

Benchmarking Oversampling Strategies to Enhance the Performance of Machine Learning Algorithms in Hypertension Classification

Aenur Hakim Maulia ^{1*}, Abu Salam ^{2**}

* Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
111202214639@mhs.dinus.ac.id¹, abu.salam@dsn.dinus.ac.id²

Article Info

Article history:

Received 2025-12-01

Revised 2025-12-31

Accepted 2026-01-13

Keyword:

*Data Oversampling,
Hypertension classification,
Machine Learning,
Random Oversampling,
Linear Discriminant Analysis*

ABSTRACT

This study benchmarks the effectiveness of three oversampling techniques, namely SMOTE, Random Oversampling (ROS), and ADASYN, in enhancing machine learning performance for multiclass hypertension classification. Using key physiological features and four optimized algorithms Logistic Regression, Support Vector Machine, Linear Discriminant Analysis, and Artificial Neural Networks, model performance was assessed using accuracy, F1-macro, and ROC AUC metrics. The experimental results indicate that the combination of SMOTE and Linear Discriminant Analysis (LDA) yields the highest overall performance, achieving an accuracy of 0.9773 and an F1-macro score of 0.9848. Logistic Regression demonstrates optimal results when paired with ROS, also reaching an accuracy of 0.9773. Artificial Neural Networks show the most substantial performance improvement under ADASYN, particularly reflected in higher F1-macro values. Although Support Vector Machine is less sensitive to oversampling interventions, it achieves a strong ROC AUC score of 0.9776 when trained using SMOTE. Overall, the findings confirm that oversampling techniques significantly improve classification performance in multilevel hypertension prediction, with SMOTE combined with LDA emerging as the most effective configuration.



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

I. PENDAHULUAN

Hipertensi merupakan salah satu penyakit tidak menular dengan prevalensi tinggi di dunia. Berdasarkan data yang dikeluarkan oleh Organisasi Kesehatan Dunia (WHO), pada tahun 2024 terdapat sekitar 1,4 miliar orang dewasa berusia 30–79 tahun menderita hipertensi, atau mencakup 33% dari populasi dalam rentang usia tersebut [1]. Di Indonesia, kasus hipertensi juga semakin banyak ditemukan pada kelompok usia produktif 26–35 tahun dengan prevalensi 4,1%, yang dipengaruhi oleh berbagai faktor seperti stress kerja, diabetes melitus, gangguan kesehatan mental, serta konsumsi makanan instan yang berlebihan [2]. Menurut pedoman European Society of Hypertension (ESH), tekanan darah manusia terdiri dari dua komponen utama, antara lain tekanan darah sistolik yaitu tekanan dalam pembuluh darah saat jantung berdetak, dan tekanan diastolik yang terjadi di antara ketukan denyut jantung, hipertensi didiagnosis jika tekanan darah sistolik ≥ 140 mmHg atau diastolik ≥ 90 mmHg yang terjadi secara

menetap berdasarkan beberapa kali pengukuran klinis. Pedoman ESH tersebut juga menegaskan bahwa hipertensi dapat meningkatkan risiko komplikasi seperti penyakit jantung, stroke, dan gagal jantung [3]. Oleh karena itu, deteksi dini dan klasifikasi risiko hipertensi sangat penting dilakukan untuk mencegah komplikasi lebih lanjut. Dalam masalah ini, teknologi informasi dan machine learning berperan penting dalam mempercepat proses diagnosis dan klasifikasi hipertensi [4], [5], [6].

Efektivitas model machine learning sangat dipengaruhi oleh kualitas data yang digunakan. Salah satu tantangan umum yang sering dijumpai adalah ketidakseimbangan kelas, penelitian sebelumnya oleh Novianti et al. [7] menemukan adanya ketidakseimbangan kelas pada dataset PPG-BP yang juga digunakan dalam penelitian ini, temuan tersebut mengindikasikan bahwa proses penyeimbangan kelas sebelum pelatihan model machine learning perlu dilakukan agar pelatihan menjadi optimal. Penelitian tersebut juga telah mengeksplorasi penggunaan teknik penyeimbangan kelas

untuk meningkatkan performa algoritma Random Forest dalam klasifikasi hipertensi, akan tetapi masih terbatas pada dua teknik penyeimbangan kelas dan satu algoritma machine learning saja, serta belum melibatkan proses optimasi hyperparameter secara sistematis. Melihat kompleksitas dan variabilitas data klinis yang tinggi, penelitian ini bertujuan untuk mengisi celah tersebut dengan mengevaluasi empat model machine learning yang dikombinasikan secara terstruktur dengan tiga teknik penyeimbangan kelas yang berbeda. Salah satu pendekatan umum yang digunakan untuk mengatasi ketidakseimbangan kelas adalah teknik Oversampling, terdapat beberapa pendekatan oversampling, diantaranya seperti Synthetic Minority Oversampling Technique (SMOTE), Random Oversampling (ROS), dan Adaptive Synthetic Sampling (ADASYN) [8].

