasien tidak terkena serangan jantung (deteksi salah). Dari Confusion Matrix di atas dapat diketahui skor akurasi, presisi, *Recall*, dan *F1-Score*, ROC-AUC yang didapat model menggunakan PCA.

Tabel 4. 7 Matrik Evaluasi Skenario Model 2 Ordinal Encoding

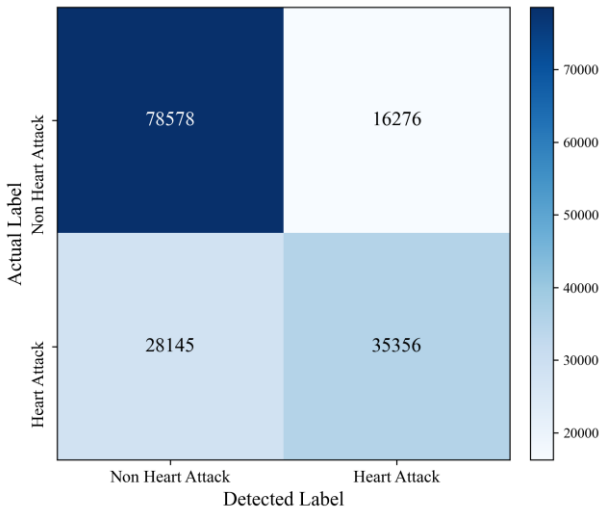
Akurasi	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
72,14%	70,58%	52,36%	60,12%	79,18%

Sebagai bagian dari uji coba lanjutan skenario 2, model juga dibangun menggunakan teknik *preprocessing* One-Hot Encoding (OHE) yang kemudian dilanjutkan dengan reduksi dimensi menggunakan Principal Component Analysis (PCA). Proses tuning menghasilkan parameter terbaik pada konfigurasi Random Forest dengan *n_estimators* = 600, *min_samples_split* = 5, *min_samples_leaf* = 2, *max_features* = 3, *max_depth* = *None*. Konfigurasi ini menghasilkan nilai ROC-AUC rata-rata sebesar 78,85% selama proses validasi silang 5-*Fold*. Hasil evaluasi lengkap per *Fold* ditunjukkan pada Tabel berikut:

Tabel 4. 8 Matriks Evaluasi 5-*Fold* Model 2 One-Hot Encoding

Fold	**Akurasi**	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
Fold 1	72,02%	68,79%	55,32%	61,32%	78,73%
Fold 2	71,92%	68,72%	55,03%	61,12%	78,87%
Fold 3	72,37%	69,33%	55,76%	61,81%	79,19%
Fold 4	71,83%	68,44%	55,22%	61,12%	78,66%
Fold 5	72,10%	68,72%	55,86%	61,63%	78,83%
Mean	72,05%	68,80%	55,44%	61,40%	78,86%

Sedangkan berdasarkan hasil deteksi evaluasi matriks OOB, diperoleh Confusion Matrix yang ditunjukkan pada Gambar 4.4.



Gambar 4. 5 Confusion Matrix (OOB) Model 2 One-Hot Encoding

Pada Gambar 4.5 ditunjukkan bahwa model menghasilkan *True Negative* (TN) = 78578, yaitu data pasien yang benar terdeteksi tidak mengalami serangan jantung, serta *False Positive* (FP) = 16276, yaitu pasien yang sebenarnya tidak mengalami serangan jantung namun terdeteksi sebagai positif. Selain itu, model menghasilkan *True Positive* (TP) = 35356, yaitu pasien yang benar terdeteksi mengalami serangan jantung, serta *False negative* (FN) = 28145, yaitu pasien yang mengalami serangan jantung tetapi tidak terdeteksi oleh model.

Tabel 4. 9 Matrix Evaluasi Model 2 One-Hot Encoding

Akurasi	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
71,94%	68,47%	55,67%	61,41%	78,71%

Dari tabel Matrix Evaluasi 4.9 di atas dapat diketahui skor akurasi, presisi, *Recall*, dan *F1-Score*, ROC-AUC yang didapat model menggunakan PCA.

4.2.3 Uji Coba Model dengan SMOTE

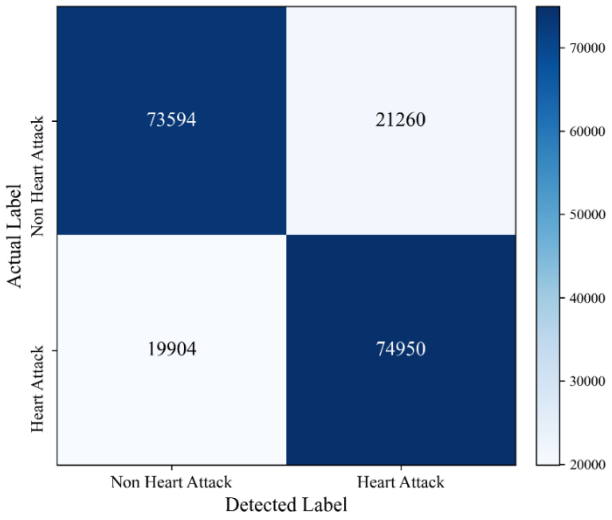
Pada skenario ketiga, dataset terlebih dahulu diproses menggunakan SMOTE untuk menyeimbangkan distribusi kelas, adapun hasil penyeimbangan data pada model ini bisa dilihat pada gambar 3.2.

Proses tuning menggunakan RandomizedSearchCV (*5-Fold cross -validation)* menghasilkan konfigurasi terbaik pada *n_estimators* = 200, *max_depth* = *None*, *max_features* = 5, *min_samples_split* = 2, dan *min_samples_leaf* = 1. Konfigurasi tersebut memberikan nilai ROC-AUC rata-rata sebesar 87,29% selama proses *cross validation* 5-*Fold*. Hasil evaluasi lengkap per *Fold* ditunjukkan pada Tabel berikut:

Tabel 4. 10 Matriks Evaluasi 5-*Fold* Model 3 Ordinal Encoding

Fold	**Akurasi**	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
Fold 1	78,27%	77,84%	79,03%	78,43%	87,62%
Fold 2	78,50%	78,47%	78,54%	78,51%	87,58%
Fold 3	77,94%	78,09%	77,67%	77,88%	87,22%
Fold 4	78,10%	78,04%	78,21%	78,13%	87,42%
Fold 5	78,48%	78,34%	78,71%	78,53%	87,56%
Mean	78,26%	78,16%	78,43%	78,29%	87,48%

Sedangkan berdasarkan hasil deteksi evaluasi matriks OOB, diperoleh Confusion Matrix yang ditunjukkan pada Gambar 4.6.



Gambar 4. 6 Confusion Matrix Skenario Model 3 Ordinal Encoding

Pada gambar 4.6 dibawah, Model menghasilkan *True Positive* (TP) = 74950 data pasien terkena serangan jantung yang dideteksi sebagai pasien terkena serangan jantung (deteksi benar). *False Positive* (FP) = 21260 data pasien tidak terkena serangan jantung yang dideteksi sebagai pasien terkena serangan jantung (deteksi salah), *True Negative* (TN) = 73594 data pasien tidak terkena serangan jantung yang dideteksi sebagai pasien tidak terkena serangan jantung (deteksi

benar), *False negative* (FN) = 19904 data pasien terkena serangan jantung yang dideteksi sebagai pasien tidak terkena serangan jantung (deteksi salah).

Tabel 4. 11 Matrik Evaluasi Skenario Model 3 Ordinal Encoding

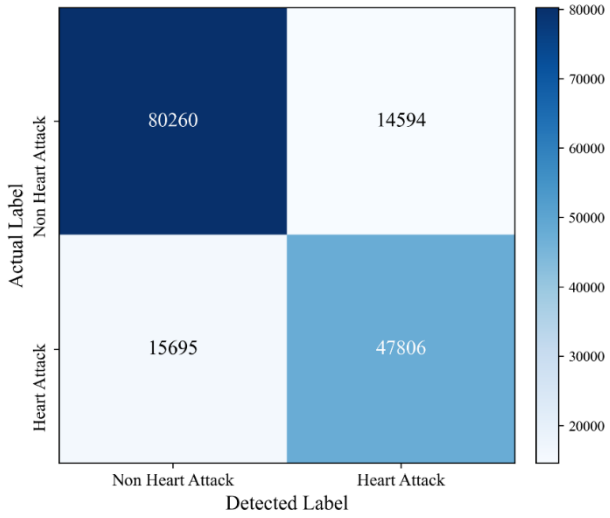
Akurasi	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
78,30%	77,90%	79,02%	78,46%	87,55%

Sebagai bagian dari uji coba lanjutan skenario 3, model juga dibangun menggunakan teknik *preprocessing* One-Hot Encoding (OHE). Proses tuning menghasilkan parameter terbaik pada *n_estimators* = 200, *min_samples_split* = 2, *min_samples_leaf* = 1, *max_features* = 5, *max_depth = None*. Konfigurasi tersebut memberikan nilai ROC-AUC rata-rata sebesar 87,63% selama proses *cross validation* 5-*Fold*. Hasil evaluasi lengkap per *Fold* ditunjukkan pada Tabel berikut:

Tabel 4. 12 Matriks Evaluasi 5-*Fold* Model 3 One-Hot Encoding

Fold	**Akurasi**	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
Fold 1	78,46%	78,44%	78,50%	78,47%	87,78%
Fold 2	78,61%	79,04%	77,86%	78,45%	87,66%
Fold 3	77,88%	78,23%	77,24%	77,74%	87,37%
Fold 4	78,36%	78,63%	77,90%	78,26%	87,61%
Fold 5	78,38%	78,71%	77,81%	78,25%	87,75%
Mean	78,34%	78,61%	77,86%	78,23%	87,63%

Sedangkan berdasarkan hasil deteksi evaluasi matriks OOB, diperoleh Confusion Matrix yang ditunjukkan pada Gambar 4.7.



Gambar 4. 7 Confusion Matrix (OOB) Model 3 One-Hot Encoding

Pada Gambar 4.7 terlihat bahwa model menghasilkan *True Negative* (TN) = 80.260, yaitu data pasien yang benar terdeteksi tidak mengalami serangan jantung, serta *False Positive* (FP) = 14.594, yaitu pasien yang sebenarnya tidak mengalami serangan jantung namun terdeteksi sebagai positif. Selain itu, model menghasilkan *True Positive* (TP) = 47.806, yaitu pasien yang benar terdeteksi mengalami serangan jantung, serta *False negative* (FN) = 15.695, yaitu data pasien yang mengalami serangan jantung tetapi tidak terdeteksi oleh model..

Tabel 4. 13 Matrix Evaluasi Model 3 One-Hot Encoding

Akurasi	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
80,87%	76,61%	75,28%	75,94%	89,56%

Dari tabel Matrix Evaluasi 4.11 di atas dapat diketahui skor akurasi, presisi, *Recall*, dan *F1-Score*, ROC-AUC yang didapat model menggunakan SMOTE dengan teknik *preprocessing* One-Hot Encoding.

4.2.4 Uji Coba Model dengan PCA dan SMOTE

Pada skenario keempat, dataset terlebih dahulu diproses menggunakan *Principal Component Analysis* (PCA) untuk mereduksi dimensi dan menghilangkan multikolinearitas antar fitur. PCA menurunkan jumlah fitur dari 27 fitur awal menjadi 26 komponen utama, dengan variansi kumulatif sebesar 97,76%, sehingga sebagian besar informasi penting pada dataset tetap terjaga.

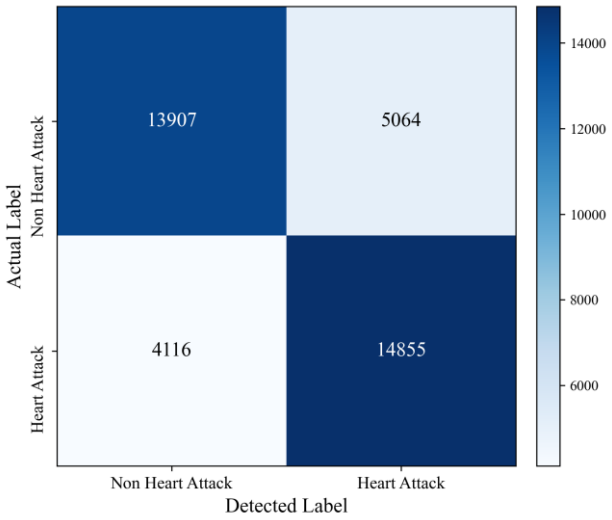
Setelah proses PCA, dilakukan penyeimbangan kelas menggunakan teknik SMOTE (*Synthetic Minority Oversampling Technique*). Teknik ini menambah sampel sintetis pada kelas *Heart Attack* sehingga distribusi kelas lebih seimbang. Jumlah sampel meningkat dari 158.355 menjadi 189.708 setelah dilakukan SMOTE, sehingga model memiliki representasi yang lebih baik terhadap pola pada kelas minoritas.

