

Optimizing XGBoost for Heart Disease Risk Classification Using Optuna and Random Search on the Behavioral Risk Factor Surveillance System (BRFSS) 2023 Dataset

Muhammad Dzaky^{1*}, Adam Prayogo Kuncoro^{2*}, Riyanto^{3*}

* Informatika, Fakultas Ilmu Komputer, Universitas Amikom Purwokerto
muh.dzaky1919@gmail.com¹, adam@amikompurwokerto.ac.id², riyanto@amikompurwokerto.ac.id³

Article Info

Article history:

Received 2025-11-30

Revised 2026-01-19

Accepted 2026-01-30

Keyword:

Heart Disease,
XGBoost,
Optuna,
Random Search,
BRFSS 2023.

ABSTRACT

Heart disease is a critical public health issue in Indonesia, contributing to approximately 1,5 million deaths annually. Although machine learning methods, particularly Extreme Gradient Boosting (XGBoost), have demonstrated strong performance in medical classification tasks, their optimization on large-scale and highly imbalanced health datasets remains underexplored. This study optimizes XGBoost for heart disease risk classification using the Behavioral Risk Factor Surveillance System (BRFSS) 2023 dataset, consisting of 290.156 samples after preprocessing. Two hyperparameter optimization approaches, Optuna and Random Search, are evaluated across three class imbalance handling techniques, namely class weighting, SMOTE, and Random Undersampling (RUS). Model evaluation focuses on AUC and recall to prioritize sensitivity in identifying individuals at risk. The results show that the OptunaRUS and RandomWeight models achieve the most stable performance, with OptunaRUS attaining an AUC of 83,06% and a recall of 75,69% on the test dataset. Feature importance analysis indicates that age range and hypertension are the most influential predictors. These findings confirm that hyperparameter optimization on large-scale health data improves model discriminative capability and generalization, while selective sampling strategies such as RUS provide more stable performance than generative methods in high-dimensional datasets.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. PENDAHULUAN

Penyakit jantung merupakan salah satu penyebab kematian tertinggi di dunia maupun di Indonesia. Kondisi ini terjadi akibat adanya gangguan pada pembuluh darah, katup, atau otot jantung yang menyebabkan fungsi jantung tidak bekerja dengan normal [1]. Risiko terjadinya penyakit jantung dapat dipengaruhi oleh berbagai faktor, seperti usia, jenis kelamin, hipertensi, kolesterol tinggi, diabetes, kebiasaan merokok, kurangnya aktivitas fisik, hingga konsumsi alkohol [2]. Berdasarkan laporan Profil Kesehatan Indonesia 2023 [3], penyakit jantung merupakan penyebab kematian tertinggi kedua setelah stroke, dengan angka kematian penyakit jantung iskemik mencapai 95,68 per 100.000 penduduk. Secara keseluruhan, penyakit jantung dan stroke diperkirakan menyebabkan sekitar 1,5 juta kematian setiap tahunnya di

Indonesia, sehingga diperlukan upaya deteksi risiko dan pencegahan yang lebih efektif.

Namun, tingginya urgensi penanganan penyakit jantung di Indonesia belum diimbangi oleh ketersediaan data mikro (*unit-level data*) nasional di bidang kesehatan untuk mendukung analisis riset mendalam. Merujuk pada laporan Survei Kesehatan Indonesia (SKI) 2023 [4], informasi yang tersedia bagi publik masih terbatas pada ringkasan statistik di tingkat daerah. Selain itu, laman Badan Kebijakan Pembangunan Kesehatan (BKPK) Kementerian Kesehatan menyatakan bahwa akses terhadap data mikro SKI memerlukan mekanisme perizinan khusus dan tidak disediakan sebagai data *open access* bagi publik. Akibatnya, data mikro nasional level individu yang mengintegrasikan variabel demografis, fisiologis, dan gaya hidup dengan kondisi kesehatan responden belum dapat dimanfaatkan

secara luas dalam penelitian berbasis data. Kondisi ini membatasi pemanfaatan data kesehatan nasional untuk pengembangan dan evaluasi model analitik berbasis data berskala besar.

Meskipun terdapat tantangan dalam aksesibilitas data mikro nasional, pemanfaatan teknologi tetap menjadi langkah strategis yang perlu diambil untuk mendukung upaya deteksi dan pencegahan penyakit jantung secara dini. Menurut Pradana dkk. [5], pendekatan *data mining* berbasis algoritma *machine learning* dapat membantu menganalisis pola kesehatan yang kompleks untuk mendeteksi risiko penyakit secara akurat. Pernyataan tersebut sejalan dengan hasil penelitian oleh Nuraeni [6] yang menunjukkan bahwa model prediktif berbasis algoritma *machine learning* mampu memprediksi risiko penyakit jantung dengan tingkat akurasi yang tinggi. Secara umum, *data mining* diartikan sebagai proses penggalian pola atau informasi unik dari kumpulan data besar yang dapat digunakan sebagai dasar pengambilan keputusan. Salah satu tugas utama *data mining*, yaitu klasifikasi, sering kali dijadikan sebagai pendekatan utama dalam konteks medis karena mampu mengelompokkan data pasien berdasarkan kemiripan karakteristik tertentu, sehingga mendukung proses diagnosis dan prediksi penyakit [7].

Dalam konteks evaluasi kinerja algoritma *machine learning* untuk tugas klasifikasi, sejumlah penelitian terdahulu menunjukkan bahwa *Extreme Gradient Boosting* (XGBoost) merupakan algoritma yang efektif dan stabil di berbagai domain dan skala data. Penelitian oleh Dullah dkk. [8] menerapkan XGBoost untuk prediksi risiko penyakit jantung menggunakan 1.319 sampel pasien dengan sembilan fitur dan memperoleh akurasi sebesar 98,11%, yang meningkat menjadi 98,48% setelah menerapkan teknik *resampling Synthetic Minority Oversampling Technique* (SMOTE). Sementara itu, Velarde dkk. [9] menunjukkan keandalan XGBoost pada domain non-medis dalam tugas asesmen risiko pelanggan telekomunikasi dengan nilai F1-score sebesar 84% pada 1.000 sampel, 87% pada 10.000 sampel, dan 89% pada 100.000 sampel. Temuan-temuan tersebut menunjukkan bahwa XGBoost mampu memodelkan hubungan nonlinier antarfitur secara efektif serta mempertahankan kinerja yang konsisten ketika diterapkan pada dataset dengan ukuran dan karakteristik yang bervariasi.

Di sisi lain, sejumlah penelitian menunjukkan bahwa penerapan metode optimasi hiperparameter mampu meningkatkan kinerja model klasifikasi berbasis *machine learning*. Yaqin dkk. [10] melaporkan bahwa optimasi hiperparameter menggunakan Optuna pada algoritma Random Forest meningkatkan akurasi model dari 95% menjadi 98,62% pada dataset medis untuk klasifikasi risiko kanker paru-paru yang terdiri dari 309 sampel. Sementara itu, Sitanggung dan Sitompul [11] menerapkan Random Search untuk mengoptimasi Random Forest dalam tugas deteksi kelangsungan hidup pasien gagal jantung pada dataset yang terdiri dari 299 sampel dan berhasil meningkatkan akurasi model dari 80% menjadi 83,33%.

Optuna menerapkan pendekatan optimasi berbasis probabilistik yang secara adaptif mengeksplorasi ruang parameter dan memfokuskan pencarian pada kombinasi yang menjanjikan. Sebaliknya, Random Search melakukan eksplorasi parameter secara acak tanpa mekanisme adaptif, namun tetap lebih efisien dibandingkan pencarian menyeluruh karena tidak mengevaluasi seluruh kombinasi parameter. Oleh karena itu, kedua pendekatan tersebut umumnya dinilai lebih efisien secara komputasional dibandingkan metode konvensional seperti Grid Search, yang mengevaluasi seluruh kombinasi parameter secara sistematis sehingga memerlukan beban komputasi yang lebih besar.

Berdasarkan kajian terhadap beberapa penelitian terdahulu, terdapat sejumlah celah penelitian yang belum terjawab. Pertama, sejauh penelusuran penulis, belum ditemukan kajian yang membandingkan penerapan Optuna dan Random Search pada XGBoost dalam tugas klasifikasi risiko penyakit jantung. Kedua, pengujian kinerja XGBoost yang telah dioptimasi tersebut belum mencakup perbandingan berbagai pendekatan penanganan ketidakseimbangan kelas yang merepresentasikan pembobotan kelas, penambahan sampel minoritas, dan pengurangan sampel mayoritas. Ketiga, kajian yang mengevaluasi performa XGBoost teroptimasi pada dataset klasifikasi risiko penyakit jantung berskala besar relatif terbatas. Sejalan dengan keterbatasan tersebut, terdapat kebutuhan untuk memahami perbedaan kinerja XGBoost yang dioptimasi menggunakan Optuna dan Random Search, menilai pengaruh berbagai strategi penanganan ketidakseimbangan kelas terhadap performa model, serta mengkaji keberlakuan hasil optimasi dan strategi penanganan ketidakseimbangan kelas pada dataset berskala besar dalam konteks data kesehatan populasi.

Celah penelitian tersebut menjadi dasar perumusan tujuan penelitian ini, yaitu menganalisis dan membandingkan kinerja model XGBoost yang dioptimasi menggunakan Optuna dan Random Search dalam tugas klasifikasi risiko penyakit jantung berbasis dataset *Behavioral Risk Factor Surveillance System* (BRFSS) 2023. Selain itu, penelitian ini juga mengevaluasi pengaruh tiga pendekatan penanganan ketidakseimbangan kelas, yaitu pembobotan kelas (*class weighting*), penambahan sampel minoritas melalui *Synthetic Minority Oversampling Technique* (SMOTE), serta pengurangan sampel mayoritas melalui *Random Undersampling* (RUS), terhadap performa model. Dengan demikian, penelitian ini turut menguji konsistensi hasil optimasi hiperparameter dan strategi penanganan ketidakseimbangan kelas ketika diterapkan pada dataset BRFSS 2023.

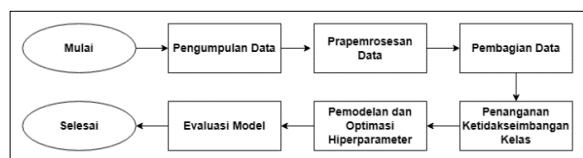
Sebagai respons terhadap keterbatasan akses data mikro kesehatan berskala individu di Indonesia, penelitian ini memanfaatkan dataset *Behavioral Risk Factor Surveillance System* (BRFSS) 2023 yang dikembangkan oleh *Centers for Disease Control and Prevention* (CDC) sebagai survei kesehatan populasi berskala besar di Amerika Serikat yang disediakan secara terbuka (*open access*) untuk keperluan riset publik. Dataset ini mencakup ratusan ribu entri responden

dengan ratusan variabel yang merepresentasikan faktor demografis, fisiologis, dan gaya hidup, termasuk variabel yang relevan terhadap risiko penyakit jantung seperti usia, hipertensi, diabetes, kebiasaan merokok, aktivitas fisik, dan konsumsi alkohol. Karakteristik dataset BRFSS yang besar dan variatif menjadikannya relevan jika digunakan sebagai dasar pengujian model klasifikasi *machine learning* serta evaluasi metode optimasi hiperparameter [12]. Dalam konteks tersebut, meskipun dataset ini tidak dikumpulkan dari populasi Indonesia, pemanfaatannya diposisikan sebagai pendekatan alternatif yang memungkinkan eksplorasi metodologis secara valid pada skala data medis yang besar.