Selain oversampling, pemilihan algoritma machine learning juga berpengaruh terhadap performa model. Berbagai algoritma seperti Logistic Regression (LR), Support Vector Machine (SVM), Linear Discriminant Analysis (LDA), dan Artificial Neural Network (ANN) sering digunakan sebagai model pembanding dalam evaluasi kinerja oversampling [9], [10], [11], [12]. Sementara itu, dalam konteks performa model, Penelitian Handayani [13] menunjukkan bahwa Logistic Regression memiliki performa prediksi yang kompetitif dalam klasifikasi penyakit jantung. Nasien et al. [14] melaporkan bahwa metode ini mampu menghasilkan akurasi yang stabil pada klasifikasi diabetes, Jibril et al. menggunakan algoritma Support Vector Machine (SVM) menyatakan bahwa metode ini mampu meningkatkan akurasi prediksi melalui pemilihan hyperplane yang optimal [15]. Linear Discriminant Analysis merupakan metode yang digunakan untuk menemukan kombinasi fitur linier yang memaksimalkan separabilitas kelas. Vamshi Kumar et al. [16] menunjukkan bahwa LDA efektif meningkatkan performa klasifikasi serangan jantung melalui transformasi linier fitur. Sementara Samarina et al. menerapkan LDA untuk membangun model prediksi hipertensi pada anak mendapatkan akurasi 90% [17]. Untuk Artificial Neural Network (ANN), Ebrahimi dan Farshid [18] menunjukkan bahwa ANN mampu mencapai akurasi tinggi dalam diagnosis obesitas dan hipertensi. Untuk memperoleh performa optimal, proses tuning hyperparameter diperlukan. Muzayannah et al. [19] membuktikan bahwa GridSearchCV dapat meningkatkan kinerja model secara signifikan dibanding metode optimasi lainnya.

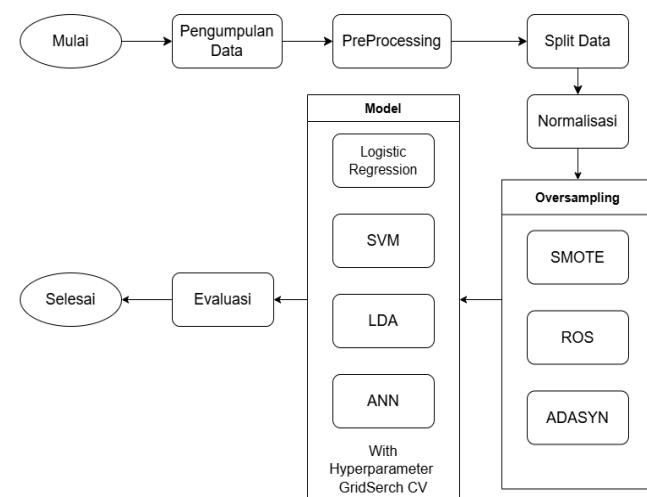
Untuk mengukur performa model, penerapan evaluasi juga diperlukan guna mengidentifikasi kombinasi teknik oversampling dan model machine learning yang paling optimal dalam meningkatkan kinerja model klasifikasi. Penelitian ini akan menggunakan 6 metrik evaluasi yang meliputi Accuracy, Precision, Recall, F1-Score, dan Specificity [20], [21], [22], [23]. Selain itu, ROC AUC digunakan karena mampu menggambarkan trade-off antara True Positive Rate dan False Positive Rate. Literatur terbaru juga menunjukkan bahwa AUC dapat dianggap sebagai

bentuk balanced accuracy yang relevan untuk kasus imbalanced classification [24].