Proses tuning menggunakan RandomizedSearchCV (*5-Fold cross - validation)* menghasilkan konfigurasi terbaik pada *n_estimators* = 800, *max_depth* = *None*, *max_features* = 3, *min_samples_split* = 2, dan *min_samples_leaf* = 2. Konfigurasi tersebut memberikan nilai ROC-AUC rata-rata sebesar 79,18% selama proses *cross validation* 5-*Fold*. Hasil evaluasi lengkap per *Fold* ditunjukkan pada Tabel berikut:

Tabel 4. 14 Matriks Evaluasi 5-*Fold* Model 4 Ordinal Encoding

Fold	**Akurasi**	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
Fold 1	71,71%	63,94%	67,59%	65,71%	79,21%
Fold 2	71,66%	64,08%	66,72%	65,37%	79,08%
Fold 3	71,71%	63,96%	67,46%	65,67%	79,58%
Fold 4	71,18%	63,41%	66,51%	64,92%	78,90%
Fold 5	71,91%	64,38%	67,02%	65,68%	79,31%
Mean	71,63%	63,95%	67,06%	65,47%	79,22%

Sedangkan berdasarkan hasil deteksi evaluasi matriks OOB, diperoleh Confusion Matrix yang ditunjukkan pada Gambar 4.8.



Gambar 4. 8 Confusion Matrix Skenario Model 4 Ordinal Encoding

Pada gambar 4.8 model menghasilkan *True Positive* (TP) = 14855 data pasien terkena serangan jantung yang dideteksi sebagai pasien terkena serangan jantung (deteksi benar). *False Positive* (FP) = 5064 data pasien terkena serangan jantung yang didetetksi sebagai pasien terkena serangan jantung (deteksi salah), *True Negative* (TN) = 13907 data pasien tidak terkena serangan jantung yang dideteksi sebagai pasien tidak terkena serangan jantung (deteksi benar), *False negative* (FN) = 4116 data pasien terkena serangan jantung yang dideteksi sebagai pasien tidak terkena serangan jantung (deteksi salah).

Tabel 4. 15 Matrik Evaluasi Skenario Model 4 Ordinal Encoding

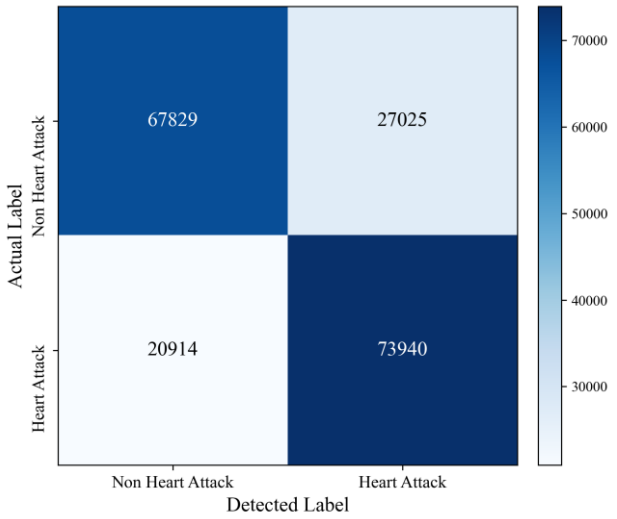
Akurasi	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
75,81%	74,58%	78,30%	76,39%	84,46%

Sebagai bagian dari uji coba lanjutan, model juga dibangun menggunakan teknik *preprocessing* One-Hot Encoding (OHE) yang kemudian dilanjutkan dengan penerapan metode PCA dan SMOTE. Proses tuning menghasilkan parameter terbaik pada *n_estimators* = 600, *min_samples_split* = 5, *min_samples_leaf* = 2, *max_features* = "*sqrt*", *max_depth* = *None*. Konfigurasi tersebut memberikan nilai ROC-AUC rata-rata sebesar 76,70% selama proses *cross validation* 5-*Fold*. Hasil evaluasi lengkap per *Fold* ditunjukkan pada Tabel berikut:

Tabel 4. 16 Matriks Evaluasi 5- *Fold* Model 4 One-Hot Encoding

Fold	**Akurasi**	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
Fold 1	69,12%	60,67%	65,34%	62,92%	76,07%
Fold 2	69,80%	61,43%	66,35%	63,79%	76,85%
Fold 3	69,01%	60,49%	65,46%	62,88%	76,13%
Fold 4	70,07%	61,79%	66,49%	64,05%	77,80%
Fold 5	69,37%	60,81%	66,42%	63,49%	76,38%
Mean	69,47%	61,04%	66,01%	63,43%	76,65%

Sedangkan berdasarkan hasil deteksi evaluasi matriks OOB, diperoleh Confusion Matrix yang ditunjukkan pada Gambar 4.9.



Gambar 4. 9 Confusion Matrix Skenario Model 4 One-Hot Encoding

Pada gambar 4.9 model menghasilkan *True Negative* (TN) = 67829, yaitu pasien yang benar terdeteksi tidak mengalami serangan jantung, serta *False Positive* (FP) = 27025, yaitu pasien yang sebenarnya tidak mengalami serangan jantung namun terdeteksi sebagai positif. Selain itu, model menghasilkan *True Positive* (TP) = 73940, yaitu pasien yang benar terdeteksi mengalami serangan jantung, serta *False negative* (FN) = 20914, yaitu pasien yang mengalami serangan jantung tetapi tidak terdeteksi oleh model. Nilai-nilai ini mencerminkan bahwa model mampu mempertahankan keseimbangan performa antara presisi dan *Recall* setelah penerapan PCA dan SMOTE.

Tabel 4. 17 Matrix Evaluasi Skenario OOB Model 4 One-Hot Encoding

Akurasi	**Presisi**	***Recall***	***F1-Score***	**ROC-AUC**
74,73%	73,23%	77,95%	75,52%	83,17%

Dari tabel Matrix Evaluasi 4.17 di atas dapat diketahui skor akurasi, presisi, *Recall*, dan *F1-Score*, ROC-AUC yang didapat model menggunakan PCA dan SMOTE dengan teknik *preprocessing* One-Hot Encoding.

4.3 Pembahasan Analisis Skenario Model

Pada sub bab ini akan dibahas mengenai hasil uji coba dari implementasi Random Forest yang dikombinasikan dengan metode penyeimbangan data SMOTE serta korelasi antar fitur menggunakan *Principal Component Analysis* (PCA). Empat skenario pengujian dibandingkan untuk mengevaluasi dampak penyeimbangan data dan korelasi antar fitur terhadap performa model.

4.3.1 Analisis Skenario Model 1

Berikut merupakan tabel Analisa skenario model 1 menggunakan dua teknik *preprocessing* yang berbeda yakni Ordinal Encoding dan One-Hot Encoding

Tabel 4. 18 Analisa Skenario Model 1

Analisa Skenario Model 1			
Preprocessing		**Ordinal Encoding**	**One-Hot Encoding**
Best Parameter	*n_estimators*	600	200
	max_depth	10	10
	max_features	sqrt	sqrt
	min_samples_split	5	10
	min_samples_leaf	4	4
Confusion Matrix	*True Positive* (TP)	36.034	34.780
	False Positive (FP)	14.225	13.206
	True Negative (TN)	80.629	81.648
	False negative (FN)	27.467	28.721
Matriks Evaluasi (*Mean 5-Fold*)	Akurasi	73,58%	73,48%
	Presisi	71,58%	72,48%
	Recall	56,58%	54,57%
	F1-Score	63,20%	62,26%
	ROC-AUC	81,61%	81,55%
Matriks Evaluasi (OOB)	Akurasi	73,67%	73,52%
	Presisi	71,69%	72,47%
	Recall	56,75%	54,77%
	F1-Score	63,30%	62,39%
	ROC-AUC	81,61%	81,54%

Berdasarkan hasil evaluasi *Mean 5-Fold cross -validation*, kedua pendekatan *preprocessing* menghasilkan nilai akurasi yang relatif serupa, yaitu 73,58% pada Ordinal Encoding dan 73,48% pada One-Hot Encoding. Meskipun nilai akurasi tersebut tergolong cukup tinggi, metrik ini belum sepenuhnya mencerminkan kemampuan model dalam mendeteksi kasus serangan jantung secara optimal. Hal ini disebabkan oleh kondisi data yang tidak seimbang, sehingga

model cenderung lebih fokus mengenali kelas mayoritas. Kondisi tersebut tercermin pada nilai *Recall* yang masih tergolong rendah, yaitu 56,58% untuk Ordinal Encoding dan 54,57% untuk One-Hot Encoding, yang menunjukkan bahwa masih terdapat sejumlah besar pasien yang mengalami serangan jantung namun tidak berhasil terdeteksi oleh model.

Hasil evaluasi tambahan menggunakan metode *Out-of-Bag* (OOB) menunjukkan pola performa yang konsisten dengan hasil validasi silang. Pada Ordinal Encoding, model menghasilkan nilai akurasi sebesar 73,67% dengan *Recall* 56,75%, sedangkan pada One-Hot Encoding diperoleh akurasi 73,52% dan *Recall* 54,77%. Konsistensi antara hasil 5-*Fold Cross -Validation* dan OOB mengindikasikan bahwa performa model cukup stabil, namun tetap menunjukkan keterbatasan dalam mengenali kelas minoritas. Hal ini sejalan dengan temuan Septiana Rizky *et al*. (2022) yang menyatakan bahwa model klasifikasi tanpa penanganan ketidakseimbangan data cenderung mengalami bias terhadap kelas mayoritas.

4.3.2 Analisis Skenario Model 2

Berikut merupakan tabel Analisa skenario model 2 menggunakan dua teknik *preprocessing* yang berbeda yakni Ordinal Encoding dan One-Hot Encoding.

Tabel 4. 19 Analisis Skenario Model 2

Analisa Skenario Model 2			
Preprocessing		**Ordinal Encoding**	**One-Hot Encoding**
Best Parameter	*n_estimators*	600	600
	max_depth	None	None
	max_features	3	3
	min_samples_split	10	5
	min_samples_leaf	4	2
Confusion Matrix	*True Positive* (TP)	33.255	35.356
	False Positive (FP)	13.857	16.276
	True Negative (TN)	80.997	78.578
	False negative (FN)	30.246	28.145
Matriks Evaluasi (*Mean 5-Fold*)	Akurasi	72,25%	72,05%
	Presisi	71,03%	68,80%
	Recall	52,02%	55,44%
	F1-Score	60,05%	61,40%
	ROC-AUC	79,34%	78,86%
Matriks Evaluasi (OOB)	Akurasi	72,14%	71,94%
	Presisi	70,58%	68,47%
	Recall	52,36%	55,67%
	F1-Score	60,12%	61,41%
	ROC-AUC	79,18%	78,71%

Berdasarkan hasil evaluasi *Mean 5-Fold Cross -Validation* menggunakan Ordinal Encoding + PCA, model menghasilkan nilai akurasi sebesar 72,25% dengan ROC-AUC 79,34%. Jika dibandingkan dengan model dasar tanpa PCA, terjadi penurunan nilai ROC-AUC yang cukup signifikan. Penurunan ini menunjukkan bahwa proses reduksi dimensi melalui PCA menyebabkan sebagian informasi diskriminatif yang penting dalam membedakan kelas *Heart Attack* dan *Non Heart Attack* ikut tereduksi. Kondisi tersebut tercermin pada nilai *Recall* sebesar 52,02%, yang menunjukkan bahwa lebih dari separuh kasus serangan jantung belum berhasil terdeteksi oleh model. Meskipun demikian, nilai *F1-Score*

sebesar 60,05% menunjukkan keseimbangan yang relatif stabil antara presisi dan *Recall*.

Hasil evaluasi tambahan menggunakan metode *Out-of-Bag* (OOB) pada Ordinal Encoding + PCA menunjukkan pola performa yang konsisten dengan hasil validasi silang. Model menghasilkan nilai akurasi sebesar 72,14% dengan ROC-AUC 79,18%, serta nilai *Recall* sebesar 52,36%. Konsistensi antara hasil 5-*Fold Cross -Validation* dan OOB mengindikasikan bahwa performa model cukup stabil dan tidak mengalami overfitting, namun tetap menunjukkan keterbatasan dalam mendeteksi kelas minoritas.

Sebagai bagian dari uji coba lanjutan pada skenario ini, model juga dibangun menggunakan One-Hot Encoding (OHE) yang dikombinasikan dengan PCA. Berdasarkan hasil *Mean 5-Fold cross -validation*, pendekatan ini menghasilkan nilai akurasi sebesar 72,05% dengan ROC-AUC 78,86%. Meskipun nilai akurasi dan ROC-AUC sedikit lebih rendah dibandingkan Ordinal Encoding + PCA, model dengan OHE + PCA menunjukkan nilai *Recall* yang lebih tinggi, yaitu 55,44%, yang mengindikasikan kemampuan yang lebih baik dalam mendeteksi kasus serangan jantung.