Penelitian ini berkontribusi pada pemahaman metodologis mengenai bagaimana optimasi hiperparameter dan strategi penanganan ketidakseimbangan kelas memengaruhi kinerja model *machine learning* pada data medis berskala besar. Hasil penelitian ini menunjukkan batasan dan potensi masing-masing pendekatan optimasi dalam konteks klasifikasi risiko penyakit, sehingga dapat menjadi rujukan dalam pemilihan metode yang lebih tepat pada penelitian sejenis. Oleh karena itu, penelitian ini memberikan wawasan empiris mengenai pengembangan model prediksi risiko penyakit berbasis data besar dan membuka peluang bagi penelitian lanjutan di bidang kesehatan berbasis data.

II. METODE

Metode penelitian ini merujuk pada tahapan umum dalam proyek *data mining* yang disesuaikan dengan tujuan penelitian, meliputi pengumpulan data, prapemrosesan data, pembagian data, penanganan ketidakseimbangan kelas, pemodelan dan optimasi hiperparameter, serta evaluasi model. Urutan tahapan tersebut dirancang untuk memastikan bahwa setiap tahapan pengolahan dan analisis data dilakukan secara sistematis. Ilustrasi mengenai alur tahapan penelitian ditampilkan pada Gambar 1.



Gambar 1. Alur tahapan penelitian

A. Pengumpulan Data

Penelitian ini menggunakan data sekunder dari survei *Behavioral Risk Factor Surveillance System* (BRFSS) tahun 2023 yang diperoleh melalui laman resmi *Centers for Disease Control and Prevention* (CDC) dan tersedia secara terbuka untuk keperluan penelitian. Dataset BRFSS 2023 disediakan dalam format .xpt dan masih berupa data mentah karena belum melalui tahap prapemrosesan. Dataset ini mencakup sebanyak 433.233 entri responden dengan 350 variabel yang merepresentasikan karakteristik demografis, kondisi kesehatan, serta perilaku berisiko individu. Skala data yang besar dengan keragaman variabel yang luas menjadikan BRFSS memerlukan serangkaian penyesuaian sebelum

digunakan sebagai dasar pengembangan model untuk klasifikasi risiko penyakit tertentu.

B. Prapemrosesan Data

Tahap prapemrosesan data pada penelitian ini difokuskan untuk menangani karakteristik data BRFSS 2023 yang memiliki ruang lingkup yang luas dan masih bersifat mentah. Tujuan dari tahap ini adalah untuk memastikan bahwa model klasifikasi risiko penyakit jantung yang dikembangkan mampu menangkap pola informasi secara optimal tanpa terpengaruh oleh anomali maupun redundansi fitur [13]. Tahapan tersebut mencakup seleksi variabel, pembersihan data, serta transformasi dan standarisasi fitur, sehingga diperoleh dataset yang relevan dan representatif terhadap kondisi penyakit jantung.

1) *Persiapan Data Awal*: Tahap awal prapemrosesan dimulai dengan konversi dataset BRFSS 2023 dari format .xpt menjadi .csv untuk memudahkan proses identifikasi dan analisis data. Selanjutnya, dilakukan validasi tahun sampel data berdasarkan variabel IYEAR dengan merujuk pada *codebook* BRFSS 2023. Proses validasi ini bertujuan untuk memastikan bahwa seluruh data yang digunakan dalam proses analisis dan pengembangan model sepenuhnya berasal dari data tahun 2023.

2) *Pemilihan dan Pemetaan Nilai Variabel*: Proses pemilihan dan pemetaan nilai variabel masih mengacu pada *codebook* BRFSS 2023 guna memastikan konsistensi definisi variabel beserta kesesuaian interpretasi nilainya. Variabel yang relevan terhadap kondisi penyakit jantung dipilih dan diubah namanya dari yang semula berbentuk kode menjadi nama yang lebih deskriptif agar lebih mudah diinterpretasikan. Terdapat dua belas variabel terpilih yang meliputi RentangUsia, JenisKelamin, BMI, TekananDarahTinggi, KolesterolTinggi, Diabetes, KesulitanBerjalan, KesehatanUmum, StatusMerokok, AktivitasFisik, KonsumsiAlkohol, dan PenyakitJantung. Seluruh variabel tersebut kemudian dipetakan label nilainya berdasarkan definisi kategori yang tersedia dalam *codebook*. Variabel PenyakitJantung ditetapkan sebagai variabel target dengan label 0 untuk responden tanpa risiko dan label 1 untuk responden dengan risiko penyakit jantung.

3) *Pembersihan Data*: Tahap pembersihan data mencakup dua prosedur utama, yaitu eliminasi sampel dengan nilai hilang (*missing values*) serta penanganan data pencilan (*outlier*). Kedua prosedur tersebut diterapkan untuk memastikan dataset yang diperoleh setelah proses seleksi variabel sepenuhnya bebas dari anomali berupa ketidaklengkapan data maupun keberadaan nilai ekstrem yang berpotensi mengurangi kemampuan generalisasi model. Proses pembersihan data tidak mencakup identifikasi duplikasi sampel karena dataset BRFSS 2023 disusun pada tingkat individu responden, dengan setiap responden diidentifikasi berdasarkan nomor telepon, sehingga potensi keberadaan observasi ganda dianggap minimal.

Penanganan *missing values* dilakukan dengan menghapus baris data yang mengandung nilai hilang menggunakan fungsi $df = df.dropna()$. Langkah ini bertujuan agar seluruh observasi yang digunakan dalam pelatihan model memiliki informasi yang lengkap pada seluruh variabel yang dianalisis. Keberadaan nilai hilang dapat menyebabkan hilangnya informasi, memengaruhi representativitas data, serta berpotensi menimbulkan bias dalam analisis dan menurunkan kinerja model klasifikasi, sehingga penanganannya diperlukan untuk mengurangi dampak tersebut dalam tahap prapemrosesan data [1].

Selanjutnya, penanganan *outlier* diterapkan khusus pada variabel BMI yang merupakan satu-satunya variabel numerik berskala kontinu di antara dua belas variabel terpilih. Nilai BMI dibatasi pada rentang 12 hingga 60 kg/m² untuk mengurangi pengaruh nilai ekstrem yang jarang terjadi dan berpotensi mendistorsi struktur distribusi populasi dewasa sekaligus mempertahankan variasi biologis valid yang penting dalam konteks risiko penyakit jantung. Penanganan *outlier* memiliki peranan penting dalam tahap prapemrosesan data karena nilai ekstrem dapat mengganggu struktur distribusi data dan memengaruhi kemampuan model dalam mempelajari pola yang relevan [14]. Dengan demikian, pembatasan ini dimaksudkan untuk menjaga stabilitas distribusi data sehingga hubungan antara faktor risiko dan penyakit jantung dapat dipelajari secara lebih representatif oleh model. Meskipun secara statistik beberapa observasi di atas batas atas *interquartile range* (IQR) tetap terdeteksi sebagai *outlier*, nilai-nilai tersebut tetap dipertahankan karena relevansi klinisnya dalam merepresentasikan obesitas ekstrem sebagai faktor risiko penyakit jantung, sehingga representativitas data tetap dapat dijaga tanpa mengorbankan stabilitas distribusi.

4) *Penerapan One-Hot Encoding dan Standarisasi Data*: *One-hot encoding* selanjutnya diterapkan setelah data melalui tahap pembersihan untuk mengonversi variabel kategorik ke dalam representasi numerik berbentuk fitur biner [15]. Proses ini diterapkan khusus pada variabel kategorik meliputi RentangUsia, KesehatanUmum, dan StatusMerokok, di mana StatusMerokok bersifat nominal, sedangkan RentangUsia dan KesehatanUmum memiliki sifat ordinal namun tetap dikodekan secara biner untuk meminimalkan asumsi hubungan linear antar tingkatan kategori dan memungkinkan setiap kategori dimanfaatkan secara independen oleh model klasifikasi risiko penyakit jantung. Pemecahan variabel kategorik menjadi beberapa fitur biner membuat setiap kategori direpresentasikan secara terpisah, sehingga meningkatkan kejelasan representasi data meskipun jumlah fitur dalam dataset bertambah.

Setelah penerapan *one-hot encoding*, seluruh fitur yang tersedia dalam dataset distandarisasi menggunakan *StandardScaler*, termasuk fitur biner hasil pemecahan variabel kategorik serta fitur numerik kontinu seperti BMI. Standarisasi ini dilakukan untuk menyeragamkan skala nilai antar variabel sehingga perbedaan rentang numerik tidak diinterpretasikan oleh model sebagai perbedaan tingkat

kepentingan fitur [16]. Pendekatan ini relevan mengingat fitur dalam dataset memiliki karakteristik yang bervariasi, sehingga tanpa standarisasi, fitur dengan rentang nilai yang lebih besar berpotensi memengaruhi proses pembelajaran secara tidak proporsional. Meskipun algoritma XGBoost relatif toleran terhadap variasi skala fitur, standarisasi tetap diterapkan untuk memastikan konsistensi representasi data dan stabilitas proses pelatihan.

C. Pembagian Data

Dataset BRFSS 2023 yang telah melalui seluruh tahap prapemrosesan dibagi menjadi data latih dan data uji untuk memisahkan proses pembelajaran dan evaluasi model. Pembagian data dilakukan dengan rasio 80% sebagai data latih dan 20% sebagai data uji yang secara umum digunakan dalam penelitian pengembangan model klasifikasi berbasis algoritma *machine learning*. Skema ini dirancang untuk menghasilkan evaluasi kinerja model yang objektif melalui pengujian pada data yang tidak terlibat dalam proses pelatihan.

D. Penanganan Ketidakseimbangan Kelas

Meskipun data telah melalui tahap prapemrosesan dan pembagian data, distribusi kelas pada variabel target masih menunjukkan ketidakseimbangan antara kelas mayoritas dan kelas minoritas. Berdasarkan identifikasi awal, sekitar 90,7% data berada pada kelas mayoritas (kelas 0), dengan 9,3% sisanya berada pada kelas minoritas (kelas 1). Ketidakseimbangan distribusi kelas ini berpotensi menyebabkan proses pembelajaran model menjadi bias terhadap kelas mayoritas, yang membuat model lebih dominan mempelajari pola dari kelas mayoritas dan kurang optimal dalam mengenali karakteristik kelas minoritas [13]. Oleh karena itu, diperlukan penerapan strategi penanganan ketidakseimbangan kelas sebelum tahap pemodelan.

Penelitian ini menerapkan tiga pendekatan penanganan ketidakseimbangan kelas yang merepresentasikan mekanisme penanganan yang berbeda. Pendekatan tersebut meliputi *class weighting* yang bekerja melalui pemberian bobot berbeda antar kelas dalam proses pembelajaran, *Synthetic Minority Oversampling Technique* (SMOTE) yang bekerja dengan menambahkan sampel pada kelas minoritas, serta *Random Undersampling* (RUS) yang bekerja dengan mengurangi sampel pada kelas mayoritas [17], [18], [19]. Ketiga pendekatan ini digunakan sebagai skenario yang berbeda untuk mengeksplorasi dan membandingkan efektivitas masing-masing mekanisme pada dataset BRFSS 2023 yang berukuran besar dan heterogen.

Penanganan ketidakseimbangan kelas hanya diterapkan pada data latih yang mencakup 80% dari data bersih, sementara itu data uji dibiarkan mempertahankan distribusi kelas aslinya. Pendekatan ini bertujuan untuk mencegah terjadinya *data leakage*, yaitu kondisi ketika informasi dari data uji secara tidak langsung memengaruhi proses pelatihan model dan menghasilkan estimasi kinerja yang terlalu optimistis [18]. Selain itu, mempertahankan distribusi kelas

asli pada data uji memungkinkan evaluasi dilakukan pada kondisi yang merepresentasikan situasi nyata, sehingga kinerja model yang diperoleh mencerminkan kemampuan generalisasi secara lebih objektif.