Berdasarkan temuan-temuan tersebut, penelitian ini bertujuan untuk membandingkan dan menganalisis pengaruh tiga teknik oversampling, yaitu SMOTE, ROS, ADASYN terhadap performa beberapa algoritma klasifikasi seperti Logistic Regression, SVM, LDA, dan ANN pada dataset PPG-BP. 6 metrik evaluasi juga digunakan untuk memberikan gambaran detail mengenai performa model dalam mengidentifikasi pasien. Dengan membandingkan berbagai teknik oversampling yang dikombinasikan dengan model machine learning, penelitian ini diharapkan dapat memberikan acuan terkait teknik yang paling efektif dalam membangun model deteksi dan klasifikasi hipertensi dengan performa yang lebih efektif.

II. METODE

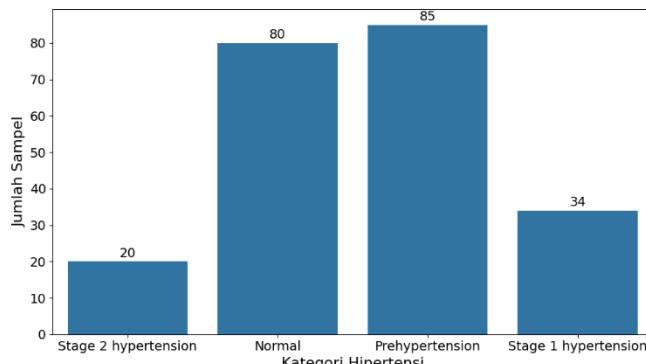
Penelitian ini dimulai dengan pengumpulan dataset dari platform Figshare, yang kemudian diikuti dengan rangkaian tahapan preprocessing yang mencakup pemeriksaan tipe data, menghapus fitur tidak relevan, identifikasi dan penanganan missing values dengan imputasi data baik numerik maupun kategorikal, menghapus data duplikat, dan konversi fitur kategorikal menjadi numerik. Setelah preprocessing selesai, dataset dibagi menjadi data training dan data testing. Data yang sudah dilakukan split kemudian akan dinormalisasi menggunakan Z-Score. Tahap berikutnya adalah penanganan ketidakseimbangan kelas menggunakan tiga teknik oversampling, yaitu SMOTE, ROS, dan ADASYN. Setelah itu, empat algoritma machine learning seperti SVM, Logistic Regression, LDA, dan ANN akan dilatih pada data training menggunakan Hyperparameter GridSearchCV. Terakhir hasil evaluasi performa masing-masing teknik oversampling yang dikombinasikan dengan model machine learning akan dibandingkan dengan kondisi tanpa oversampling. Diagram alur penelitian ini disajikan pada Gambar 1



Gambar 1. Alur Penelitian

A. Pengumpulan Data

Pada tahapan pengumpulan data, penelitian ini menggunakan dataset PPG-BP Dataset yang diperoleh dari platform Figshare. Dataset ini dikembangkan oleh Liang Yongbo et al. [25] menggunakan data pasien rawat inap di Guilin People's Hospital, Tiongkok, dengan tujuan untuk menganalisis skrining awal gangguan kardiovaskular seperti hipertensi, diabetes, dan penyakit kardiovaskular lainnya tanpa prosedur medis yang invasif. Dataset ini terdiri dari 657 rekam data yang dikumpulkan dari 219 subjek orang dewasa dengan rentang usia 20–89 tahun. Dataset ini memiliki beberapa fitur yang akan digunakan dalam penelitian ini, antara lain Age (Usia), Sex (Jenis Kelamin), Height (Tinggi), Weight (Berat), Systolic Blood Pressure (SBP), Diastolic Blood Pressure (DBP), dan Heart Rate (Denyut Jantung), serta variabel target Hypertension yang terdiri dari empat kategori diagnostik, yaitu Normal, Prehypertension, Stage 1 hypertension, dan Stage 2 hypertension [26]. Distribusi dataset asli pada kelas target Hypertension dapat dilihat pada Gambar 2.



Gambar 2. Distribusi Kelas Target 'Hypertension' pada Data Asli

Gambar 2 menunjukkan bahwa distribusi kelas Hypertension pada dataset memiliki data yang tidak seimbang dengan rincian jumlah Normal 80, Pre-Hypertension 85, Stage 1 Hypertension 34, dan Stage 2 Hypertension 20. Dengan demikian proses oversampling sangat dibutuhkan dalam menangani kasus imbalance ini.