Hasil evaluasi OOB pada OHE + PCA memperkuat temuan tersebut, dengan nilai *Recall* sebesar 55,67% dan *F1-Score* 61,41%. Peningkatan *Recall* ini menunjukkan bahwa representasi fitur kategorikal melalui OHE, meskipun mengalami reduksi dimensi akibat PCA, tetap mampu mempertahankan informasi penting yang mendukung deteksi kelas *Heart Attack*. Namun demikian, nilai presisi

yang relatif lebih rendah menunjukkan adanya peningkatan kesalahan prediksi positif, sehingga masih terdapat *trade-off* antara sensitivitas dan ketepatan prediksi.

4.3.3 Analisis Skenario Model 3

Berikut merupakan tabel Analisa skenario model 3 menggunakan dua teknik *preprocessing* yang berbeda yakni Ordinal Encoding dan One-Hot Encoding.

Tabel 4. 20 Analisis Skenario Model 3

Analisa Skenario Model 3			
Preprocessing		**Ordinal Encoding**	**One-Hot Encoding**
Best Parameter	*n_estimators*	200	200
	max_depth	None	None
	max_features	5	5
	min_samples_split	2	2
	min_samples_leaf	1	1
Confusion Matrix	*True Positive* (TP)	74.950	47.806
	False Positive (FP)	21.260	14.594
	True Negative (TN)	73.594	80.260
	False negative (FN)	19.904	15.695
Matriks Evaluasi (*Mean 5-Fold*)	Akurasi	78,26%	78,34%
	Presisi	78,16%	78,61%
	Recall	78,43%	77,86%
	F1-Score	78,29%	78,23%
	ROC-AUC	87,48%	87,63%
Matriks Evaluasi (OOB)	Akurasi	78,30%	80,87%
	Presisi	77,90%	76,61%
	Recall	79,02%	75,28%
	F1-Score	78,46%	75,94%
	ROC-AUC	87,55%	89,56%

Berdasarkan hasil evaluasi *Mean 5-Fold Cross -Validation* menggunakan Ordinal Encoding + SMOTE, model menghasilkan nilai akurasi sebesar 78,26% dengan ROC-AUC 87,48%, yang menunjukkan peningkatan signifikan

dibandingkan model dasar dan model dengan PCA. Peningkatan performa ini terutama terlihat pada nilai *Recall* sebesar 78,43%, yang menandakan bahwa sebagian besar kasus serangan jantung berhasil terdeteksi oleh model. Hal ini menunjukkan bahwa penyeimbangan data melalui SMOTE secara efektif mengurangi bias model terhadap kelas mayoritas dan meningkatkan sensitivitas model terhadap kelas minoritas. Nilai *F1-Score* sebesar 78,29% juga mengindikasikan keseimbangan yang baik antara presisi dan *Recall*.

Hasil evaluasi tambahan menggunakan metode *Out-of-Bag* (OOB) pada Ordinal Encoding + SMOTE menunjukkan performa yang konsisten dengan hasil validasi silang. Model menghasilkan nilai akurasi 78,30%, *Recall* 79,02%, dan ROC-AUC 87,55%, yang menegaskan bahwa peningkatan performa yang diperoleh bukan merupakan hasil overfitting, melainkan peningkatan kemampuan generalisasi model dalam mendeteksi kasus serangan jantung.

Sebagai bagian dari uji coba lanjutan pada skenario ini, model juga dibangun menggunakan One-Hot Encoding (OHE) yang dikombinasikan dengan SMOTE. Berdasarkan hasil *Mean 5-Fold cross -validation*, pendekatan ini menghasilkan nilai akurasi sebesar 78,34% dengan ROC-AUC 87,63%. Meskipun nilai akurasi dan *F1-Score* relatif sebanding dengan Ordinal Encoding + SMOTE, pendekatan OHE + SMOTE menunjukkan nilai presisi yang sedikit lebih tinggi, yang mengindikasikan bahwa prediksi positif yang dihasilkan cenderung lebih akurat.

Hasil evaluasi OOB pada OHE + SMOTE menunjukkan peningkatan performa yang paling signifikan, dengan nilai akurasi mencapai 80,87% dan ROC-

AUC sebesar 89,56%, yang merupakan nilai tertinggi dibandingkan seluruh skenario sebelumnya. Meskipun nilai *Recall* pada OHE + SMOTE (75,28%) sedikit lebih rendah dibandingkan Ordinal Encoding + SMOTE, peningkatan nilai ROC-AUC menunjukkan bahwa model memiliki kemampuan diskriminasi yang sangat baik dalam membedakan kelas *Heart Attack* dan *Non Heart Attack* secara keseluruhan.

4.3.4 Analisis Skenario Model 4

Berikut merupakan tabel Analisa skenario model 4 menggunakan dua teknik *preprocessing* yang berbeda yakni Ordinal Encoding dan One-Hot Encoding.

Tabel 4. 21 Analisis Skenario Model 4

Analisa Skenario Model 4			
Preprocessing		**Ordinal Encoding**	**One-Hot Encoding**
Best Parameter	*n_estimators*	800	600
	max_depth	None	None
	max_features	3	sqrt
	min_samples_split	2	5
	min_samples_leaf	2	2
Confusion Matrix	*True Positive* (TP)	14.855	73.940
	False Positive (FP)	5.064	27.025
	True Negative (TN)	13.907	67.829
	False negative (FN)	4.116	20.914
Matriks Evaluasi (*Mean 5-Fold*)	Akurasi	71,63%	69,47%
	Presisi	63,95%	61,04%
	Recall	67,06%	66,01%
	F1-Score	65,47%	63,43%
	ROC-AUC	79,22%	76,65%
Matriks Evaluasi (OOB)	Akurasi	75,81%	74,73%
	Presisi	74,58%	73,23%

Analisa Skenario Model 4			
Preprocessing		**Ordinal Encoding**	**One-Hot Encoding**
Matriks Evaluasi (OOB)	*Recall*	78,30%	77,95%
	F1-Score	76,39%	75,52%
	ROC-AUC	84,46%	83,17%

Berdasarkan hasil *Mean 5-Fold Cross -Validation* menggunakan Ordinal Encoding + PCA + SMOTE, model menghasilkan nilai akurasi sebesar 71,63% dengan ROC-AUC 79,22%. Jika dibandingkan dengan skenario Model 3 (SMOTE tanpa PCA), performa ini mengalami penurunan, khususnya pada nilai ROC-AUC. Hal ini mengindikasikan bahwa meskipun PCA mampu mengurangi dimensi dan kompleksitas data, proses reduksi tersebut juga menyebabkan hilangnya sebagian informasi diskriminatif yang penting bagi model dalam membedakan kelas *Heart Attack* dan *Non Heart Attack*. Namun demikian, nilai *Recall* sebesar 67,06% menunjukkan bahwa model masih mampu mendeteksi sebagian besar kasus serangan jantung dengan cukup baik.

Hasil evaluasi *Out-of-Bag* (OOB) pada Ordinal Encoding + PCA + SMOTE menunjukkan peningkatan performa dibandingkan hasil validasi silang, dengan akurasi mencapai 75,81% dan *Recall* 78,30%. Nilai ini mengindikasikan bahwa kombinasi PCA dan SMOTE mampu meningkatkan sensitivitas model terhadap kelas minoritas ketika dievaluasi menggunakan data OOB, meskipun peningkatan tersebut belum mampu melampaui performa terbaik pada skenario Model 3.

Pada pendekatan One-Hot Encoding + PCA + SMOTE, hasil *Mean 5-Fold Cross -Validation* menunjukkan nilai akurasi sebesar 69,47% dengan ROC-AUC 76,65%, yang merupakan nilai terendah dibandingkan skenario lainnya. Meskipun

nilai *Recall* relatif tinggi (66,01%), penurunan nilai presisi menyebabkan performa keseluruhan model menjadi kurang optimal. Hal ini menunjukkan bahwa kombinasi OHE dengan PCA menghasilkan ruang fitur yang kurang efisien setelah reduksi dimensi, sehingga kemampuan model dalam menangkap pola penting menjadi terbatas.

Hasil evaluasi OOB pada OHE + PCA + SMOTE menunjukkan performa yang lebih baik dibandingkan validasi silang, dengan akurasi 74,73% dan *Recall* 77,95%. Namun, nilai ROC-AUC (83,17%) masih berada di bawah performa terbaik yang dicapai pada skenario Model 3. Dengan demikian, dapat disimpulkan bahwa kombinasi PCA dan SMOTE tidak memberikan peningkatan performa yang lebih baik dibandingkan penerapan SMOTE tanpa PCA.

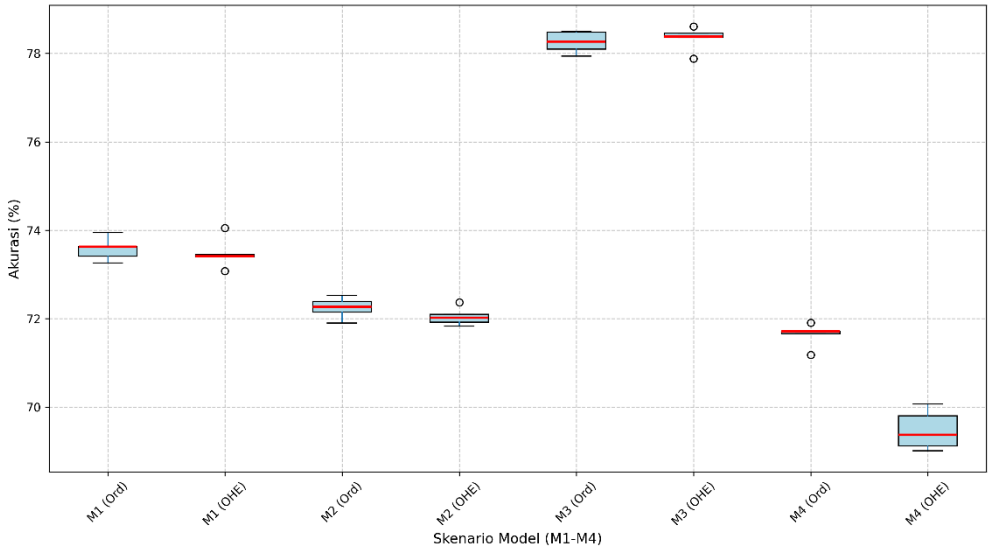
4.3.5 Analisis Skenario Model Berdasarkan Matriks Evaluasi

Untuk memahami secara mendalam dampak dari pendekatan yang digunakan, perlu dilakukan analisis secara terpisah terhadap lima metrik evaluasi utama, yaitu akurasi, presisi, *Recall*, *F1-Score*, dan ROC-AUC. Evaluasi 5-*Fold Cross -Validation* divisualisasikan menggunakan Boxplot untuk menggambarkan distribusi dan variabilitas performa model pada setiap skenario.

Selain itu, uji t berpasangan (*Paired T-Test*) dilakukan untuk menguji signifikansi statistik perbedaan performa antar skenario model, dengan menampilkan dua perbandingan yang memiliki nilai t-statistik absolut tertinggi per metrik sebagai representasi perbedaan paling signifikan. Evaluasi *Out-of-Bag* (OOB) disajikan dalam bentuk diagram batang sebagai validasi tambahan.

4.3.5.1 Akurasi

Boxplot berikut menyajikan distribusi nilai akurasi dari 5-*Fold Cross - Validation* pada tiap skenario.



Gambar 4. 10 Boxplot Distribusi Nilai Akurasi Seluruh Model

Berdasarkan visualisasi Boxplot, Model 1 pada kedua teknik encoding menunjukkan distribusi akurasi yang sempit dan stabil, dengan rata-rata 73,58% ± 0,23 pada Ordinal Encoding dan 73,48% ± 0,32 pada OHE. Model 2 menunjukkan nilai yang sedikit lebih rendah, yaitu 72,25% ± 0,21 (Ordinal) dan 72,05% ± 0,19 (OHE), mengindikasikan bahwa penerapan PCA tanpa penanganan ketidakseimbangan kelas justru menurunkan akurasi secara nyata. Peningkatan paling signifikan terjadi pada Model 3 (RF + SMOTE), dengan rata-rata akurasi mencapai 78,26% ± 0,22 (Ordinal) dan 78,34% ± 0,25 (OHE), disertai sebaran antar-*Fold* yang sangat sempit.