E. Pemodelan dan Optimasi Hiperparameter

Tahap pemodelan dilakukan dengan melatih algoritma *Extreme Gradient Boosting* (XGBoost) menggunakan data latih yang telah melalui seluruh tahapan prapemrosesan. Proses pemodelan dilakukan baik pada model dasar XGBoost maupun pada model yang dioptimasi menggunakan dua pendekatan optimasi hiperparameter, yaitu Optuna dan Random Search. Kedua pendekatan optimasi tersebut juga diterapkan pada tiga skenario penanganan ketidakseimbangan kelas, yang meliputi *class weighting*, *Synthetic Minority Oversampling Technique* (SMOTE), dan *Random Undersampling* (RUS), sehingga menghasilkan sembilan kombinasi pemodelan. Seluruh kombinasi ini digunakan untuk mengeksplorasi pengaruh strategi optimasi dan penanganan ketidakseimbangan kelas terhadap kinerja model klasifikasi risiko penyakit jantung secara menyeluruh dan representatif melalui mekanisme penanganan yang berbeda, dengan konfigurasi masing-masing skenario dirangkum pada Tabel 1.

TABEL I
SKENARIO PEMODELAN DAN KONFIGURASI MODEL XGBOOST

Kode Skenario	Konfigurasi Model
BaseWeight	XGBoost + <i>Class Weighting</i>
OptunaWeight	XGBoost + Optuna + <i>Class Weighting</i>
RandomWeight	XGBoost + Random Search + <i>Class Weighting</i>
BaseSMOTE	XGBoost + SMOTE
OptunaSMOTE	XGBoost + Optuna + SMOTE
RandomSMOTE	XGBoost + Random Search + SMOTE
BaseRUS	XGBoost + <i>Random Undersampling</i> (RUS)
OptunaRUS	XGBoost + Optuna + <i>Random Undersampling</i> (RUS)
RandomRUS	XGBoost + Random Search + <i>Random Undersampling</i> (RUS)

Dalam seluruh skenario pemodelan, konfigurasi dasar XGBoost diterapkan secara konsisten, dengan nilai parameter yang meliputi *random_state=42*, *eval_metric='logloss'*, dan *tree_method='hist'*. Pada skenario *class weighting*, nilai *scale_pos_weight* ditetapkan sesuai dengan rasio ketidakseimbangan kelas yang relevan pada data latih, sedangkan pada skenario SMOTE dan RUS, *scale_pos_weight* ditetapkan sebesar 1 untuk menghindari redundansi mekanisme penanganan ketidakseimbangan kelas. Pencarian parameter optimal dilakukan melalui Optuna dan Random Search yang masing-masing memiliki jumlah iterasi sebanyak 50 kali untuk menjamin perbandingan performa yang adil. Proses pelatihan dan optimasi dievaluasi menggunakan *5-fold stratified cross-validation* dengan metrik *Receiver Operating Characteristic-Area Under Curve* (ROC-AUC). Rata-rata ROC-AUC dipilih sebagai acuan

utama pemilihan konfigurasi terbaik karena bersifat *threshold-independent*, sehingga memberikan estimasi generalisasi model yang stabil sebelum evaluasi komprehensif dengan metrik klasifikasi lengkap pada data uji.

1) *Extreme Gradient Boosting* (XGBoost): *Extreme Gradient Boosting* (XGBoost) merupakan algoritma *machine learning* berbasis *gradient boosting* yang menggunakan *decision tree* sebagai model dasarnya. Algoritma ini bekerja secara berurutan, di mana setiap pohon baru dibangun untuk memperbaiki kesalahan prediksi dari pohon sebelumnya sehingga model akhir yang dihasilkan merupakan gabungan dari seluruh pohon dalam proses *ensemble*. XGBoost dikenal memiliki efisiensi komputasi yang tinggi, kemampuan generalisasi yang baik, serta mekanisme regulasi untuk mencegah *overfitting* [20], [21]. Disebutkan pada penelitian oleh Zhang dkk. [22], algoritma XGBoost memiliki beberapa komponen utama dalam proses kerjanya yang dijabarkan melalui sejumlah persamaan sebagai berikut.

Rumus prediksi (*ensemble model*) XGBoost ditunjukkan pada Rumus (1):

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i), \quad f_k \in \mathcal{F} \quad (1)$$

Keterangan Rumus (1):

\hat{y}_i = prediksi sampel ke i .

K = jumlah pohon dalam *ensemble*.

$f_k(\cdot)$ = pohon keputusan ke k .

\mathcal{F} = ruang fungsi pohon keputusan.

Fungsi objektif yang digunakan dalam proses pelatihan XGBoost ditunjukkan pada Rumus (2), dengan komponen regularisasi yang dirumuskan pada Rumus (3):

$$\mathcal{L} = \sum_{i=1}^n l(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (2)$$

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \|w\|^2 \quad (3)$$

Keterangan Rumus (2) dan Rumus (3):

\mathcal{L} = fungsi objektif total.

$l(y_i, \hat{y}_i)$ = *loss* antara aktual dan prediksi.

$\Omega(f_k)$ = regulasi pohon ke k .

T = jumlah daun pohon.

w = bobot tiap daun.

γ = penalti penambahan daun.

λ = regulasi L2 bobot daun.

Rumus (1) menjelaskan bahwa prediksi akhir pada model XGBoost diperoleh dari penjumlahan kontribusi seluruh pohon keputusan yang dibangun secara bertahap melalui proses *boosting*. Setiap pohon baru berfungsi untuk memperbaiki kesalahan prediksi dari pohon sebelumnya, sehingga model akhir merupakan hasil *ensemble* dari seluruh pohon yang telah dilatih.

Sementara itu, Rumus (2) dan (3) menggambarkan tujuan pelatihan model XGBoost secara konseptual, yaitu meminimalkan kesalahan prediksi sekaligus mengendalikan kompleksitas model melalui komponen regularisasi. Komponen regularisasi berfungsi memberikan penalti

terhadap pertumbuhan struktur pohon, seperti penambahan jumlah daun, serta membatasi bobot pada setiap daun agar model tidak menjadi terlalu kompleks. Dengan menyeimbangkan antara pengurangan kesalahan dan pembatasan kompleksitas tersebut, XGBoost mampu menghasilkan model yang akurat namun tetap terkontrol sehingga mengurangi potensi *overfitting*.

2) *Optuna*: *Optuna* merupakan *framework* optimasi hiperparameter yang mendukung pendekatan optimasi berbasis model, salah satunya melalui algoritma *Tree-structured Parzen Estimator* (TPE). *Framework* ini bekerja dengan menganalisis hasil evaluasi sebelumnya untuk mempelajari hubungan antara konfigurasi parameter dan performa model, sehingga dapat menentukan arah pencarian parameter yang lebih potensial. Pendekatan adaptif ini menjadikan proses pencarian lebih efisien dibandingkan metode konvensional seperti *Grid Search*. Dengan memfokuskan pencarian pada wilayah ruang parameter yang menjanjikan, *Optuna* mampu meningkatkan efisiensi pencarian sekaligus menghasilkan konfigurasi yang lebih optimal [10].

Selain efisien, *Optuna* juga memiliki fleksibilitas yang tinggi karena mendukung konsep *define-by-run*, yang memungkinkan pengguna mendefinisikan ruang pencarian secara dinamis selama proses eksekusi. Selain itu, mekanisme *pruning* yang dimilikinya dapat menghentikan percobaan yang tidak menjanjikan sejak dini, sehingga sumber daya komputasi dapat digunakan dengan lebih optimal. Kombinasi metode TPE, *define-by-run*, dan *pruning* menjadikan *Optuna* efektif digunakan dalam berbagai penelitian, termasuk pada model *machine learning* maupun *deep learning* yang memerlukan penyesuaian hiperparameter secara optimal [23].

3) *Random Search*: Di sisi lain, *Random Search* merupakan metode optimasi hiperparameter yang bekerja dengan cara memilih kombinasi parameter secara acak dari ruang pencarian yang telah ditentukan. Berbeda dengan *Grid Search* yang mengevaluasi seluruh kemungkinan kombinasi, *Random Search* hanya menguji sebagian konfigurasi, sehingga proses pencarian menjadi lebih efisien dengan waktu komputasi lebih singkat. Pendekatan ini efektif diterapkan pada model dengan dataset berukuran besar dan fitur yang kompleks, karena eksplorasi acak memungkinkan pencarian solusi yang baik tanpa harus mengevaluasi seluruh kombinasi, sehingga beban komputasi dapat dikurangi secara signifikan [24].

4) *Ruang Lingkup Optimasi Hiperparameter*: Pada skenario pemodelan yang melibatkan optimasi hiperparameter menggunakan *Optuna* dan *Random Search*, proses optimasi difokuskan pada hiperparameter utama XGBoost yang memengaruhi kompleksitas dan kemampuan generalisasi model. Ruang lingkup nilai hiperparameter pada kedua pendekatan dikonfigurasi secara identik untuk menjaga kesetaraan ruang pencarian. Perbedaan hanya terletak pada mekanisme eksplorasi, dengan *Optuna* melakukan pencarian adaptif pada ruang parameter kontinu

yang lebih halus berdasarkan hasil evaluasi sebelumnya, sedangkan *Random Search* mengevaluasi kombinasi nilai diskret secara acak. Dengan demikian, variasi kinerja yang dihasilkan dapat dipandang sebagai konsekuensi langsung dari perbedaan strategi pencarian dalam ruang parameter yang setara. Rincian ruang lingkup hiperparameter ditunjukkan pada Tabel 2.

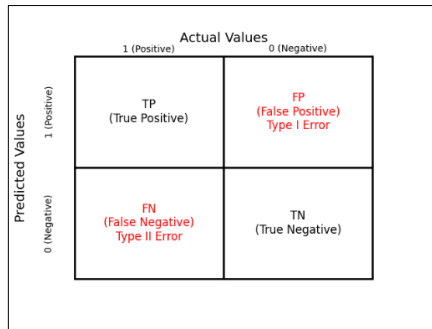
TABEL II
RUANG LINGKUP OPTIMASI HIPERPARAMETER XGBOOST

Hiperparameter	Optuna	Random Search
<i>n_estimators</i>	100–1000	{100, 300, 500, 1000}
<i>max_depth</i>	2–20	{3, 6, 10, 15}
<i>learning_rate</i>	0.001–0.3	{0.01, 0.05, 0.1, 0.2}
<i>subsample</i>	0.3–1.0	{0.6, 0.8, 1.0}
<i>colsample_bytree</i>	0.3–1.0	{0.6, 0.8, 1.0}

F. Evaluasi Model

Setelah proses pelatihan model selesai, evaluasi dilakukan dengan mengujikan seluruh skenario pemodelan menggunakan data uji untuk menilai kinerja model dalam tugas klasifikasi risiko penyakit jantung. Tahap evaluasi merupakan bagian penting dalam pengembangan model *machine learning* karena berfungsi untuk menilai efektivitas, reliabilitas, dan kemampuan generalisasi, sekaligus mengidentifikasi kekuatan dan kelemahan model. Penilaian dilakukan menggunakan metrik *confusion matrix*, akurasi, dan nilai *macro average* dari presisi, *recall*, serta F1-score, dilengkapi visualisasi kurva ROC-AUC untuk memberikan gambaran menyeluruh mengenai kualitas prediksi model [16], [25]. Selain itu, proses evaluasi mencakup analisis *feature importance* untuk menilai kontribusi relatif masing-masing fitur terhadap keputusan model.