Sebaliknya, Model 4 (RF + SMOTE + PCA) mengalami penurunan akurasi menjadi 71,63% ± 0,24 (Ordinal) dan 69,47% ± 0,40 (OHE), bahkan lebih rendah

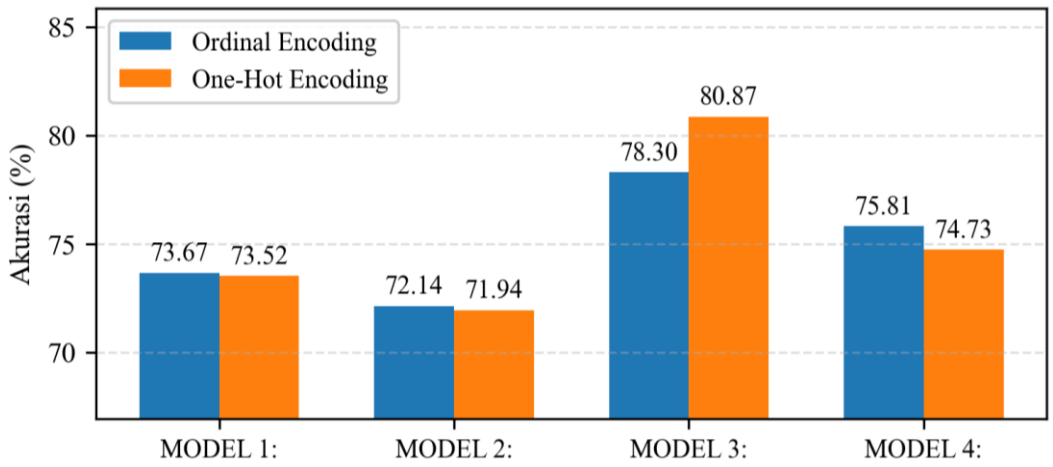
dari baseline Model 1, mengindikasikan bahwa PCA setelah SMOTE merusak informasi penting terutama pada OHE. Dua perbandingan dengan perbedaan akurasi paling signifikan secara statistik disajikan pada tabel berikut.

Tabel 4. 22 Perbandingan *Paired T-Test* Akurasi

Perbandingan	t-Statistik	p-Value	Signifikansi
M1 vs M2 (OHE)	6.4267	3.01×10^{-3}	** ($p < 0.01$)
M1 vs M3 (Ord)	-26.3241	1.24×10^{-5}	*** ($p < 0.001$)
M1 vs M4 (OHE)	25.3866	1.43×10^{-5}	*** ($p < 0.001$)
M2 vs M3 (Ord)	-37.3371	3.07×10^{-6}	*** ($p < 0.001$)
M2 vs M4 (Ord)	9.1570	7.90×10^{-4}	*** ($p < 0.001$)
M3 vs M4 (Ord)	54.4012	6.84×10^{-7}	*** ($p < 0.001$)

Secara keseluruhan, seluruh pasangan perbandingan menunjukkan perbedaan yang signifikan secara statistik dengan nilai $p < 0,01$ hingga $p < 0,001$. Di antara semua pasangan yang diuji, perbandingan M3 vs M4 menghasilkan nilai t-statistik tertinggi ($t = 54,40$, $p < 0,001$), menegaskan bahwa perbedaan akurasi antara skenario RF + SMOTE tanpa PCA dibandingkan RF + SMOTE + PCA merupakan perbedaan yang paling besar dan paling konsisten. Hal ini memberikan bukti empiris yang kuat bahwa penerapan PCA setelah SMOTE justru menurunkan performa model secara nyata, kemungkinan karena PCA mereduksi variansi yang penting untuk membedakan kelas minoritas yang telah dibentuk oleh SMOTE.

Skenario RF + SMOTE tanpa PCA dibandingkan RF + SMOTE + PCA merupakan perbedaan yang paling besar dan paling konsisten. Hal ini memberikan bukti empiris yang kuat bahwa penerapan PCA setelah SMOTE justru menurunkan performa model secara nyata, kemungkinan karena PCA mereduksi variansi yang penting untuk membedakan kelas minoritas yang telah dibentuk oleh SMOTE.

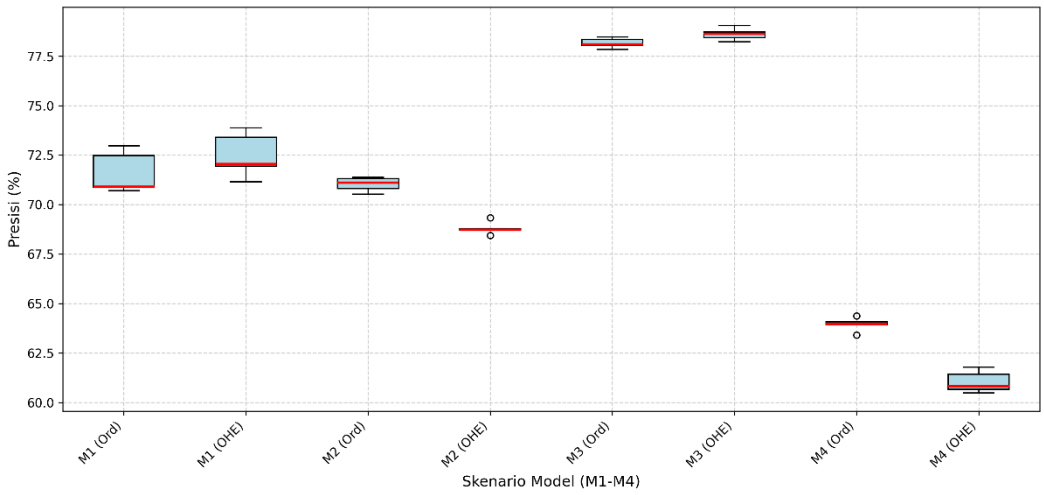


Gambar 4. 11 Diagram Distribusi Nilai Akurasi Seluruh Model (OOB)

Hasil OOB memperkuat temuan evaluasi 5-*Fold*. Pada Model 3, akurasi OOB mencapai 78,30% untuk Ordinal Encoding dan 80,87% untuk OHE, nilai tertinggi di antara seluruh skenario. Model 4 kembali menunjukkan penurunan akurasi OOB, terutama pada OHE, mengonfirmasi konsistensi dampak negatif PCA pada kedua metode evaluasi.

4.3.5.2 Presisi

Selain akurasi, presisi juga digunakan untuk mengevaluasi kemampuan model dalam mendeteksi kelas positif secara tepat. Boxplot berikut menampilkan distribusi nilai presisi dari 5-*Fold Cross -Validation* pada empat skenario model.



Gambar 4. 12 Boxplot Distribusi Nilai Presisi Seluruh Model

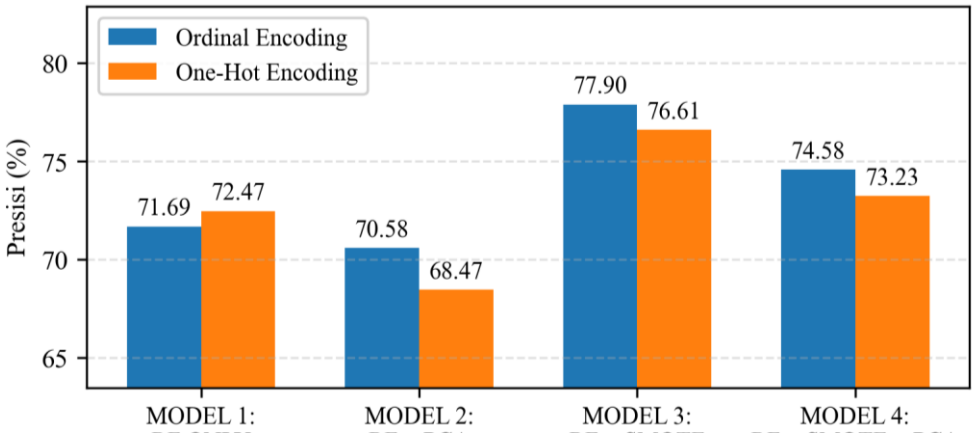
Berdasarkan visualisasi Boxplot, Model 1 menunjukkan rata-rata presisi sebesar 71,58% ± 0,95 (Ordinal) dan 72,48% ± 1,00 (OHE). Model 2 memperlihatkan nilai 71,03% ± 0,32 (Ordinal) dan 68,80% ± 0,29 (OHE), di mana perbedaan terhadap Model 1 tidak signifikan pada Ordinal Encoding (t = 0,89, p = 0,424), namun signifikan pada OHE (t = 6,30, p = 0,003), mengindikasikan bahwa PCA tidak selalu memberikan dampak yang berarti terhadap presisi. Pada Model 3, presisi meningkat signifikan menjadi 78,16% ± 0,22 (Ordinal) dan 78,61% ± 0,27 (OHE). Model 4 mengalami penurunan drastis menjadi 63,95% ± 0,31 (Ordinal) dan 61,04% ± 0,49 (OHE), bahkan lebih rendah dari baseline Model 1. Dua perbandingan dengan perbedaan presisi paling signifikan secara statistik disajikan pada tabel berikut.

Tabel 4. 23 Perbandingan *Paired T-Test* Presisi

Perbandingan	t-Statistik	p-Value	Signifikansi
M1 vs M2 (OHE)	6.3019	3.24×10^{-3}	** ($p < 0.01$)
M1 vs M3 (Ord)	-14.5300	1.30×10^{-4}	*** ($p < 0.001$)
M1 vs M4 (OHE)	36.7093	3.29×10^{-6}	*** ($p < 0.001$)
M2 vs M3 (OHE)	-38.6179	2.69×10^{-6}	*** ($p < 0.001$)
M2 vs M4 (Ord)	61.3948	4.22×10^{-7}	*** ($p < 0.001$)

Perbandingan	t-Statistik	p-Value	Signifikansi
M3 vs M4 (Ord)	103.9276	5.14×10^{-8}	*** ($p < 0.001$)

Berdasarkan tabel tersebut, perbandingan M3 vs M4 menghasilkan nilai t-statistik tertinggi di antara seluruh perbandingan (Ordinal: t = 103,93), menegaskan bahwa penurunan presisi akibat PCA setelah SMOTE merupakan perbedaan yang paling ekstrem dan paling konsisten, jauh melampaui perbandingan lainnya.



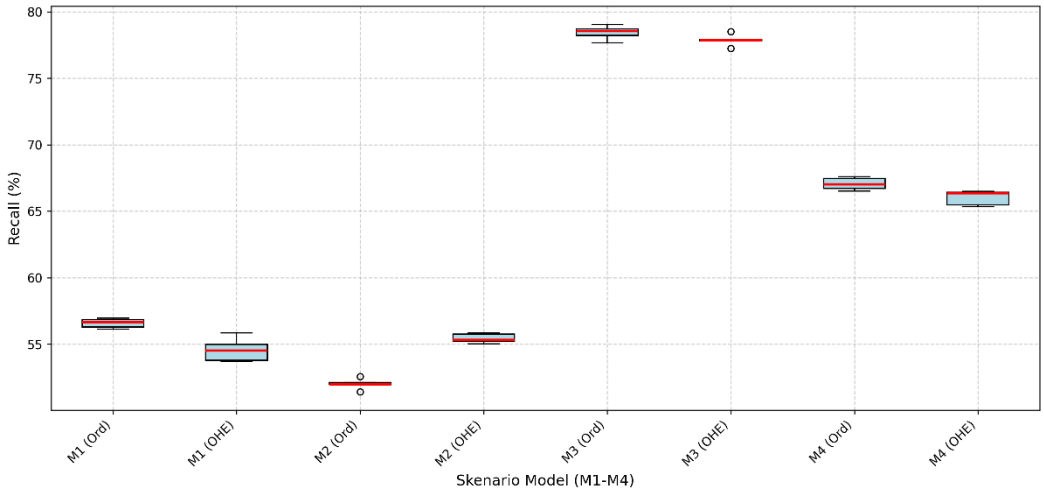
Gambar 4. 13 Perbandingan Presisi Model Tiap Skenario (OOB)

Hasil OOB menunjukkan pola yang konsisten. Presisi OOB pada Model 3 mencapai 77,90% (Ordinal) dan 76,61% (OHE), nilai tertinggi di antara seluruh skenario, sementara Model 4 kembali menunjukkan penurunan pada kedua teknik encoding.

4.3.5.3 *Recall*

Recall merupakan metrik yang sangat krusial dalam evaluasi model, terutama pada konteks medis, karena mengukur kemampuan model dalam mengenali seluruh kasus positif secara menyeluruh. Boxplot berikut memberikan

gambaran distribusi nilai *Recall* dari 5-*Fold Cross -Validation* pada setiap skenario model.