1) *Confusion Matrix*: *Confusion matrix* merupakan tabel evaluasi yang membandingkan hasil prediksi model dengan kondisi aktual data uji. Tabel ini terdiri atas empat komponen utama, yaitu *True Positive* (TP) dan *True Negative* (TN) yang merepresentasikan prediksi benar untuk kelas positif dan negatif, serta *False Positive* (FP) dan *False Negative* (FN) yang menunjukkan kesalahan klasifikasi, di mana FP terjadi ketika model memprediksi kelas positif pada data yang sebenarnya negatif, sedangkan FN terjadi ketika model memprediksi kelas negatif pada data yang sebenarnya positif [26]. Pada penelitian yang dilakukan oleh Hakim dkk. [26], evaluasi kinerja model *machine learning* menggunakan *confusion matrix* divisualisasikan pada Gambar 2 untuk menunjukkan kontribusi masing-masing komponen terhadap evaluasi model.

Gambar 2. Ilustrasi *confusion matrix*

Berdasarkan visualisasi *confusion matrix* pada Gambar 2, komponen utama yang meliputi *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) digunakan sebagai dasar dalam perhitungan berbagai metrik evaluasi. Berdasarkan komponen-komponen tersebut, metrik seperti akurasi, presisi, *recall*, dan F1-score dapat dihitung untuk menilai kualitas prediksi model. Uraian mengenai rumus masing-masing metrik disajikan pada bagian berikut.

Rumus akurasi dijabarkan pada Rumus (4):

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

Pada Rumus (4), akurasi merepresentasikan proporsi prediksi yang benar terhadap keseluruhan data uji. Semakin tinggi nilai akurasi, semakin baik kemampuan model dalam melakukan klasifikasi secara keseluruhan. Meskipun demikian, metrik ini belum tentu mampu merepresentasikan performa model secara seimbang antar kelas, khususnya ketika dataset memiliki distribusi kelas yang tidak seimbang.

Rumus presisi dijabarkan pada Rumus (5):

$$Presisi = \frac{TP}{TP+FP} \quad (5)$$

Pada Rumus (5), presisi merepresentasikan tingkat ketepatan model dalam memprediksi kelas positif. Semakin tinggi nilai presisi, semakin besar proporsi prediksi positif yang benar-benar sesuai dengan fakta, sehingga kesalahan klasifikasi berupa *False Positive* (FP) dapat diminimalkan. Meskipun demikian, presisi yang tinggi tidak selalu menjamin bahwa model mampu menangkap seluruh data positif yang terdapat dalam dataset.

Rumus *recall* dijabarkan pada Rumus (6):

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

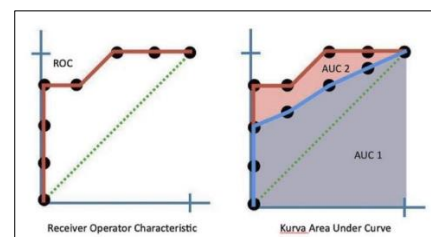
Pada Rumus (6), *recall* merepresentasikan kemampuan model dalam mengidentifikasi seluruh data positif yang terdapat dalam dataset. Nilai *recall* yang tinggi menunjukkan bahwa model mampu meminimalkan kesalahan prediksi *False Negative* (FN), sehingga data positif yang penting tidak terlewatkan. Meskipun demikian, nilai *recall* yang tinggi tetap perlu diimbangi dengan presisi agar performa prediksi model tetap stabil dan akurat.

Rumus F1-score dijabarkan pada Rumus (7):

$$F1\text{-score} = 2 \times \frac{Presisi \times Recall}{Presisi + Recall} \quad (7)$$

Pada Rumus (7), F1-score merupakan rata-rata harmonis antara presisi dan *recall* yang digunakan untuk menilai keseimbangan performa model dalam proses klasifikasi. Nilai F1-score yang tinggi mengindikasikan bahwa model tidak hanya tepat dalam memprediksi kelas positif, tetapi juga konsisten dalam mendeteksi seluruh data positif yang relevan. Oleh karena itu, metrik ini sangat berguna pada kondisi ketika distribusi kelas tidak seimbang.

2) *Grafik ROC-AUC*: Berbeda dengan *confusion matrix* yang menyajikan ringkasan kinerja model dalam bentuk nilai-nilai klasifikasi, *Receiver Operating Characteristic* (ROC) merupakan grafik evaluasi yang digunakan untuk menilai kinerja model klasifikasi biner. Kurva ROC menggambarkan hubungan antara *True Positive Rate* (TPR) dan *False Positive Rate* (FPR) yang diturunkan dari komponen pada *confusion matrix*. Dengan memvisualisasikan perubahan nilai TPR dan FPR pada berbagai ambang keputusan (*threshold*), kurva ROC membantu menggambarkan keseimbangan antara kemampuan model dalam mendeteksi kelas positif dan tingkat kesalahan prediksi terhadap kelas negatif, sehingga memberikan pemahaman kinerja model yang lebih jelas dan informatif [27]. Disebutkan pada penelitian oleh Kristiawan dan Widjaja [28], visualisasi grafik ROC ditampilkan pada Gambar 3.



Gambar 3. Ilustrasi grafik ROC-AUC

Dilihat pada Gambar 3, grafik ROC tidak hanya menampilkan hubungan antara *True Positive Rate* (TPR) dan *False Positive Rate* (FPR), tetapi juga dilengkapi dengan komponen tambahan berupa *Area Under Curve* (AUC). Pada grafik ini, sumbu Y merepresentasikan TPR, sedangkan sumbu X merepresentasikan FPR. Luas area di bawah kurva ROC (AUC) digunakan untuk menilai kinerja model klasifikasi secara keseluruhan. Nilai AUC banyak digunakan sebagai metrik evaluasi karena mampu mencerminkan performa model pada berbagai ambang batas keputusan (*threshold*), sehingga memberikan gambaran kinerja yang lebih komprehensif dibandingkan metrik tunggal dari *confusion matrix* [28]. Adapun klasifikasi performa nilai AUC menurut Amrin dkk. [29] disajikan pada kategori berikut sebagai referensi penilaian model.

Disebutkan pada penelitian oleh Amrin dkk. [29], nilai AUC dikategorikan ke dalam beberapa tingkatan, yang meliputi rentang nilai 0,90–1,00 sebagai klasifikasi sangat

baik, 0,80–0,90 sebagai klasifikasi baik, 0,70–0,80 sebagai klasifikasi cukup, 0,60–0,70 sebagai klasifikasi kurang, serta 0,50–0,60 sebagai klasifikasi gagal. Kategori tersebut memberikan gambaran mengenai tingkat kualitas pemisahan kelas yang dihasilkan oleh model klasifikasi berdasarkan nilai AUC. Dengan demikian, nilai AUC dapat digunakan sebagai acuan evaluatif untuk menilai performa model secara umum dalam melakukan tugas klasifikasi.

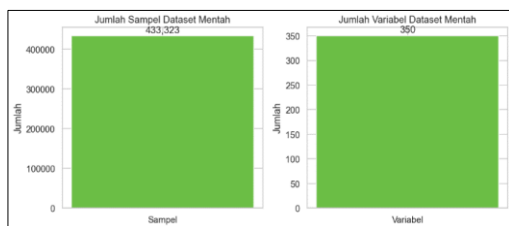
3) *Feature Importance: Feature importance* dalam model *machine learning* merupakan pendekatan yang digunakan untuk menjelaskan kontribusi relatif setiap fitur terhadap kinerja model dalam menghasilkan prediksi. Melalui analisis *feature importance*, dapat diketahui fitur-fitur yang memiliki pengaruh paling signifikan maupun yang memberikan kontribusi rendah dalam proses pengambilan keputusan model. Informasi yang dihasilkan dari *feature importance* membantu meningkatkan pemahaman terhadap cara kerja model serta mengidentifikasi faktor-faktor utama yang memengaruhi hasil prediksi [13].

III. HASIL DAN PEMBAHASAN

Secara keseluruhan, semua tahapan dalam penelitian ini diimplementasikan menggunakan bahasa pemrograman Python. Tahap prapemrosesan data awal dilakukan menggunakan Visual Studio Code, yang meliputi proses konversi data, pemilihan dan pemetaan nilai variabel, serta pembersihan data. Selanjutnya, prapemrosesan lanjutan dilakukan pada platform Google Colaboratory yang mencakup *one-hot encoding* dan standarisasi data. Pada platform yang sama, proses dilanjutkan dengan pembagian data latih dan data uji, penanganan ketidakseimbangan kelas, pemodelan, hingga evaluasi model. Seluruh skenario pemodelan kemudian dievaluasi untuk menilai kemampuan masing-masing model dalam melakukan tugas klasifikasi risiko penyakit jantung.

A. Dataset

Pada kondisi awal setelah melalui proses konversi data, dataset BRFSS 2023 masih merepresentasikan kondisi data mentah sebagaimana yang ditunjukkan pada Gambar 4. Visualisasi tersebut memberikan gambaran awal mengenai struktur dan dimensi dataset sebelum dilakukan tahap prapemrosesan dan seleksi variabel, sehingga seluruh variabel dan sampel masih tersedia secara utuh serta belum secara spesifik disesuaikan dengan konteks klasifikasi risiko penyakit jantung.



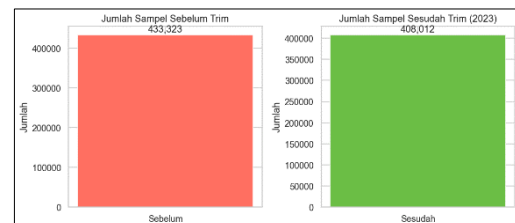
Gambar 4. Dimensi awal dataset BRFSS 2023

Pada Gambar 4, ditunjukkan bahwa dataset terdiri dari total 433.323 sampel dengan 350 variabel. Meskipun mayoritas data berasal dari tahun 2023, terdapat sekitar 25 ribu entri yang tercatat sebagai data tahun 2024 setelah diidentifikasi menggunakan variabel IYEAR. Berdasarkan *codebook* BRFSS 2023, variabel-variabel yang tersedia merepresentasikan indikator faktor demografis, kondisi medis, serta perilaku berisiko responden yang disurvei.

B. Hasil Prapemrosesan Data

Keseluruhan tahap prapemrosesan pada dataset BRFSS 2023 menghasilkan data yang lebih bersih, relevan, dan siap digunakan untuk pembangunan model klasifikasi risiko penyakit jantung. Setelah dilakukan seleksi variabel relevan, pembersihan entri bermasalah, serta transformasi dan standarisasi fitur, struktur dataset menjadi lebih representatif terhadap kondisi penyakit jantung. Dengan demikian, dataset yang dihasilkan memiliki fitur yang lebih terfokus dengan sampel yang berkualitas, sehingga mendukung proses pemodelan secara efektif.

1) *Persiapan Data Awal:* Sebagai upaya untuk memastikan akurasi representasi tahun dalam dataset, sampel BRFSS 2023 difilter berdasarkan variabel IYEAR sehingga hanya observasi tahun 2023 yang dilibatkan pada tahap prapemrosesan selanjutnya. Tahap ini dilakukan sebelum proses pemilihan variabel, penamaan ulang, dan pemetaan nilai. Perubahan jumlah sampel sebagai hasil dari seleksi ini disajikan pada Gambar 5.



Gambar 5. Jumlah sampel sebelum dan setelah seleksi IYEAR

Gambar 5 memvisualisasikan pengurangan jumlah sampel selama tahap seleksi awal. Berdasarkan total 433.233 entri yang tersedia, identifikasi melalui variabel IYEAR menunjukkan terdapat 25.221 entri yang berasal dari data tahun 2024. Seluruh entri tersebut kemudian dieliminasi guna menjaga homogenitas data, sehingga diperoleh dataset akhir sebanyak 408.012 sampel. Langkah ini memastikan bahwa analisis selanjutnya sepenuhnya merepresentasikan data BRFSS tahun 2023 sesuai dengan fokus penelitian.