Gambar 4. 14 Boxplot Distribusi Nilai Presisi Seluruh Model

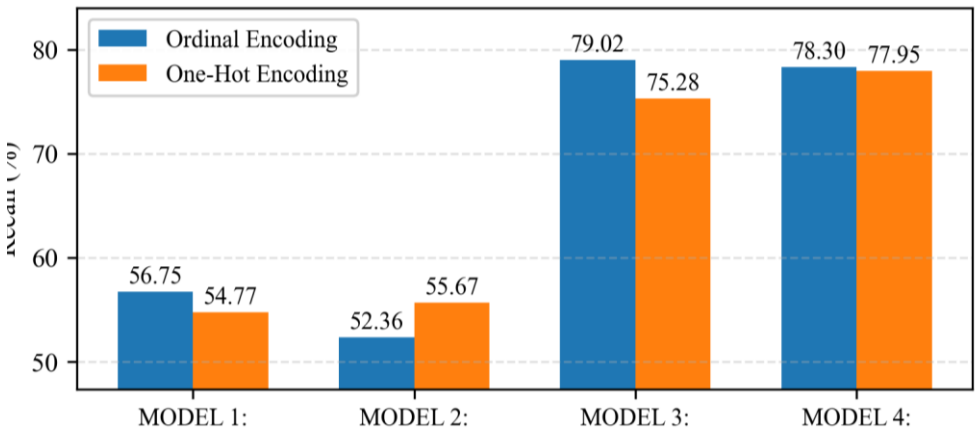
Visualisasi Boxplot memperlihatkan bahwa Model 1 memiliki nilai *Recall* rata-rata 56,58% ± 0,32 (Ordinal) dan 54,57% ± 0,79 (OHE), sementara Model 2 menunjukkan nilai 52,02% ± 0,37 (Ordinal) dan 55,44% ± 0,32 (OHE). Peningkatan *Recall* paling drastis terjadi pada Model 3, dengan rata-rata mencapai 78,43% ± 0,46 (Ordinal) dan 77,86% ± 0,40 (OHE), mencerminkan efektivitas SMOTE dalam meningkatkan sensitivitas model terhadap kelas minoritas. Pada Model 4, *Recall* menurun menjadi 67,06% ± 0,41 (Ordinal) dan 66,01% ± 0,50 (OHE), meskipun masih lebih tinggi dari Model 1 dan Model 2. Dua perbandingan dengan perbedaan *Recall* paling signifikan secara statistik disajikan pada tabel berikut.

Tabel 4. 24 Perbandingan *Paired T-Test Recall*

Perbandingan	t-Statistik	p-Value	Signifikansi
M1 vs M2 (Ord)	16.3122	8.27×10^{-5}	*** ($p < 0.001$)
M1 vs M3 (Ord)	-72.7052	2.14×10^{-7}	*** ($p < 0.001$)
M1 vs M4 (Ord)	-45.1029	1.45×10^{-6}	*** ($p < 0.001$)

Perbandingan	t-Statistik	p-Value	Signifikansi
M2 vs M3 (Ord)	-77.2465	1.68×10^{-7}	*** ($p < 0.001$)
M2 vs M4 (Ord)	-100.1377	5.96×10^{-8}	*** ($p < 0.001$)
M3 vs M4 (Ord)	38.2885	2.78×10^{-6}	*** ($p < 0.001$)

Berdasarkan tabel tersebut, perbandingan M2 vs M4 (Ordinal) menghasilkan t-statistik tertinggi (t = -100,14, p < 0,001), menunjukkan bahwa perbedaan *Recall* antara model tanpa SMOTE (Model 2) dan model dengan SMOTE+PCA (Model 4) merupakan yang paling besar dan konsisten. Perbandingan M2 vs M3 (Ordinal) menyusul dengan t = -77,25 (p < 0,001), menegaskan bahwa SMOTE tanpa PCA memberikan peningkatan *Recall* yang jauh lebih besar dibandingkan baseline PCA.

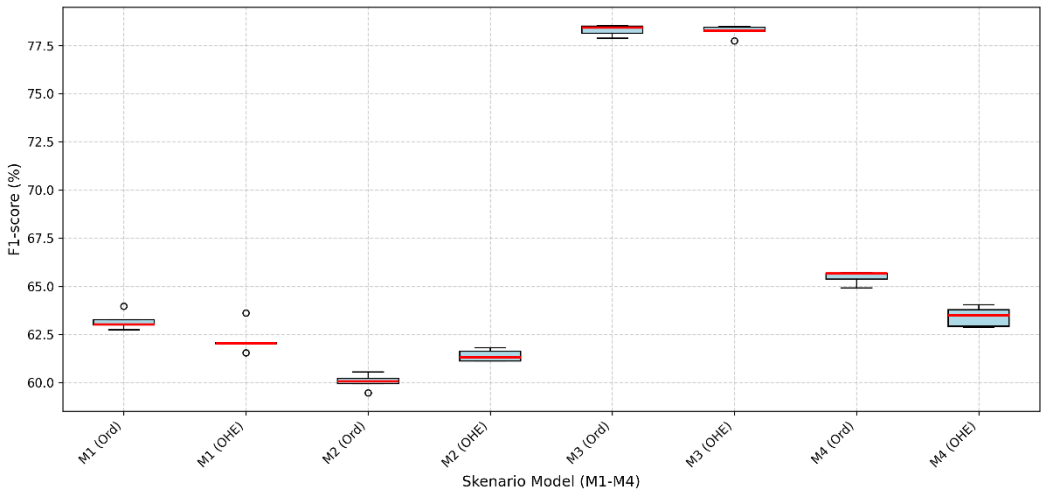


Gambar 4. 15 Perbandingan Recall Model Tiap Skenario (OOB)

Hasil OOB menunjukkan peningkatan yang sejalan, dengan *Recall* Model 3 mencapai 79,02% (Ordinal) dan 75,28% (OHE). Model 4 kembali menunjukkan penurunan *Recall* OOB pada kedua teknik encoding, mengonfirmasi konsistensi dampak negatif penerapan PCA setelah SMOTE.

4.3.5.4 *F1-Score*

Analisis *F1-Score* membantu menunjukkan sejauh mana model mampu memberikan prediksi positif yang akurat sekaligus tidak melewatkan kasus positif yang sebenarnya. Boxplot berikut menampilkan distribusi nilai *F1-Score* dari 5-*Fold Cross -Validation* pada setiap skenario model.



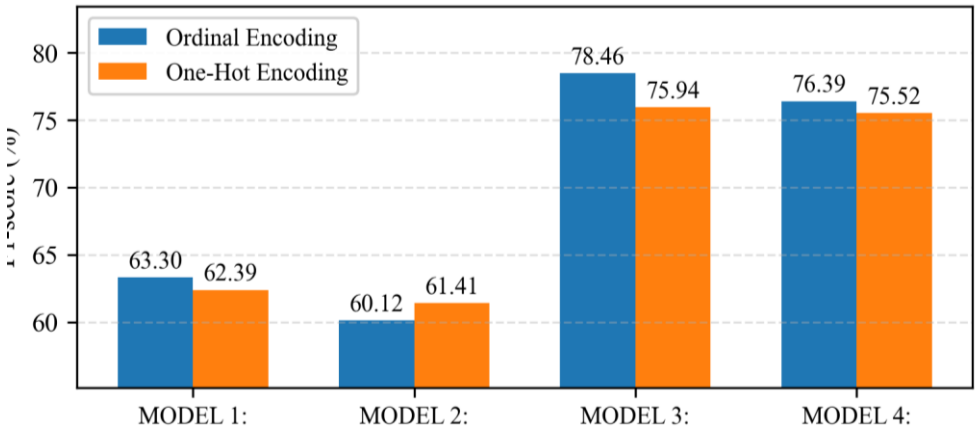
Gambar 4. 16 Boxplot Distribusi Nilai *F1-Score* Seluruh Model

Berdasarkan visualisasi Boxplot, Model 1 memiliki *F1-Score* rata-rata 63,20% ± 0,42 (Ordinal) dan 62,26% ± 0,70 (OHE), sementara Model 2 menunjukkan nilai 60,05% ± 0,36 (Ordinal) dan 61,40% ± 0,28 (OHE). Pada Model 3, *F1-Score* meningkat tajam menjadi 78,30% ± 0,25 (Ordinal) dan 78,23% ± 0,26 (OHE), mencerminkan keseimbangan presisi dan *Recall* yang sangat baik setelah distribusi kelas diseimbangkan menggunakan SMOTE. Pada Model 4, *F1-Score* menurun menjadi 65,47% ± 0,30 (Ordinal) dan 63,43% ± 0,46 (OHE), meskipun masih lebih tinggi dari baseline secara signifikan. Dua perbandingan dengan perbedaan *F1-Score* paling signifikan secara statistik disajikan pada tabel berikut.

Tabel 4. 25 Perbandingan *Paired T-Test* F1-Score

Perbandingan	t-Statistik	p-Value	Signifikansi
M1 vs M2 (Ord)	8.4167	1.09×10^{-3}	** ($p < 0.01$)
M1 vs M3 (Ord)	-54.9713	6.56×10^{-7}	*** ($p < 0.001$)
M1 vs M4 (Ord)	-6.3653	3.12×10^{-3}	** ($p < 0.01$)
M2 vs M3 (Ord)	-76.2576	1.77×10^{-7}	*** ($p < 0.001$)
M2 vs M4 (Ord)	-63.1907	3.76×10^{-7}	*** ($p < 0.001$)
M3 vs M4 (Ord)	71.8501	2.25×10^{-7}	*** ($p < 0.001$)

Berdasarkan tabel tersebut, perbandingan M2 vs M3 (Ordinal) menghasilkan t-statistik tertinggi (t = -76,26, p < 0,001), menunjukkan bahwa peningkatan *F1-Score* dari model PCA tanpa SMOTE ke model SMOTE tanpa PCA merupakan perbedaan yang paling besar. Perbandingan M3 vs M4 (Ordinal) menyusul dengan t = 71,85 (p < 0,001), menegaskan bahwa penambahan PCA setelah SMOTE menurunkan keseimbangan presisi-*Recall* secara sangat nyata.

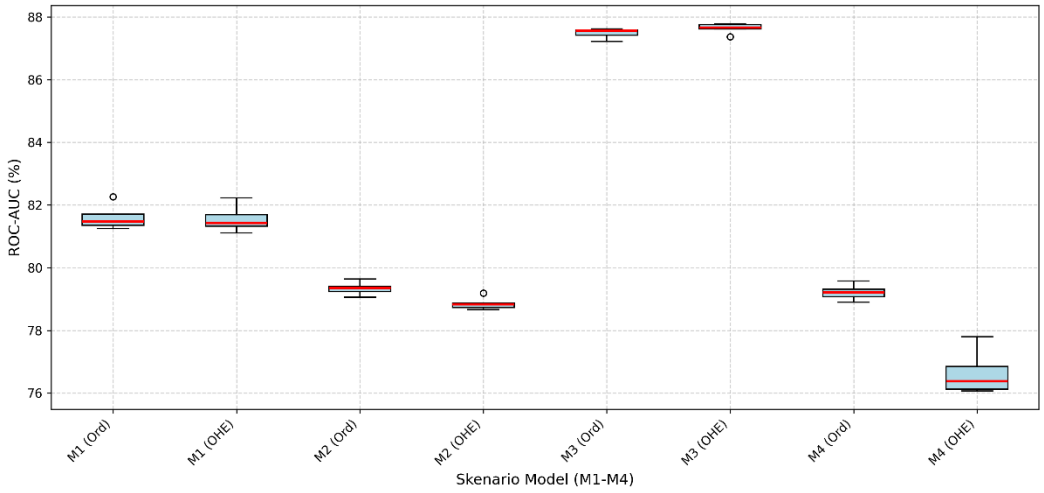


Gambar 4. 17 Perbandingan *F1-Score* model Tiap Skenario (OOB)

Hasil OOB menunjukkan tren yang serupa, dengan *F1-Score* Model 3 mencapai 78,46% (Ordinal) dan 75,94% (OHE). Model 4 kembali mengalami penurunan terutama pada OHE, mengonfirmasi konsistensi dampak negatif PCA pada keseimbangan performa model.

4.3.5.5 ROC-AUC

Analisis ROC-AUC penting untuk melihat kualitas pemisahan kelas yang dihasilkan model pada setiap skenario, terlebih pada kasus medis yang mengharuskan tingkat diskriminasi yang baik. Boxplot berikut menyajikan distribusi nilai ROC-AUC dari 5-*Fold Cross -Validation* pada empat skenario model.



Gambar 4. 18 Boxplot Perbandingan ROC-AUC Seluruh Model

Berdasarkan visualisasi Boxplot, Model 1 memiliki ROC-AUC rata-rata 81,61% ± 0,36 (Ordinal) dan 81,56% ± 0,39 (OHE). Model 2 menunjukkan penurunan menjadi 79,34% ± 0,19 (Ordinal) dan 78,86% ± 0,18 (OHE), dengan perbedaan yang sangat signifikan terhadap Model 1 pada kedua encoding, mengindikasikan bahwa PCA tanpa SMOTE secara nyata menurunkan kemampuan diskriminasi kelas.

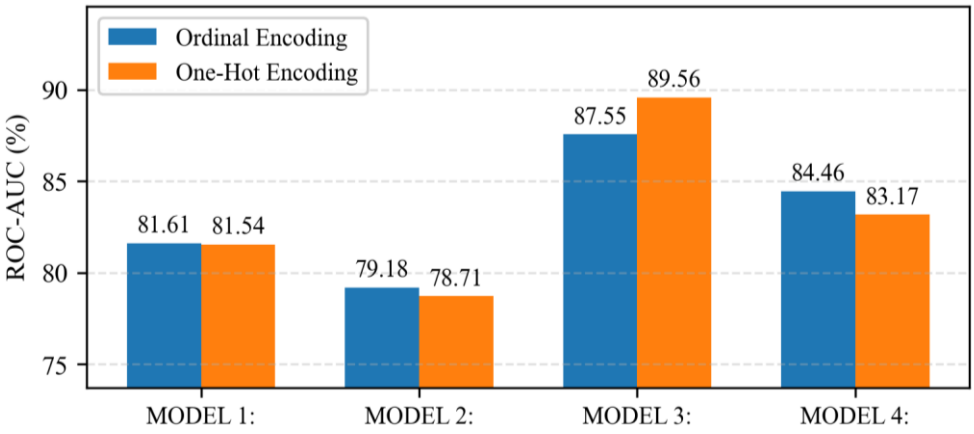
Pada Model 3, ROC-AUC meningkat signifikan menjadi 87,48% ± 0,15 (Ordinal) dan 87,63% ± 0,15 (OHE), dengan sebaran paling sempit di antara semua skenario. Model 4 mengalami penurunan menjadi 79,22% ± 0,23 (Ordinal) dan

76,65% ± 0,64 (OHE), dengan sebaran OHE yang paling lebar, memperkuat temuan bahwa PCA menyebabkan hilangnya pola pembeda antar kelas. Dua perbandingan dengan perbedaan ROC-AUC paling signifikan secara statistik disajikan pada tabel berikut.

Tabel 4. 26 Perbandingan *Paired T-Test* ROC-AUC

Perbandingan	t-Statistik	p-Value	Signifikansi
M1 vs M2 (OHE)	11.1565	3.67×10^{-4}	*** ($p < 0.001$)
M1 vs M3 (Ord)	-28.1205	9.51×10^{-6}	*** ($p < 0.001$)
M1 vs M4 (OHE)	30.0094	7.34×10^{-6}	*** ($p < 0.001$)
M2 vs M3 (OHE)	-56.5746	5.84×10^{-7}	*** ($p < 0.001$)
M2 vs M4 (OHE)	5.8635	4.22×10^{-3}	** ($p < 0.01$)
M3 vs M4 (Ord)	50.6641	9.08×10^{-7}	*** ($p < 0.001$)

Berdasarkan tabel tersebut, perbandingan M2 vs M3 pada kedua encoding menghasilkan t-statistik tertinggi (OHE: t = -56,57, p < 0,001; Ordinal: t = -55,66, p < 0,001), menunjukkan bahwa peningkatan kemampuan diskriminasi kelas dari Model 2 ke Model 3 merupakan perbedaan yang paling besar dan konsisten, menegaskan bahwa SMOTE memberikan kontribusi paling nyata dalam meningkatkan kualitas pemisahan kelas dibandingkan perubahan lainnya.



Gambar 4. 19 Perbandingan ROC-AUC Model Tiap Skenario (OOB)

Hasil OOB memperkuat temuan evaluasi 5-*Fold*. Model 3 mencapai ROC-AUC OOB tertinggi, yaitu 87,55% (Ordinal) dan 89,56% (OHE). Model 4 kembali mengalami penurunan terutama pada OHE, sementara Ordinal Encoding cenderung lebih stabil terhadap reduksi dimensi dibandingkan One-Hot Encoding.

4.3.6 Kesimpulan Analisis Perbandingan Skenario Model

Subbab ini menyajikan ringkasan perbandingan performa empat skenario model prediksi serangan jantung berdasarkan hasil evaluasi 5-*Fold cross validation* dan *Out-of-Bag* (OOB). Evaluasi dilakukan menggunakan lima metrik utama, yaitu akurasi, presisi, *Recall*, *F1-Score*, dan ROC-AUC, dengan dua teknik *preprocessing* kategorikal, yakni Ordinal Encoding dan One-Hot Encoding. Berikut ringkasan hasil evaluasi disajikan dalam bentuk tabel untuk memudahkan analisis perbedaan performa antar skenario model dan metode validasi.

Tabel 4. 27 Performa Matriks Evaluasi 5-*Fold* pada Setiap Skenario

Skenario	Ordinal Encoding					One-Hot Encoding				
	Akurasi	Presisi (%)	*Recall* (%)	*F1-Score*	ROC-AUC	Akurasi	Presisi (%)	*Recall* (%)	*F1-Score*	ROC-AUC
Model 1	73,58 ± 0,23	71,58 ± 0,95	56,58 ± 0,32	63,20 ± 0,42	81,61 ± 0,36	73,48 ± 0,32	72,48 ± 1,00	54,57 ± 0,79	62,26 ± 0,70	81,56 ± 0,39
Model 2	72,25 ± 0,21	71,03 ± 0,32	52,02 ± 0,37	60,05 ± 0,36	79,34 ± 0,19	72,05 ± 0,19	68,80 ± 0,29	55,44 ± 0,32	61,40 ± 0,28	78,86 ± 0,18
Model 3	78,26 ± 0,22	78,16 ± 0,22	78,43 ± 0,46	78,30 ± 0,25	87,48 ± 0,15	78,34 ± 0,25	78,61 ± 0,27	77,86 ± 0,40	78,23 ± 0,26	87,63 ± 0,15
Model 4	71,63 ± 0,24	63,95 ± 0,31	67,06 ± 0,41	65,47 ± 0,30	79,22 ± 0,23	69,47 ± 0,40	61,04 ± 0,49	66,01 ± 0,50	63,43 ± 0,46	76,65 ± 0,64

Tabel 4.28 menyajikan hasil evaluasi performa model menggunakan metode 5-*Fold cross validation* pada empat skenario model dengan dua teknik *encoding*, yaitu Ordinal Encoding dan One-Hot Encoding. Evaluasi dilakukan menggunakan lima metrik utama, yakni akurasi, presisi, *Recall*, *F1-Score*, dan ROC-AUC.

Berdasarkan tabel tersebut, Model 1 (Random Forest) dan Model 2 (Random Forest + PCA) menunjukkan performa yang relatif stabil namun masih terbatas, khususnya pada metrik *Recall* dan *F1-Score*. Nilai *Recall* pada kedua model ini cenderung rendah, mengindikasikan bahwa model masih belum optimal dalam mendeteksi kasus serangan jantung sebagai kelas minoritas. Hasil uji t berpasangan mengonfirmasi bahwa perbedaan performa antara Model 1 dan Model 2 tidak selalu signifikan pada seluruh metrik khususnya pada presisi Ordinal Encoding ($t = 0,89$, $p = 0,424$) dan *Recall* OHE ($t = -2,23$, $p = 0,090$) yang mengindikasikan bahwa penerapan PCA sebagai teknik simplifikasi tidak memberikan dampak yang berarti pada kondisi tersebut.

Peningkatan performa yang signifikan terjadi pada Model 3 (Random Forest + SMOTE). Pada skenario ini, seluruh metrik evaluasi mengalami kenaikan yang konsisten baik pada Ordinal Encoding maupun One-Hot Encoding. Hasil uji t berpasangan menunjukkan bahwa seluruh peningkatan performa Model 3 dibandingkan Model 1 maupun Model 2 signifikan secara statistik pada semua metrik ($p < 0,001$), menegaskan bahwa penerapan SMOTE bukan sekadar peningkatan kebetulan melainkan kontribusi nyata dalam menyeimbangkan distribusi kelas. Peningkatan paling ekstrem tercatat pada perbandingan M2 vs M3 untuk metrik *Recall* ($t = -77,25$, $p < 0,001$) dan *F1-Score* ($t = -76,26$, $p < 0,001$),

serta pada perbandingan M3 vs M4 untuk presisi ($t = 103{,}93$, $p < 0{,}001$), yang seluruhnya menunjukkan perbedaan yang sangat besar dan sangat konsisten antar *Fold*.

Namun, pada Model 4 (Random Forest + SMOTE + PCA) terlihat adanya penurunan performa dibandingkan Model 3, terutama pada One-Hot Encoding. Hasil uji t berpasangan mengonfirmasi bahwa penurunan ini signifikan secara statistik pada seluruh metrik ($p < 0{,}001$), menunjukkan bahwa penerapan PCA setelah SMOTE berpotensi menghilangkan informasi penting dari fitur kategorikal berdimensi tinggi sehingga berdampak negatif terhadap performa model. Menariknya, perbandingan Model 1 dan Model 4 menunjukkan bahwa pada sebagian metrik seperti *Recall* dan *F1-Score*, Model 4 masih lebih tinggi dari baseline secara signifikan ($p < 0{,}05$), yang mengindikasikan bahwa gabungan SMOTE+PCA tetap memberikan kontribusi positif meskipun tidak seoptimal Model 3.

Secara keseluruhan, hasil evaluasi 5-*Fold* beserta uji t berpasangan menunjukkan bahwa Model 3 merupakan skenario dengan performa terbaik dan paling seimbang di antara seluruh skenario yang diuji, dengan seluruh peningkatan performanya terbukti signifikan secara statistik.

Perbandingan M3 vs M4 secara khusus dipilih sebagai perbandingan yang paling informatif karena keduanya merupakan skenario yang dapat dibandingkan secara langsung dan setara sama-sama menggunakan SMOTE, dengan satu-satunya perbedaan adalah ada atau tidaknya PCA yang diterapkan sebelum SMOTE. Perbandingan ini secara langsung menjawab pertanyaan apakah penambahan PCA

sebelum SMOTE memberikan manfaat atau justru merugikan performa model. Hasilnya, nilai t-statistik pada pasangan ini merupakan yang tertinggi di antara seluruh perbandingan ($t = 54{,}40$, $p < 0{,}001$), membuktikan bahwa penurunan akurasi akibat penerapan PCA sebelum SMOTE bukan sekadar variasi acak, melainkan efek yang nyata, besar, dan konsisten di seluruh *Fold* validasi.

Tabel 4. 28 Perbandingan Matriks Evaluasi OOB pada setiap Skenario

Skenario	**Ordinal Encoding**					**One-Hot Encoding**				
	Akurasi (%)	**Presisi (%)**	***Recall* (%)**	***F1-Score*(**	**ROC-AUC**	**Akurasi (%)**	**Presisi (%)**	***Recall* (%)**	***F1-Score***	**ROC-AUC**
Model 1	73,67	71,69	56,75	63,30	81,61	73,52	72,47	54,77	62,39	81,54
Model 2	72,14	70,58	52,36	60,12	79,18	71,94	68,47	55,6	61,41	78,71
Model 3	78,3	77,90	79,02	78,46	87,55	80,87	76,61	75,28	75,94	89,56
Model 4	75,81	74,58	78,30	76,39	84,46	74,73	73,23	77,95	75,52	83,17

Tabel 4.29 menampilkan hasil evaluasi performa model menggunakan metode *Out-of-Bag* (OOB), yang merupakan metode validasi internal pada algoritma Random Forest. Evaluasi ini bertujuan untuk mengukur kemampuan generalisasi model terhadap data yang tidak digunakan dalam proses pelatihan setiap pohon keputusan.

Hasil evaluasi OOB menunjukkan pola yang konsisten dengan hasil 5-*Fold cross* validation. Model 1 dan Model 2 masih memiliki performa yang relatif rendah terutama pada metrik Recall dan F1-Score, dengan selisih hingga 26,66 poin di bawah Model 3, yang menandakan bahwa model tanpa penanganan ketidakseimbangan kelas belum mampu mendeteksi kasus positif secara optimal.

Pada Model 3, terjadi peningkatan performa yang paling signifikan dibandingkan skenario lainnya. *Recall* meningkat drastis hingga 22,27 hingga 26,66 poin pada *encoding* ordinal, dan 19,68 hingga 20,51 poin pada one-hot encoding, dibandingkan Model 1 dan Model 2. ROC-AUC juga meningkat signifikan, dari 79,18%–81,61% pada Model 1 dan 2 menjadi 87,55% (ordinal) dan 89,56% (one-hot) pada Model 3, atau naik sebesar 5,94 hingga 10,85 poin.