2) *Pemilihan dan Pemetaan Nilai Variabel:* Mengingat dataset awal memiliki total 350 variabel, proses seleksi dilakukan untuk menentukan variabel yang relevan dengan tujuan penelitian. Tahap ini mencakup seleksi variabel, penamaan ulang, serta pemetaan nilai yang berpedoman pada *codebook* BRFSS 2023. Rincian mengenai variabel terpilih,

beserta jenis, deskripsi, dan hasil pemetaan nilainya disajikan pada Tabel 3.

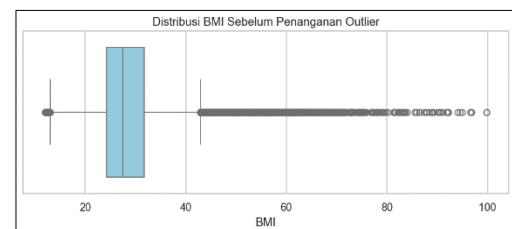
TABEL III
RINCIAN VARIABEL TERPILIH BESERTA JENIS, DESKRIPSI, DAN HASIL PEMETAAN NILAINYA

Nama Variabel	Jenis	Deskripsi	Pemetaan Nilai
RentangUsia	Kategorik	Kelompok usia responden (tahun)	1=18–24, 2=25–34, 3=35–44, 4=45–54, 5=55–64, 6≥65
JenisKelamin	Biner	Jenis kelamin responden	0=Perempuan, 1=Laki-laki
BMI	Numerik	Indeks massa tubuh responden (kg/m ²)	Rentang nilai numerik
TekananDarahTinggi	Biner	Riwayat hipertensi responden	0=Tidak, 1=Ya
KolesterolTinggi	Biner	Riwayat kolesterol tinggi responden	0=Tidak, 1=Ya
Diabetes	Biner	Riwayat diabetes responden	0=Tidak, 1=Ya
KesulitanBerjalan	Biner	Kesulitan berjalan pada responden	0=Tidak, 1=Ya
KesehatanUmum	Kategorik	Persepsi responden terhadap kondisi kesehatannya	1–5 (sangat baik–buruk)
StatusMerokok	Kategorik	Riwayat merokok responden	0–3 (tidak pernah, setiap hari, kadang, mantan)
AktivitasFisik	Biner	Aktivitas fisik intens responden selama satu bulan terakhir	0=Tidak, 1=Ya
KonsumsiAlkohol	Biner	Riwayat konsumsi alkohol responden	0=Tidak, 1=Ya
PenyakitJantung	Biner	Indikator risiko penyakit jantung pada responden	0=Tidak berisiko, 1=Berisiko

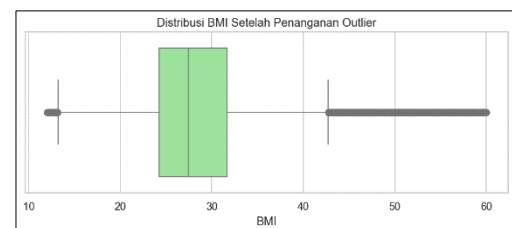
Berdasarkan pemaparan pada Tabel 3, dua belas variabel terpilih terdiri atas tiga variabel kategorik (RentangUsia, KesehatanUmum, dan StatusMerokok), delapan variabel biner (JenisKelamin, TekananDarahTinggi, KolesterolTinggi, Diabetes, KesulitanBerjalan, AktivitasFisik, KonsumsiAlkohol, dan PenyakitJantung), serta satu variabel numerik, yaitu BMI. Variabel PenyakitJantung ditetapkan sebagai variabel target dengan dua kelas, yaitu berisiko dan tidak berisiko, sedangkan sebelas variabel lainnya berperan sebagai variabel prediktor.

Pemilihan variabel dalam penelitian ini dilakukan melalui observasi terhadap studi oleh Firmansyah dan Yulianto [12] yang menggunakan dataset BRFSS 2015 sebagai dasar pengembangan model prediksi risiko penyakit jantung dengan 22 variabel. Berdasarkan hasil observasi tersebut, penelitian ini memilih dua belas variabel yang dinilai paling relevan dan representatif dalam menggambarkan karakteristik individu yang berkaitan dengan risiko penyakit jantung.

3) *Pembersihan Data*: Setelah dataset mencakup dua belas variabel terpilih beserta pemetaan nilai yang ditetapkan, tahap pembersihan data dilakukan untuk memastikan kelengkapan dan kewajaran nilai pada setiap sampel. Pada tahap ini, seluruh sampel yang teridentifikasi memiliki nilai hilang (*missing values*) dieliminasi dari dataset. Selanjutnya, penanganan nilai ekstrem (*outlier*) diterapkan pada variabel BMI sebagai satu-satunya variabel numerik dengan membatasi nilai BMI pada rentang 12 hingga 60. Dengan demikian, sampel yang memiliki nilai BMI di luar rentang tersebut turut dieliminasi. Visualisasi distribusi nilai BMI sebelum dan setelah penanganan *outlier* disajikan pada Gambar 6 dan Gambar 7.



Gambar 6. Distribusi nilai BMI sebelum penanganan *outlier*

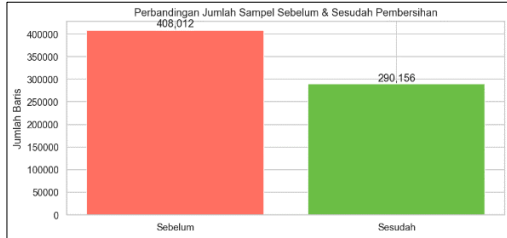


Gambar 7. Distribusi nilai BMI setelah penanganan *outlier*

Gambar 6 memvisualisasikan sebaran nilai BMI sebelum dilakukan penanganan *outlier*, di mana terlihat adanya nilai-nilai ekstrem yang memanjang secara signifikan hingga mendekati angka 100. Sementara itu pada Gambar 7, ditunjukkan hasil distribusi setelah dilakukan pembatasan nilai pada rentang 12 hingga 60, yang menyebabkan persebaran data menjadi lebih terkonsentrasi dan representatif, dengan sisa *outlier* yang hanya terdeteksi pada kisaran nilai di bawah 15 serta pada rentang 45 hingga 60.

Meskipun nilai BMI telah dibatasi pada rentang 12–60, hasil deteksi *outlier* menggunakan pendekatan *interquartile range* (IQR) menunjukkan masih terdapat 9.275 observasi yang teridentifikasi sebagai *outlier*, dengan batas bawah sebesar 13,18 dan batas atas sebesar 42,75. Observasi-observasi tersebut mayoritas berada pada kisaran BMI tinggi dan membentuk ekor distribusi, yang mencerminkan heterogenitas distribusi data, sehingga tetap dipertahankan

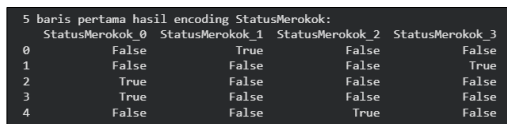
dalam analisis meskipun terdeteksi sebagai *outlier* secara statistik. Setelah melalui tahap pembersihan data yang meliputi eliminasi sampel dengan nilai hilang serta penanganan *outlier*, perbandingan jumlah sampel sebelum dan sesudah pembersihan ditampilkan pada Gambar 8.



Gambar 8. Perbandingan jumlah sampel sebelum dan setelah pembersihan

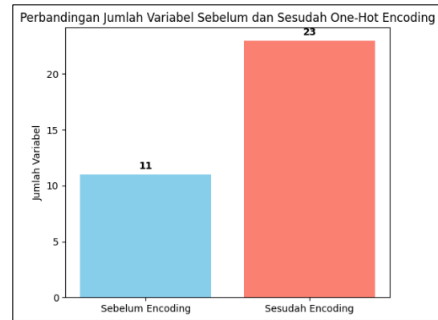
Gambar 8 menunjukkan perubahan jumlah sampel pada dataset, dari yang semula berjumlah 408.012 menjadi 290.156 setelah melalui tahap pembersihan data. Pengurangan tersebut merupakan hasil dari eliminasi sebanyak 117.856 sampel yang teridentifikasi memiliki nilai hilang serta nilai ekstrem (*outlier*). Sebagai tahap akhir pembersihan, variabel biner dan kategorik dalam dataset ditetapkan dalam tipe int8, sedangkan BMI sebagai satu-satunya variabel numerik ditetapkan dalam tipe float32 untuk efisiensi penyimpanan dan komputasi.

4) *Penerapan One-hot Encoding*: *One-hot encoding* kemudian diterapkan pada variabel kategorik setelah seluruh data melalui tahap pembersihan. Variabel yang dikodekan meliputi RentangUsia, KesehatanUmum, dan StatusMerokok, di mana setiap kategori diubah menjadi variabel biner yang berdiri sendiri. Sebagai ilustrasi hasil transformasi data setelah proses *encoding*, hasil penerapan *one-hot encoding* pada variabel StatusMerokok ditampilkan pada Gambar 9.



Gambar 9. Hasil *encoding* pada variabel StatusMerokok

Gambar 9 menunjukkan lima sampel pertama hasil *encoding* pada variabel StatusMerokok. Hasil pemecahan variabel ini menghasilkan kolom baru berupa StatusMerokok_0, StatusMerokok_1, StatusMerokok_2, dan StatusMerokok_3, di mana setiap kolom merepresentasikan masing-masing kategori menjadi format biner (*True* = benar, *False* = salah). Ilustrasi ini memperlihatkan bagaimana proses *one-hot encoding* mengubah data kategorik menjadi representasi numerik biner yang siap digunakan dalam analisis selanjutnya. Dikarenakan proses *encoding* bekerja dengan memecah variabel kategorik menjadi beberapa variabel baru, jumlah variabel dalam dataset turut bertambah, sebagaimana yang ditunjukkan pada Gambar 10.



Gambar 10. Perbandingan jumlah variabel sebelum dan setelah *encoding*

Gambar 10 menampilkan perbandingan jumlah variabel prediktor dalam dataset sebelum dan sesudah penerapan *one-hot encoding* pada variabel kategorik. Sebelum proses *encoding*, dataset terdiri dari 11 variabel prediktor, yang meningkat menjadi 23 variabel setelah *encoding*. Pemecahan variabel kategorik menjadi beberapa fitur biner ini memperjelas representativitas masing-masing kategori, sehingga memudahkan model *machine learning* dalam mempelajari pola dari setiap kategori secara independen.

5) *Standarisasi Data*: Standarisasi Data: Setelah semua variabel kategorik melalui tahap *one-hot encoding*, seluruh variabel yang tersedia dalam dataset distandarisasi menggunakan StandardScaler. Tabel 4 menampilkan contoh hasil standarisasi untuk lima variabel pertama, di mana nilai mean dan standar deviasi diperoleh setelah transformasi.

TABEL IV
HASIL STANDARISASI UNTUK LIMA VARIABEL PERTAMA

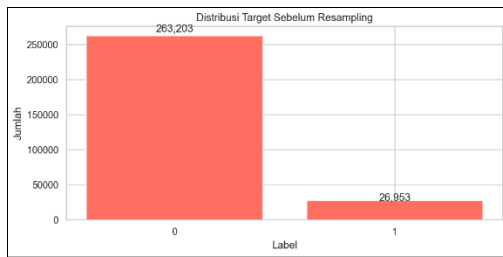
Variabel	Mean	Standar Deviasi
JenisKelamin	0,00	1,00
TekananDarahTinggi	0,00	1,00
KolesterolTinggi	0,00	1,00
Diabetes	0,00	1,00
KesulitanBerjalan	0,00	1,00

Tabel 4 menampilkan hasil standarisasi lima variabel pertama dalam dataset, di mana setiap variabel yang ditunjukkan memiliki nilai mean 0,00 dengan standar deviasi 1,00. Nilai ini menunjukkan bahwa seluruh fitur telah berhasil ditransformasi ke skala yang seragam. Standarisasi ini krusial karena memiliki peran untuk menghilangkan bias akibat perbedaan rentang nilai antar-variabel, sehingga tidak ada fitur yang dianggap lebih mendominasi dalam proses pembelajaran model. Dengan skala yang konsisten, model *machine learning* dapat mempelajari pola setiap fitur secara adil, mempercepat konvergensi, dan menghasilkan prediksi yang lebih stabil serta objektif.