Sebaliknya, Model 4 kembali menunjukkan penurunan performa, terutama pada One-Hot Encoding. Penurunan nilai ROC-AUC dan *F1-Score* mengindikasikan bahwa kombinasi SMOTE dan PCA tidak memberikan keuntungan tambahan, bahkan cenderung menurunkan kualitas pemodelan.

Secara keseluruhan, Model 3 (Random Forest dengan SMOTE) dapat disimpulkan sebagai model terbaik dalam penelitian ini untuk kedua jenis *encoding*. Model ini memiliki keseimbangan performa yang paling optimal pada seluruh metrik, terutama *Recall* dan *F1-Score* yang sangat penting dalam konteks deteksi risiko serangan jantung. Peningkatan sensitivitas model dalam mengenali pasien berisiko jauh lebih bernilai dibanding hanya mempertahankan akurasi pada data yang tidak seimbang.

Dalam penelitian ini, akurasi menjadi indikator penting untuk menilai sejauh mana model mampu menghasilkan deteksi yang tepat sesuai dengan kondisi sebenarnya. Ketepatan informasi ini sejalan dengan prinsip verifikasi yang diajarkan dalam Al-Qur’an, sebagaimana termaktub dalam QS. Al-Hujurat ayat 6. Allah berfirman:

يٰٓاَيُّهَا الَّذِيْنَ اٰمَنُوْٓا اِنْ جَاۤءَكُمْ فَاسِقٌۢ بِنَبَاٍ فَتَبَيَّنُوْٓا اَنْ تُصِيْبُوْا قَوْمًاۢ بِجَهَالَةٍ فَتُصْبِحُوْا عَلٰى مَا فَعَلْتُمْ نٰدِمِيْنَ

"*Wahai orang-orang yang beriman, jika seorang fasik datang kepadamu membawa berita penting, maka telitilah kebenarannya agar kamu tidak mencelakakan suatu kaum karena ketidaktahuan(-mu) yang berakibat kamu menyesali perbuatanmu itu*." *(Q.S. Al-Hujurat: 6)*

Menurut Tafsir Kementerian Agama Republik Indonesia, ayat ini menegaskan pentingnya melakukan pemeriksaan, verifikasi, dan ketelitian terhadap setiap informasi sebelum diambil sebagai dasar keputusan. Hal ini bertujuan agar seseorang tidak menjadi korban dari data atau berita yang tidak akurat. Sementara itu, menurut Tafsir Ibnu Katsir, ayat ini berfungsi sebagai peringatan agar tidak bersandar pada informasi yang meragukan, karena keputusan yang salah dapat menimbulkan dampak negatif bagi diri sendiri maupun orang lain.

Dalam konteks penelitian ini, ayat tersebut memberikan landasan filosofis bahwa model klasifikasi harus menghasilkan prediksi yang akurat agar tidak menyesatkan dalam interpretasi, terutama pada bidang medis yang menyangkut keselamatan pasien. Akurasi yang tinggi mencerminkan ketelitian dan kebenaran informasi yang dihasilkan model, sehingga proses pengambilan keputusan menjadi lebih valid dan dapat dipertanggungjawabkan, sebagaimana prinsip verifikasi yang diajarkan dalam ayat tersebut.

4.4 Hasil Implementasi Web

Sistem deteksi risiko serangan jantung diimplementasikan dalam bentuk aplikasi web menggunakan *framework* Streamlit dengan bahasa pemrograman Python. Aplikasi ini dirancang untuk memudahkan pengguna dalam melakukan

deteksi serangan jantung dengan antarmuka yang intuitif dan interaktif. Berikut adalah hasil implementasi dari setiap komponen aplikasi web yang telah dikembangkan.

4.4.1 Arsitektur Sistem

Aplikasi web ini dibangun dengan arsitektur modular yang terdiri dari beberapa komponen utama:

a. *Frontend Layer*: Menggunakan Streamlit sebagai *framework* untuk membangun antarmuka pengguna yang responsif dan interaktif. Streamlit dipilih karena kemampuannya dalam membuat aplikasi *data science* dengan cepat tanpa memerlukan pengetahuan *web development* yang mendalam.

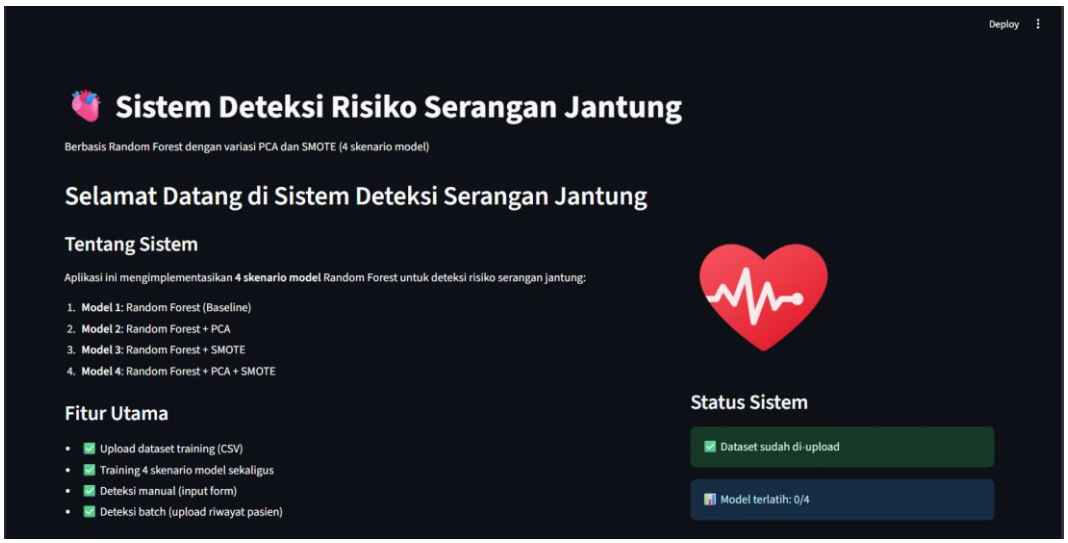
b. *Processing Layer*: Menggunakan library scikit-learn untuk *preprocessing* data dan *Training* model *machine learning*. *Library imbalanced-learn* digunakan untuk menangani ketidakseimbangan kelas dengan teknik SMOTE.

c. *Model Layer*: Mengimplementasikan empat skenario model Random Forest dengan variasi penggunaan PCA (*Principal Component Analysis*) dan SMOTE (*Synthetic Minority Over-sampling Technique*).

d. *Session State Management*: Menggunakan fitur session state dari *Streamlit* untuk menyimpan model yang sudah dilatih, *prep*AUC*essor*, dan status aplikasi agar tetap persisten selama sesi pengguna.

4.4.2 Halaman Utama

Halaman utama menampilkan informasi sistem secara ringkas, termasuk deskripsi fitur, status dataset, serta jumlah model yang telah dilatih. Tampilan ini menjadi titik awal bagi pengguna untuk mengakses seluruh fungsi dalam aplikasi.



Gambar 4. 20 Tampilan Halaman Utama *Website*

Halaman utama aplikasi berfungsi sebagai *landing page* yang memberikan informasi komprehensif tentang sistem. Halaman ini menampilkan deskripsi singkat tentang aplikasi, empat skenario model yang tersedia, fitur-fitur utama sistem, dan panduan penggunaan aplikasi. Selain itu, halaman ini juga menampilkan status sistem secara *real-time* yang menunjukkan apakah dataset sudah di-*Upload* dan jumlah model yang telah dilatih. Informasi status ini membantu pengguna untuk mengetahui langkah selanjutnya yang perlu dilakukan.

4.4.3 Halaman Unggah dan Melatih Dataset

Halaman ini merupakan komponen inti dari aplikasi yang memungkinkan pengguna untuk mengunggah dataset *Training* dan melatih empat skenario model *machine learning*. Fitur-fitur yang tersedia pada halaman ini meliputi:

4.4.3.1 Fitur Unggah Dataset

Fitur *Upload Dataset* digunakan untuk mengunggah data pelatihan ke dalam sistem. Melalui fitur ini, pengguna dapat memilih *file* CSV yang akan diproses untuk kebutuhan pelatihan model. Tampilan fitur tersebut ditunjukkan pada Gambar 4. 21.

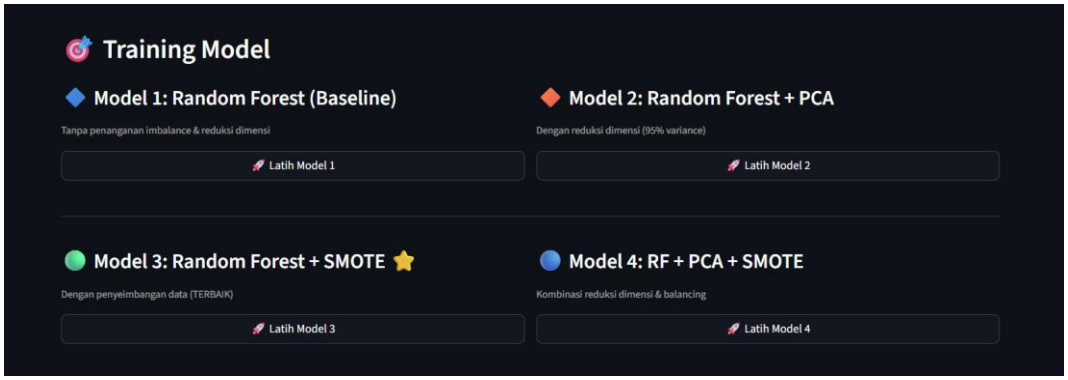
Gambar 4. 21 Fitur *Upload Dataset Training*

Sistem ini menyediakan *file Uploader* yang mendukung format CSV dengan deteksi delimiter otomatis. Setelah dataset berhasil di *Upload*, sistem akan menampilkan informasi ringkasan dataset meliputi total data, jumlah fitur, distribusi kelas (jumlah kasus serangan jantung dan non-serangan jantung), serta *preview* lima baris pertama dari dataset. Fitur deteksi delimiter otomatis

memungkinkan sistem untuk membaca berbagai format CSV dengan separator yang berbeda seperti koma, *semicolon*, *tab*, atau *pipe*.

4.4.3.2 Fitur Melatih Model

Setelah dataset di-*Upload*, pengguna dapat melatih empat skenario model secara terpisah. Setiap model memiliki tombol *Training* tersendiri. Tampilan halaman tersebut ditunjukkan pada Gambar 4. 22.



Gambar 4. 22 Fitur *Training Model*

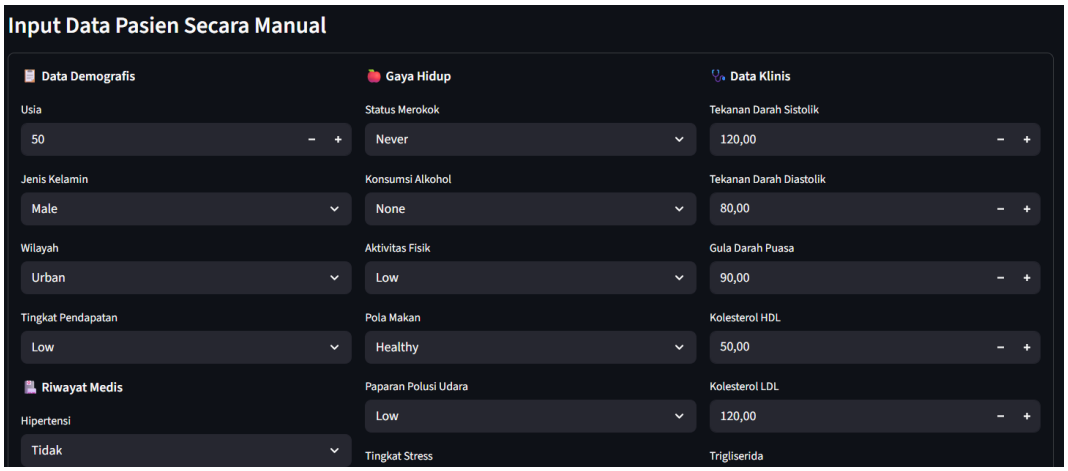
Keempat skenario model yang diimplementasikan adalah:

a. Model 1 - Random Forest (*Baseline*): Model dasar tanpa penanganan ketidakseimbangan kelas dan tanpa reduksi dimensi. Menggunakan 200 *estimators* dengan *max_depth*=10.
b. Model 2 - Random Forest dengan PCA: Model dengan reduksi dimensi menggunakan PCA yang mempertahankan 95% *variance*. Menggunakan 600 *n_estimators* dengan *max_depth*=5.
c. Model 3 - Random Forest dengan SMOTE: Model dengan penyeimbangan data menggunakan teknik SMOTE. Menggunakan 200 *n_estimators* tanpa batasan *max_depth* dan *criterion entropy*.

d. Model 4 - Random Forest dengan PCA dengan SMOTE: Model kombinasi yang menggunakan baik PCA untuk reduksi dimensi maupun SMOTE untuk penyeimbangan data. Menggunakan 600 *estimators*.