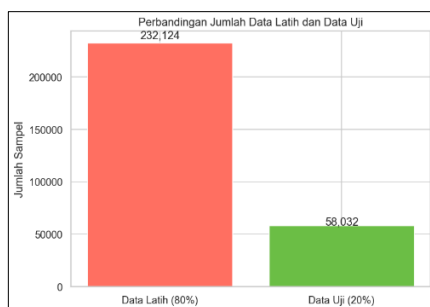
C. Hasil Pembagian Data

Setelah data melalui seluruh tahapan prapemrosesan, diperoleh dataset akhir yang siap digunakan pada tahap pemodelan. Meskipun telah melalui proses prapemrosesan, distribusi kelas pada dataset masih menunjukkan proporsi yang belum seimbang, sebagaimana yang ditampilkan pada

Gambar 11. Selanjutnya, dataset tersebut dibagi menjadi data latih dan data uji, dengan distribusi pembagian data latih dan data uji ditampilkan pada Gambar 12.



Gambar 11. Distribusi kelas target sebelum penyeimbangan data

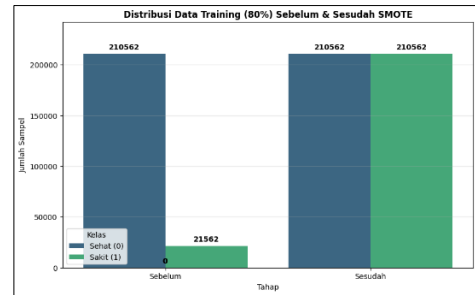


Gambar 12. Distribusi data latih dan data uji

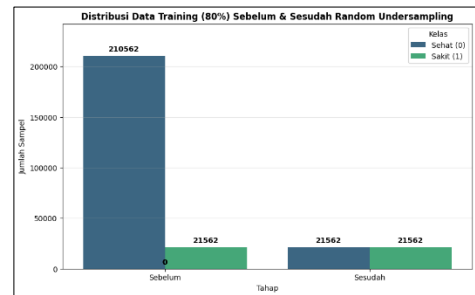
Berdasarkan Gambar 11, distribusi kelas pada dataset menunjukkan kondisi yang belum seimbang, dengan kelas 0 sebanyak 263.203 entri (90,7%) dan kelas 1 sebanyak 26.953 entri (9,3%), yang mencerminkan dominasi kelas 0 yang cukup besar. Ketimpangan ini berpotensi menimbulkan bias prediksi yang menyebabkan model cenderung mengklasifikasikan sampel ke kelas mayoritas, sehingga diperlukan proses penyeimbangan data sebelum tahap pemodelan dilakukan pada tugas klasifikasi risiko penyakit jantung. Selanjutnya, Gambar 12 menunjukkan pembagian dataset menjadi dua bagian, yaitu 232.124 sampel (80%) sebagai data latih dan 56.032 sampel (20%) sebagai data uji.

D. Hasil Penanganan Ketidakseimbangan Kelas

Sebagai ilustrasi upaya penanganan ketidakseimbangan kelas pada data latih, Gambar 13 dan Gambar 14 menunjukkan perubahan proporsi kelas sebelum dan setelah penerapan teknik *Synthetic Minority Oversampling Technique* (SMOTE) dan *Random Undersampling* (RUS). Sementara itu, pada skenario yang menerapkan teknik *class weighting*, proporsi kelas pada data latih tetap tidak berubah, karena pendekatan ini hanya memberikan pembobotan yang berbeda antara kelas mayoritas dan kelas minoritas tanpa mengubah distribusi data.



Gambar 13. Distribusi kelas data latih: sebelum dan setelah SMOTE



Gambar 14. Distribusi kelas data latih: sebelum dan setelah RUS

Teknik penanganan ketidakseimbangan kelas hanya diterapkan pada data latih untuk mencegah terjadinya *data leakage*, sehingga data uji tetap dipertahankan dengan distribusi kelas aslinya yang tidak seimbang (*imbalanced*). Pada Gambar 13, penerapan SMOTE menambahkan jumlah sampel minoritas yang semula berjumlah 21.562 sampel menjadi setara dengan kelas mayoritas, sehingga kedua kelas masing-masing berjumlah 210.562 sampel. Sementara itu, pada skenario *Random Undersampling* (RUS) pada Gambar 14, jumlah sampel kelas mayoritas yang semula 210.562 sampel dikurangi hingga setara dengan kelas minoritas, sehingga kedua kelas pada data latih masing-masing berjumlah 21.562 sampel.

E. Evaluasi Model

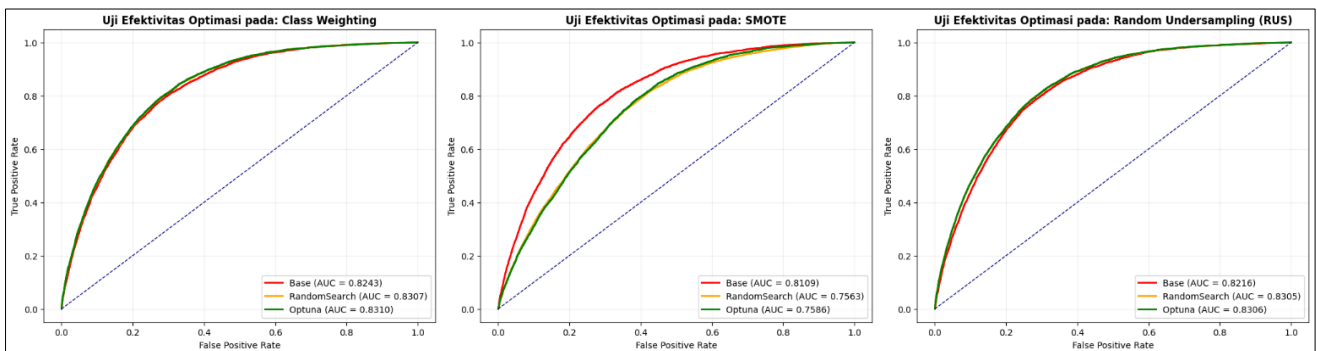
Proses evaluasi dilakukan secara menyeluruh pada semua skenario pemodelan untuk menilai kinerja model dalam tugas klasifikasi risiko penyakit jantung berbasis dataset BRFSS 2023. Evaluasi diterapkan menggunakan berbagai metrik evaluasi yang meliputi akurasi, presisi, *recall*, F1-score, serta kurva ROC-AUC. Melalui berbagai metrik yang disebutkan, evaluasi memberikan gambaran kinerja yang komprehensif dari masing-masing skenario pemodelan, sekaligus mengidentifikasi kekuatan dan kelemahannya serta menentukan model dengan kemampuan generalisasi terbaik terhadap data uji. Selain itu, analisis *feature importance* dilakukan pada model dengan kinerja terbaik guna memperoleh pemahaman lebih lanjut mengenai kontribusi masing-masing fitur dalam proses prediksi risiko penyakit jantung, sehingga faktor-faktor kunci yang memengaruhi prediksi dapat diidentifikasi.

1) *Evaluasi Kinerja Seluruh Skenario Pemodelan:* Gambaran mengenai evaluasi kinerja pada seluruh skenario

pemodelan disajikan pada Tabel 5, dengan grafik ROC-AUC untuk masing-masing skenario ditampilkan pada Gambar 15.

TABEL V
PERBANDINGAN NILAI METRIK EVALUASI PADA SELURUH SKENARIO PEMODELAN

Model	Akurasi	Presisi	Recall	F1-Score	AUC	Runtime	Best Trial
BaseWeight	72,39%	59,59%	75,11%	58,53%	82,43%	14,03 (s)	-
OptunaWeight	90,79%	73,25%	52,09%	51,78%	83,10%	4236,63 (s)	Trial ke-32
RandomWeight	71,41%	59,58%	75,62%	58,07%	83,07%	4152,97 (s)	50 Iterations
BaseSMOTE	74,56%	59,54%	73,60%	59,40%	81,09%	23,60 (s)	-
OptunaSMOTE	79,50%	57,54%	64,11%	58,54%	75,86%	17305,08 (s)	Trial ke-44
RandomSMOTE	79,02%	57,65%	64,78%	58,60%	75,63%	6777,29 (s)	50 Iterations
BaseRUS	71,01%	59,34%	75,07%	57,67%	82,16%	9,02 (s)	-
OptunaRUS	71,57%	59,63%	75,69%	58,19%	83,06%	1016,99 (s)	Trial ke-37
RandomRUS	71,73%	59,67%	75,73%	58,30%	83,05%	852,76 (s)	50 Iterations



Gambar 15. Grafik ROC-AUC pada seluruh skenario pemodelan

Evaluasi kinerja model dilakukan berdasarkan dataset BRFS 2023 yang telah melalui tahap prapemrosesan, sehingga menghasilkan sekitar 290 ribu observasi yang siap digunakan. Dataset tersebut selanjutnya dibagi menjadi data latih dan data uji dengan proporsi 80% dan 20%. Seluruh strategi penanganan ketidakseimbangan kelas diterapkan secara eksklusif pada data latih, sementara data uji dipertahankan dalam kondisi distribusi asli untuk menguji kemampuan generalisasi model. Seluruh skenario pemodelan dibangun di atas algoritma dasar XGBoost yang dikombinasikan dengan *class weighting*, SMOTE, dan *Random Undersampling* (RUS), baik tanpa optimasi maupun dengan optimasi hiperparameter. Evaluasi difokuskan terutama pada nilai AUC dan *recall*, sebagaimana yang ditampilkan pada Tabel 5 dan kurva ROC-AUC pada Gambar 15. Nilai AUC dipilih karena merepresentasikan kemampuan model dalam membedakan kelas secara global dan independen terhadap ambang klasifikasi, sedangkan *recall* mencerminkan kemampuan model dalam mengidentifikasi responden yang berisiko terhadap penyakit jantung. Sementara itu, nilai F1-score pada seluruh skenario cenderung lebih rendah dikarenakan distribusi kelas pada data uji tetap dipertahankan tidak seimbang sebagai bagian dari pengujian kemampuan generalisasi model, sehingga metrik ini menjadi salah satu yang paling terdampak oleh ketidakseimbangan kelas.

Berdasarkan hasil pada Tabel 5 dan pola kurva ROC-AUC pada Gambar 15, penerapan optimasi hiperparameter pada

model XGBoost menunjukkan peningkatan performa yang relatif terbatas pada sebagian besar skenario. Pada skenario yang menggunakan pendekatan *class weighting*, nilai AUC meningkat dari 82,43% pada BaseWeight menjadi 83,10% pada OptunaWeight, sementara pada RandomWeight nilai AUC tercatat sebesar 83,07%. Perbedaan yang relatif kecil ini menunjukkan bahwa kemampuan diskriminatif model telah terbentuk dengan baik pada konfigurasi dasar. Pola kurva ROC-AUC yang saling berhimpit antar skenario optimasi mengindikasikan bahwa pada dataset BRFS 2023 yang berskala besar dan beragam, optimasi hiperparameter lebih berperan dalam menjaga konsistensi performa dibandingkan menghasilkan peningkatan metrik yang signifikan.

Di sisi lain, pendekatan *Random Undersampling* menunjukkan karakteristik performa yang stabil pada seluruh skenario yang diuji. Berdasarkan pemaparan pada Tabel 5, BaseRUS menghasilkan nilai AUC sebesar 82,16% dengan *recall* 75,07%. Sementara itu, OptunaRUS dan RandomRUS menunjukkan peningkatan kinerja, yang ditunjukkan oleh nilai AUC masing-masing sebesar 83,06% dan 83,05%, dengan *recall* berada pada rentang 75,69–75,73%. Konsistensi performa ini mengindikasikan bahwa meskipun jumlah data latih berkurang secara signifikan setelah proses *undersampling*, model tetap mampu mempertahankan kemampuan diskriminatif dan sensitivitasnya. Temuan ini juga menunjukkan bahwa dataset BRFS 2023 memiliki tingkat heterogenitas yang memadai, namun tetap memiliki pola informasi yang bersifat redundan, sehingga pengurangan

sampel terbukti tidak menyebabkan hilangnya informasi penting yang unik.

Sebaliknya, pada skenario yang menggunakan pendekatan SMOTE, peningkatan jumlah data latih akibat *oversampling* berdampak signifikan terhadap waktu pelatihan model, khususnya pada skenario OptunaSMOTE yang memerlukan waktu lebih dari 17.000 detik. Meski begitu, berdasarkan pemaparan pada Tabel 5, peningkatan waktu dan biaya komputasi tersebut tidak diikuti oleh peningkatan performa yang sepadan. BaseSMOTE justru menghasilkan nilai AUC sebesar 81,09% dengan *recall* 73,60%, yang lebih baik dibandingkan OptunaSMOTE dengan AUC 75,86% dan *recall* 64,11%. Hal tersebut mengindikasikan bahwa pada dataset yang sudah besar dan beragam, penambahan data sintesis justru berpotensi memperbanyak *noise* daripada memperkaya representasi pola, sehingga berdampak negatif terhadap kemampuan generalisasi model. Dengan demikian, optimasi hiperparameter pada skenario SMOTE menjadi kurang efisien dan tidak memberikan keuntungan yang sebanding dengan sumber daya komputasi yang dikeluarkan.

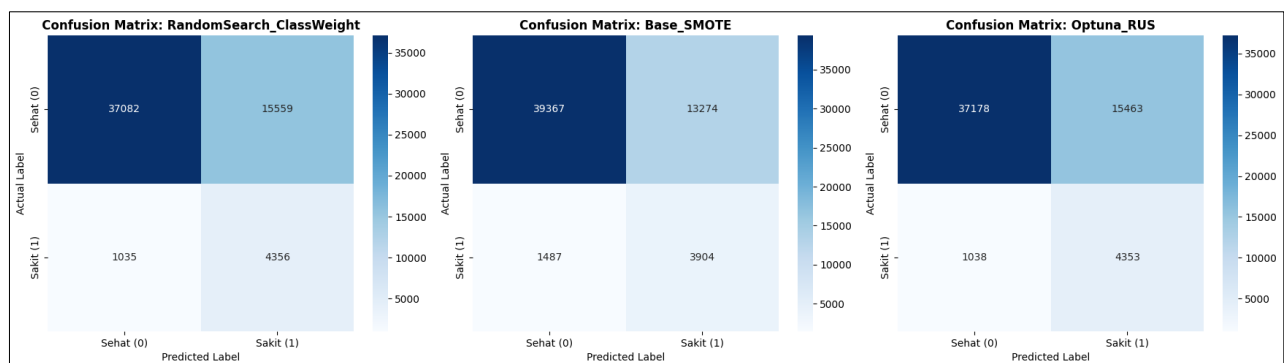
Secara keseluruhan, hasil evaluasi pada Tabel 5 dan Gambar 15 menunjukkan bahwa karakteristik dataset BRFSS 2023 yang berskala besar dan beragam lebih selaras dengan pendekatan penanganan ketidakseimbangan kelas yang bersifat selektif dibandingkan generatif. Dari masing-masing

pendekatan penanganan ketidakseimbangan kelas yang diterapkan, skenario RandomWeight pada pendekatan *class weighting*, BaseSMOTE pada pendekatan *oversampling*, dan OptunaRUS pada pendekatan *undersampling* muncul sebagai model dengan performa yang paling stabil dan *robust*. Ketiga skenario tersebut merepresentasikan konfigurasi pemodelan terbaik dari setiap pendekatan, karena mampu mempertahankan keseimbangan performa yang konsisten, khususnya pada metrik AUC dan *recall*, tanpa menunjukkan ketergantungan pada peningkatan kompleksitas komputasi yang berlebihan. Dengan karakteristik tersebut, ketiga model dianggap layak untuk dijadikan representasi utama pada analisis evaluasi lanjutan, sehingga mendukung proses pengkajian mendalam mengenai stabilitas generalisasi serta implikasi kesalahan klasifikasi pada konteks klasifikasi risiko penyakit jantung.

2) *Evaluasi Tiga Skenario Pemodelan Terbaik*: Untuk memperdalam analisis evaluasi, kajian difokuskan pada tiga skenario pemodelan yang memiliki performa paling stabil guna menilai stabilitas generalisasi serta konsekuensi kesalahan klasifikasi. Perbandingan nilai metrik evaluasi antara data latih dan data uji disajikan pada Tabel 6. Sementara itu, *confusion matrix* dari masing-masing model ditampilkan pada Gambar 16.

TABEL VI
PERBANDINGAN NILAI METRIK EVALUASI PADA SELURUH SKENARIO PEMODELAN

Model	Recall (Train)	Recall (Test)	AUC (5-fold CV)	AUC (Train)	AUC (Test)
RandomWeight	75,59%	75,62%	82,90%	83,20%	83,07%
BaseSMOTE	79,37%	73,60%	87,47%	87,91%	81,09%
OptunaRUS	75,63%	75,69%	82,75%	83,16%	83,06%



Gambar 16. Hasil *confusion matrix* pada tiga skenario pemodelan terbaik

Berdasarkan perbandingan metrik evaluasi yang disajikan pada Tabel 6, ketiga skenario pemodelan terpilih menunjukkan tingkat stabilitas generalisasi yang konsisten antara data latih dan data uji. Pada RandomWeight, nilai *recall* pada data latih sebesar 75,59% relatif seimbang dengan nilai *recall* pada data uji sebesar 75,62%, dengan nilai AUC yang juga konsisten, yakni 83,20% pada data latih dan 83,07% pada data uji. Pola serupa ditunjukkan oleh OptunaRUS, dengan nilai *recall* sebesar 75,63% pada data latih dan 75,69% pada data uji, serta nilai AUC masing-

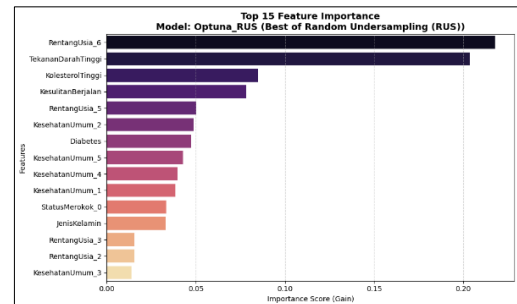
masing sebesar 83,16% dan 83,06%. Konsistensi ini mengindikasikan tidak adanya kecenderungan model mengalami *overfitting*, sekaligus menunjukkan bahwa performa model tetap terjaga dari indikasi *underfitting*. Berbeda dengan kedua skenario tersebut, BaseSMOTE memperlihatkan selisih yang lebih besar, khususnya pada nilai *recall* yang menurun dari 79,37% pada data latih menjadi 73,60% pada data uji, serta penurunan nilai AUC dari 87,91% menjadi 81,09%. Meskipun demikian, performa BaseSMOTE pada data uji masih berada dalam batas yang dapat diterima,

sehingga tetap relevan sebagai representasi skenario pemodelan dengan pendekatan *oversampling*. Temuan ini menunjukkan bahwa mempertahankan distribusi kelas tidak seimbang pada data uji dan penerapan validasi silang lima lipatan juga turut menjaga stabilitas generalisasi ketiga model.

Analisis lanjutan terhadap implikasi kesalahan klasifikasi ditunjukkan melalui *confusion matrix* pada Gambar 16. Dalam prediksi risiko penyakit jantung, *False Negative* merupakan kesalahan paling krusial karena individu yang sebenarnya berisiko diprediksi tidak berisiko, sehingga berpotensi kehilangan kesempatan deteksi dan penanganan dini. Pada skenario RandomWeight dan OptunaRUS, jumlah *False Negative* tercatat masing-masing 1.035 dan 1.038 dari total 58.032 data uji, lebih rendah dibanding BaseSMOTE yang menghasilkan 1.487 kasus. Pola ini sejalan dengan nilai *recall* pada data uji, yang menunjukkan RandomWeight dan OptunaRUS lebih sensitif dalam mengidentifikasi responden berisiko. Sebaliknya, pada skenario *False Positive*, meskipun dapat memicu pemeriksaan tambahan dan meningkatkan kecemasan pasien, umumnya masih dapat ditoleransi. RandomWeight dan OptunaRUS menghasilkan jumlah *False Positive* yang sedikit lebih tinggi dibanding BaseSMOTE, namun *trade-off* ini mencerminkan kecenderungan model untuk memprioritaskan penekanan *False Negative*, yang secara klinis lebih dianggap lebih penting.

Ringkasan dari analisis stabilitas generalisasi dan distribusi kesalahan klasifikasi menunjukkan bahwa ketiga skenario pemodelan memiliki karakteristik yang saling melengkapi dan layak dijadikan sebagai dasar analisis evaluasi lanjutan. RandomWeight dan OptunaRUS terbukti unggul dalam menjaga keseimbangan antara performa statistik dan sensitivitas klinis, tercermin dari stabilitas AUC dan *recall* serta jumlah *False Negative* yang relatif lebih rendah. Meskipun performanya menurun pada data uji, BaseSMOTE tetap relevan karena mampu memberikan perspektif komparatif terhadap dampak *oversampling* sintesis pada dataset besar dan heterogen. Pemilihan ketiga model ini didasarkan pada keseimbangan kemampuan generalisasi, efisiensi komputasi, dan dampak kesalahan klasifikasi, sehingga ketiganya cocok digunakan sebagai acuan analisis lanjutan pada pembahasan mengenai *overfitting*, *underfitting*, serta implikasi praktis dalam prediksi risiko penyakit jantung.

3) *Analisis Feature Importance Model Terbaik*: Analisis *feature importance* difokuskan pada skenario OptunaRUS sebagai model yang menunjukkan performa paling stabil dengan kemampuan generalisasi terbaik di antara tiga skenario unggulan. Kontribusi fitur divisualisasikan pada Gambar 17 untuk menyoroti variabel yang paling memengaruhi prediksi risiko penyakit jantung.



Gambar 17. *Feature importance* pada skenario pemodelan OptunaRUS

Berdasarkan visualisasi *importance score* pada Gambar 17, RentangUsia_6 dan TekananDarahTinggi muncul sebagai prediktor paling dominan dalam klasifikasi risiko penyakit jantung. Variabel lain seperti KolesterolTinggi, Diabetes, KesulitanBerjalan, KesehatanUmum, dan StatusMerokok juga menempati peringkat signifikan, menunjukkan hubungan kuat antara kondisi klinis, mobilitas fisik, gaya hidup, dan persepsi kesehatan dengan risiko kardiovaskular. Secara keseluruhan, OptunaRUS tidak hanya unggul secara performa statistik, tetapi juga mampu menangkap pola kombinasi faktor demografi, komorbiditas, dan kualitas hidup fisik yang relevan secara klinis.