4.4.4 Halaman Memasukkan Data Pasien

Halaman *Input* Data Pasien digunakan untuk memasukkan data pasien secara manual melalui form yang telah disediakan. *Form* ini dirancang agar pengguna dapat mengisi informasi pasien secara lengkap dan terstruktur. Tampilan halaman tersebut ditunjukkan pada Gambar 4. 23.



Gambar 4. 23 *Input* Data Pasien

Fitur *Input* manual memungkinkan pengguna untuk memasukkan data pasien secara langsung melalui form yang terstruktur. Form ini dibagi menjadi tiga kategori utama yaitu data demografis, data gaya hidup, dan data klinis. Data demografis mencakup usia, jenis kelamin, wilayah, dan tingkat pendapatan. Data riwayat medis meliputi hipertensi, diabetes, obesitas, lingkar pinggang, dan riwayat keluarga. Data gaya hidup mencakup status merokok, konsumsi alkohol, aktivitas

fisik, pola makan, paparan polusi udara, tingkat stress, dan jam tidur. Data klinis meliputi tekanan darah sistolik dan diastolik, gula darah puasa, kolesterol HDL dan LDL, trigliserida, hasil EKG, riwayat penyakit jantung, penggunaan obat, dan partisipasi dalam skrining gratis.

Gambar 4. 24 Hasil Deteksi Data Pasien

Setelah pengguna mengisi *form* dan menekan tombol deteksi, sistem akan menampilkan hasil deteksi dari semua model yang telah dilatih. Hasil ditampilkan dalam bentuk tabel yang menunjukkan nama model dan status deteksi (terkena serangan jantung atau tidak terkena serangan jantung).

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil evaluasi, skenario Model 3 (Random Forest dengan SMOTE) merupakan model terbaik dalam penelitian ini untuk kedua jenis encoding. Pada teknik Ordinal Encoding, model mencapai akurasi 78,30% yang berarti model mampu mengklasifikasikan sekitar 78 dari 100 sampel dengan benar, dengan Recall 79,02% yang mengindikasikan bahwa model berhasil mendeteksi hampir 8 dari 10 kasus serangan jantung yang sebenarnya. Pada teknik One-Hot Encoding, model menghasilkan akurasi lebih tinggi mencapai 80,87% yang berarti sekitar 8 dari 10 prediksi model adalah benar, dengan ROC-AUC sebesar 89,56% yang menunjukkan bahwa model memiliki kemampuan pemisahan kelas yang sangat baik, hampir mendekati kategori sempurna dalam membedakan pasien berisiko dan tidak berisiko serangan jantung.

Tingginya nilai *Recall* pada kedua teknik ini sangat krusial dalam konteks medis karena berarti hanya sekitar 1–2 dari 10 pasien yang sebenarnya berisiko serangan jantung yang terlewat oleh model, sehingga secara efektif meminimalkan risiko *false negative* yang dapat berakibat fatal. Hasil ini menegaskan bahwa teknik penyeimbangan data SMOTE memberikan kontribusi yang jauh lebih signifikan terhadap peningkatan performa model dibandingkan penggunaan PCA maupun model dasar tanpa penanganan ketidakseimbangan kelas.

5.2 Saran

Berdasarkan hasil penelitian ini menggunakan dataset “Heart Attack Prediction in Indonesia”, peneliti menyarankan agar pengembangan selanjutnya lebih berfokus pada penggunaan seluruh fitur asli tanpa melakukan reduksi dimensi, seperti *Principal Component Analysis* (PCA). Hal ini didasarkan pada temuan bahwa setiap variabel dalam domain kesehatan memiliki makna biologis dan relevansi klinis yang unik, sehingga mempertahankan fitur asli sangat penting untuk menjaga interpretasi medis yang akurat. Selain itu, disarankan untuk mengeksplorasi algoritma klasifikasi lain atau teknik penyeimbangan data yang lebih variatif guna meningkatkan sensitivitas model tanpa mengorbankan informasi dari indikator kesehatan pasien yang ada.

DAFTAR PUSTAKA

Ali, W., Williams, J., Xiong, B., Zou, J., & Daneshjou, R. (2025). Machine Learning for Early *Detection* of Hidradenitis Suppurativa: A Feasibility Study Using Medical Insurance Claims Data. *JID Innovations*, *5*(3), 100362. https://doi.org/10.1016/j.xjidi.2025.100362

Amshi, A. H., Usman, A., Prasad, R., Salihu, I. A., & Dadile, A. M. (2023). Review of Machine Learning Techniques For Class Imbalance Medical Data Set. *2023 2nd International Conference on Multidisciplinary Engineering and Applied Science (ICMEAS)*, 1–10. https://doi.org/10.1109/ICMEAS58693.2023.10429848

Aswani, T., Gummadi, Dr. J. M., & G., Dr. S. (2025). A Random Forest-Based Machine Learning Framework with PCA, SMOTE, and SHAP for Efficient and Interpretable Coronary Artery Disease Prediction. *Informatica*, *49*(22). https://doi.org/10.31449/inf.v49i22.7998

Aubaidan, B. H., Kadir, R. A., Lajb, M. T., Anwar, M., Qureshi, K. N., Taha, B. A., & Ghafoor, K. (2025). A review of intelligent data analysis: Machine learning approaches for addressing class imbalance in healthcare - challenges and perspectives. *Intelligent Data Analysis: An International Journal*, *29*(3), 699–719. https://doi.org/10.1177/1088467X241305509

Bouqentar, M. A., Terrada, O., Hamida, S., Saleh, S., Lamrani, D., Cherradi, B., & Raihani, A. (2024). Early heart disease prediction using feature engineering and machine learning algorithms. *Heliyon*, *10*(19), e38731. https://doi.org/10.1016/j.heliyon.2024.e38731

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953

Dede Kurniadi, Asri Indah Pertiwi, & Asri Mulyani. (2025). Ensemble Voting Classifier Berbasis Multi-Algoritma dan Metode SMOTE untuk Klasifikasi Penyakit Jantung. *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi*, *14*(2), 145–153. https://doi.org/10.22146/jnteti.v14i2.17157

Hairani, H., Widiyaningtyas, T., & Dwi Prasetya, D. (2024). Addressing Class Imbalance of Health Data: A Systematic Literature Review on Modified Synthetic Minority Oversampling Technique (SMOTE) Strategies. *JOIV : International Journal on Informatics Visualization*, *8*(3), 1310. https://doi.org/10.62527/joiv.8.3.2283

Hossain, S., Hasan, M. K., Faruk, M. O., Aktar, N., Hossain, R., & Hossain, K. (2024). Machine learning approach for predicting cardiovascular disease in Bangladesh: Evidence from a *cross* -sectional study in 2023. *BMC Cardiovascular Disorders*, *24*(1), 214. https://doi.org/10.1186/s12872-024-03883-2

Ibrahim, M. (2023). Evolution of Random Forest from Decision Tree and Bagging: A Bias-Variance Perspective. *Dhaka University Journal of Applied Science and Engineering*, *7*(1), 66–71. https://doi.org/10.3329/dujase.v7i1.62888

Kemenkes RI. (2021). *Peringatan Hari Jantung Sedunia 2021: Jaga Jantungmu untuk Hidup Lebih Sehat*. https://ayosehat.kemkes.go.id/peringatan-hari-jantung-sedunia-2021-jaga-jantungmu-untuk-hidup-lebih-sehat

Kemenkes RI. (2023). *Proyeksi Tenaga Kesehatan Berdasarkan Supply dan Demand/Needs Tahun 2023*. Direktorat Perencanaan Tenaga Kesehatan, Kementerian Kesehatan RI. https://repositori-ditjen-nakes.kemkes.go.id/779/1/CETAK_Dokumen%20Proyeksi%20Tenaga%20Kesehatan%20Berdasarkan%20Supply%20dan%20%20Demand_Needs%20Tahun%202023%20%28A5%29%20%281%29.pdf

Kuhn, M., & Johnson, K. (2019). *Feature Engineering and Selection: A Practical Approach for Predictive Models* (1st ed.). Chapman and Hall/CRC. https://doi.org/10.1201/9781315108230

Lamir, A. A., Razzagzadeh, S., & Rezaei, Z. (2025). *A Comprehensive Machine Learning Framework for Heart Disease Prediction: Performance Evaluation and Future Perspectives* (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2505.09969

Mehrabinezhad, A., Teshnehlab, M., & Sharifi, A. (2024). A comparative study to examine principal component analysis and kernel principal component analysis-based weighting layer for convolutional neural networks. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, *12*(1), 2379526. https://doi.org/10.1080/21681163.2024.2379526

Micheal, O. (2023, August 27). A Comprehensive Comparison between One-Hot and Ordinal Encoding. *Medium*. https://medium.com/@oyebamijimicheal10/a-comprehensive-comparison-between-one-hot-and-ordinal-encoding-6f899c4f08b3

Naseriparsa, M., & Kashani, M. M. R. (2013). Combination of PCA with SMOTE Resampling to Boost the Prediction Rate in Lung Cancer Dataset. *International Journal of Computer Applications*, *77*(3), 33–38. https://doi.org/10.5120/13376-0987

Nasution, N., Hasan, M. A., & Bakri Nasution, F. (2025). Predicting Heart Disease Using Machine Learning: An Evaluation of Logistic Regression, Random Forest, SVM, and KNN Models on the UCI Heart Disease Dataset. *IT Journal Research and Development*, *9*(2), 140–150. https://doi.org/10.25299/itjrd.2025.17941

Priyanto, D., Hairani, H., Marzuki, K., & Innuddin, M. (2025). *Optimization of* Random Forest *for Health Data Classification Using PCA and K-Means*. *15*(5).

Rimal, Y., Sharma, N., Paudel, S., Alsadoon, A., Koirala, M. P., & Gill, S. (2025). Comparative analysis of heart disease prediction using logistic regression, SVM, KNN, and random forest with *cross -validation* for improved accuracy. *Scientific Reports*, *15*(1), 13444. https://doi.org/10.1038/s41598-025-93675-1

Scientist, W. F., Data. (2023, August 3). Ordinal Encoding—A Brief Explanation. *Medium*. https://medium.com/@WojtekFulmyk/ordinal-encoding-a-brief-explanation-a29cf374dbc1

SKI. (2023). *Laporan Hasil Survei Kesehatan Indonesia (SKI) 2023*. https://kemkes.go.id/app_asset/file_content_download/17169067256655eae5553985.98376730.pdf

Talebi Moghaddam, M., Jahani, Y., Arefzadeh, Z., Dehghan, A., Khaleghi, M., Sharafi, M., & Nikfar, G. (2024). Predicting diabetes in adults: Identifying important features in unbalanced data over a 5-year cohort study using machine learning algorithm. *BMC Medical Research Methodology*, *24*(1), 220. https://doi.org/10.1186/s12874-024-02341-z

Wang, D., Wang, X., Wang, L., Li, M., Da, Q., Liu, X., Gao, X., Shen, J., He, J., Shen, T., Duan, Q., Zhao, J., Li, K., Qiao, Y., & Zhang, S. (2023). A Real-world Dataset and Benchmark For Foundation Model Adaptation in Medical Image Classification. *Scientific Data*, *10*(1), 574. https://doi.org/10.1038/s41597-023-02460-0

Wei, X., & Shi, B. (2025a). Reducing bias in coronary heart disease prediction using Smote-ENN and PCA. *PLOS One*, *20*(8), e0327569. https://doi.org/10.1371/journal.pone.0327569

Wei, X., & Shi, B. (2025b). Reducing bias in coronary heart disease prediction using Smote-ENN and PCA. *PLOS One*, *20*(8), e0327569. https://doi.org/10.1371/journal.pone.0327569

WHO. (2024). *Health Workforce*. World Health Organization. https://www.who.int/health-topics/health-workforce

WHO. (2025). *Cardiovascular diseases (CVDs)*. Cardiovascular Diseases (CVDs). https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)

Zhang, Y., Deng, L., & Wei, B. (2024). Imbalanced Data Classification Based on Improved Random-SMOTE and Feature Standard Deviation. *Mathematics*, *12*(11), 1709. https://doi.org/10.3390/math12111709

Zhu, W., Qiu, R., & Fu, Y. (2024). *Comparative Study on the Performance of Categorical Variable Encoders in Classification and Regression Tasks* (arXiv:2401.09682). arXiv. https://doi.org/10.48550/arXiv.2401.09682