IV. KESIMPULAN

Penelitian ini berhasil mengoptimalkan model klasifikasi risiko penyakit jantung berbasis XGBoost melalui penerapan optimasi hiperparameter pada dataset BRFS 2023 yang berskala besar dan memiliki distribusi kelas tidak seimbang. Hasil eksperimen menunjukkan bahwa penggunaan Optuna dan Random Search secara konsisten meningkatkan stabilitas performa model, terutama dalam menjaga keseimbangan antara kemampuan diskriminatif dan sensitivitas deteksi. Konfigurasi OptunaRUS teridentifikasi sebagai model dengan performa paling stabil, dengan capaian AUC sebesar 83,06% dan *recall* sebesar 75,69% pada data uji. Konsistensi metrik evaluasi antara data latih dan data uji mengindikasikan kemampuan generalisasi model yang baik tanpa adanya indikasi *overfitting*. Hasil evaluasi pada distribusi kelas asli menegaskan bahwa performa model relevan dalam merepresentasikan kondisi data kesehatan dunia nyata.

Dari sisi metodologis, temuan penelitian ini menegaskan bahwa pada dataset dengan tingkat heterogenitas tinggi, pendekatan penanganan ketidakseimbangan kelas secara selektif melalui *Random Undersampling (RUS)* lebih efektif dibandingkan metode generatif seperti SMOTE. Reduksi sampel mayoritas mampu menekan pengaruh *noise* sekaligus meningkatkan efisiensi komputasi, sebagaimana tercermin dari stabilitas metrik evaluasi pada data uji. *Analisis feature importance* pada model terbaik menunjukkan bahwa usia dan hipertensi merupakan fitur yang paling dominan dalam klasifikasi risiko penyakit jantung. Secara keseluruhan, penelitian ini menawarkan kerangka kerja prediktif yang stabil dan efisien untuk klasifikasi risiko penyakit jantung,

dengan potensi penerapan pada analisis data kesehatan populasi berskala besar.

DAFTAR PUSTAKA

- [1] H. Hidayat, A. Sunyoto, and H. Al Fatta, "Klasifikasi Penyakit Jantung Menggunakan Random Forest Classifier," *J. SISKOM-KB (Sistem Komput. dan Kecerdasan Buatan)*, vol. 7, no. 1, pp. 31–40, 2023, doi: 10.47970/siskom-kb.v7i1.464.
- [2] H. Sawitri, N. Maulina, T. Y. Lutfi, and N. Rahmi, "Tingkat Risiko Penyakit Jantung dan Pembuluh Darah pada Dosen dan Karyawan," *J. Ilm. Mns. Dan Kesehat.*, vol. 6, no. 1, pp. 37–43, 2023, doi: 10.31850/makes.v6i1.1914.
- [3] Kemenkes, *Profil Kesehatan Indonesia 2023*. 2024. [Online]. Available: <https://kemkes.go.id/id/profil-kesehatan-indonesia-2023>
- [4] Kemenkes, *Survei Kesehatan Indonesia (SKI) 2023*. 2023. [Online]. Available: <https://www.badankebijakan.kemkes.go.id/ski-2023-dalam-angka/>
- [5] D. G. Pradana, M. L. Alghifari, M. F. Juna, and S. D. Palaguna, "Klasifikasi Penyakit Jantung Menggunakan Metode Artificial Neural Network," *Indones. J. Data Sci.*, vol. 3, no. 2, pp. 55–60, 2022, doi: 10.56705/ijodas.v3i2.35.
- [6] N. Nuraeni, "Klasifikasi Data Mining Untuk Prediksi Penyakit Kardiovaskular," *J. TEKINKOM (Teknik Inf. dan Komputer)*, vol. 7, no. 1, pp. 161–170, 2024, doi: 10.37600/tekinkom.v7i1.1276.
- [7] S. Munawaroh, U. A. Rosyidah, and R. Yanuarti, "Klasifikasi Tingkat Kecemasan Atlet Sebelum Bertanding Menggunakan Algoritma K-Nearest Neighbor (KNN) Berbasis Website," *BIOS J. Teknol. Inf. dan Rekayasa Komput.*, vol. 5, no. 2, pp. 87–94, 2024, doi: 10.37148/bios.v5i2.120.
- [8] A. U. Dullah, A. Y. Darmawan, D. A. A. Pertiwi, and J. Unjung, "Extreme Gradient Boosting Model with SMOTE for Heart Disease Classification," *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 10, no. 1, pp. 48–62, 2025, doi: 10.14421/jiska.2025.10.1.48-62.
- [9] G. Velarde *et al.*, "Tree boosting methods for balanced and imbalanced classification and their robustness over time in risk assessment," *Intell. Syst. with Appl.*, vol. 22, 2024, doi: 10.1016/j.iswa.2024.200354.
- [10] A. A. Yaqin, M. A. Barata, and N. Mahmudah, "Implementation of the Random Forest Algorithm with Optuna Optimization in Lung Cancer Classification," *Sist. J. Sist. Inf.*, vol. 14, no. 2, pp. 561–569, 2025, doi: 10.32520/stmsi.v14i2.4877.
- [11] B. F. Sitanggang and P. Sitompul, "Deteksi Awal Kelangsunan Hidup Pasien Gagal Jantung Menggunakan Machine Learning Metode Random Forest," *Innov. J. Soc. Sci. Res.*, vol. 4, no. 2, pp. 3347–3357, 2024, doi: 10.31004/innovative.v4i2.8189.
- [12] Firmansyah and A. Yulianto, "Prediksi Penyakit Jantung Menggunakan Algoritma Random Forest," *J. Minfo Polgan (Jurnal Penelit. Manaj. Inform.)*, vol. 12, no. 2, pp. 2239–2246, 2023, doi: 10.33395/jmp.v12i2.13214.
- [13] G. Almuzadid and E. R. Subhiyaktio, "Stroke Risk Classification Using the Ensemble Learning Method of XGBoost and Random Forest," *J. Appl. Informatics Comput.*, vol. 9, no. 3, pp. 828–837, 2025, doi: 10.30871/jaic.v9i3.9528.
- [14] D. Ariyanto, D. C. R. Novitasari, and A. Hamid, "Heart Disease Classification Using Extreme Learning Machine (ELM) Method With Outlier Handling One-Class Support Vector Machine (OCSVM)," *J. Appl. Informatics Comput.*, vol. 9, no. 5, pp. 2143–2153, 2025, doi: 10.30871/jaic.v9i5.9815.
- [15] V. R. Maulani, M. A. Barata, and P. E. Yuwita, "Improving House Price Clustering Results with K-means through the Implementation of One-hot Encoding Pre-processing Technique," *J. Appl. Informatics Comput.*, vol. 9, no. 3, pp. 741–748, 2025, doi: 10.30871/jaic.v9i3.9481.
- [16] A. Digdoyo, A. S. B. Karno, W. Hastomo, E. Sestri, and R. Fitriansyah, "Prediksi Cacat Lempeng Baja Menggunakan Algoritma Bagging: Pendekatan Pembelajaran Mesin untuk Peningkatan Kualitas Produksi," *J. Ilm. KOMPUTASI*, vol. 24, pp. 87–94, 2025, doi: 10.32409/jikstik.24.1.3654.
- [17] E. Setiawan, B. Sartono, and K. A. Notodiputro, "SMOTE and Weighted Random Forest for Classification of Areas Based on Health Problems in Java," *J. Appl. Informatics Comput.*, vol. 9, no. 4, pp. 1587–1592, 2025, doi: 10.30871/jaic.v9i4.9933.
- [18] M. F. Kurniawan and D. A. Megawaty, "Comparison of Logistic Regression, Random Forest, Support Vector Machine (SVM) and K-Nearest Neighbor (KNN) Algorithms in Diabetes Prediction," *J. Appl. Informatics Comput.*, vol. 9, no. 5, pp. 2154–2162, 2025, doi: <https://doi.org/10.30871/jaic.v9i5.9815>.
- [19] G. A. P. Febriyanti and A. Baita, "Comparison of Support Vector Machine and Decision Tree Algorithm Performance with Undersampling Approach in Predicting Heart Disease Based on Lifestyle," *J. Appl. Informatics Comput.*, vol. 9, no. 2, pp. 318–327, 2025, doi: 10.30871/jaic.v9i2.8941.
- [20] D. Kurnia, M. I. Mazdadi, D. Kartini, R. A. Nugroho, and F. Abadi, "Seleksi Fitur dengan Particle Swarm Optimization pada Klasifikasi Penyakit Parkinson Menggunakan XGBoost," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 5, pp. 1083–1094, 2023, doi: 10.25126/jtiik.2023107252.
- [21] Nurbaeti, N. Sulistiyarningsih, and R. Rismayanti, "Comparison of Random Forest, Decision Tree, and XGBoost Models in Predicting Student Academic Success," *J. Artif. Intell. Softw. Eng.*, vol. 5, no. 3, pp. 920–930, 2025, doi: 10.30811/jaise.v5i3.7138.
- [22] P. Zhang, Y. Jia, and Y. Shang, "Research and application of XGBoost in imbalanced data," *Int. J. Distrib. Sens. Networks*, vol. 18, no. 263, 2022, doi: 10.1177/15501329221106935.
- [23] R. Andika and Kusriani, "Optimasi Hyperparameter Model LSTM dan Variannya untuk Peramalan Pembelian Bahan Baku Karet Alam," *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 10, no. 3, pp. 2627–2639, 2025, doi: 10.29100/jupi.v10i3.7567.
- [24] B. Siswoyo *et al.*, "Optimization of Multi-Layer Perceptron in Ensemble Using Random Search for Bankruptcy Prediction," *J. Comput. Sci.*, vol. 19, no. 2, pp. 251–260, 2023, doi: 10.3844/jcssp.2023.251.260.
- [25] F. D. Marleny, M. Fitriansyah, Sa'adah, W. A. N. Saputri, R. Ansari, and Mambang, "Segmentasi Citra Keretakan Dinding Beton Menggunakan Teknik Perbandingan Evaluasi Metrik," *Temat. J. Teknol. Inf. Komun.*, vol. 10, no. 2, pp. 28–33, 2023, doi: 10.38204/tematik.v10i1.1261.
- [26] L. Hakim, A. Sobri, L. Sunardi, and D. Nurdiansyah, "Prediksi Penyakit Jantung Berbasis Mesin Learning Dengan Menggunakan Metode K-NN," *J. Digit. Teknol. Inf.*, vol. 07, no. 02, pp. 14–20, 2024, doi: 10.32502/digital.v7i2.9429.
- [27] R. Harahap, M. Irpan, M. A. Dinata, L. Efrizoni, and Rahmaddeni, "Perbandingan Algoritma Random Forest dan XGBoost untuk Klasifikasi Penyakit Paru-Paru Berdasarkan Data Demografi Pasien," *J. Ilm. Betrik*, vol. 15, no. 02, pp. 130–141, 2024, doi: 10.36050/3v3xwn06.
- [28] Kristiawan and A. Widjaja, "Perbandingan Algoritma Machine Learning dalam Menilai Sebuah Lokasi Toko Ritel," *J. Tek. Inform. dan Sist. Inf.*, vol. 7, no. 1, pp. 35–46, 2021, doi: 10.28932/jutisi.v7i1.3182.
- [29] Amrin, Rudianto, and Sismadi, "Data Mining with Logistic Regression and Support Vector Machine for Hepatitis Disease Diagnosis," *JITE (Journal Informatics Telecommun. Eng.)*, vol. 8, no. 2, pp. 248–256, 2025, doi: 10.31289/jite.v8i2.13218.