B. Preprocessing

Preprocessing merupakan tahapan penting untuk memastikan kualitas data dalam keadaan optimal sebelum proses pemodelan dilakukan, teknik preprocessing dapat memengaruhi kinerja algoritma klasifikasi secara signifikan [27]. Pada penelitian ini, preprocessing mencakup beberapa langkah utama, dimulai dengan melakukan pemeriksaan tipe data untuk memastikan bahwa setiap fitur berada dalam format yang sesuai dan kompatibel dengan algoritma pemodelan [28]. Dilanjutkan dengan menghapus fitur yang tidak relevan, proses ini dapat meningkatkan akurasi model karena mengurangi noise dan membuat algoritma lebih fokus pada fitur yang benar-benar berkontribusi terhadap model [29]. Setelah itu dilakukan identifikasi dan penanganan

missing values pada tiap fitur. Data numerik akan diimputasi menggunakan nilai median dari fitur tersebut, Yin et al. [30] menyebutkan bahwa median merupakan teknik imputasi yang stabil terhadap outlier. Sementara itu data kategorikal akan diimputasi menggunakan label 'unknown', pendekatan ini dipilih karena ukuran dataset penelitian relatif kecil atau hanya 219 baris, sehingga penghapusan baris dapat mengurangi representativitas data, dengan demikian setiap data yang hilang tetap dapat direpresentasikan tanpa harus membuang baris. Wu et al. [31] melaporkan bahwa sekitar 18,8% penelitian klinis telah menggunakan metode ini untuk menangani missing values. Ketiga dilakukan identifikasi dan penghapusan data duplikat untuk menjamin keunikan setiap entri dalam dataset. Data duplikat dapat menimbulkan over-representasi, serta dapat menurunkan kualitas analisis dan performa model. Menurut Alotaibi et al. penghapusan duplikat merupakan tahap penting untuk meningkatkan konsistensi data [32]. Terakhir, seluruh fitur kategorikal dikonversi menjadi numerik agar data dapat dipahami dan diproses oleh algoritma. Dai et al. [33] menekankan bahwa untuk mendapatkan hasil pemodelan yang lebih optimal, data harus dikonversi terlebih dahulu ke dalam bentuk numerik.

C. Pembagian Data

Setelah melalui tahap preprocessing, dataset akan dibagi menjadi dua bagian, yaitu data training dan data testing. Pembagian ini dilakukan dengan proporsi 80% untuk data training dan 20% untuk data testing. Data training digunakan untuk melatih model, sedangkan data testing digunakan untuk mengukur dan mengevaluasi performa model yang telah dilatih. Pembagian data ditujukan untuk memungkinkan model mempelajari pola dari data training dan menguji kemampuannya dalam mengklasifikasi hasil pada data testing yang belum pernah dilihat sebelumnya, sehingga dapat dinilai kemampuannya dalam menggeneralisasi model [34].

D. Normalisasi

Setelah pembagian data, langkah selanjutnya adalah normalisasi, langkah ini bertujuan untuk menyamakan skala antar fitur sehingga tidak ada fitur yang mendominasi selama proses training model. Normalisasi dilakukan menggunakan Z-Score Scaling. Metode ini menghitung deviasi setiap nilai fitur dari rata-rata dalam satuan standar deviasi, sehingga menghasilkan fitur dengan rata-rata nol dan standar deviasi satu [35]. Z-Score Scaling atau juga disebut Standard Scaling digunakan untuk menstandarisasi data dengan menyesuaikan nilai fitur sehingga memiliki rata-rata 0 dan standar deviasi 1. Metode ini menghitung sejauh mana suatu nilai menyimpang dari rata-rata dalam satuan standar deviasi, sehingga memungkinkan perbandingan antar fitur dengan skala yang berbeda [36]. Penggunaan Z-Score Scaling sangat tepat ketika fitur-fitur dalam dataset memiliki distribusi yang berbeda dan perlu distandarisasi untuk memudahkan perbandingan antar fitur. Dengan menggunakan Z-Score Scaling pada PPG-BP datasets, nilai-nilai dalam fitur dapat distandarisasi, sehingga memastikan bahwa fitur dengan variabilitas tinggi tidak

mendominasi proses pemodelan. Adapun rumus Z-Score Scaling adalah sebagai berikut:

$$z = \frac{x - \mu}{\sigma}$$

di mana x adalah nilai fitur asli, μ adalah rata-rata fitur, dan σ adalah standar deviasi. Dengan pendekatan ini, setiap fitur memiliki bobot yang seimbang dalam analisis dan mendukung kestabilan model klasifikasi yang digunakan.

E. Oversampling

Setelah langkah normalisasi dilakukan, tahap berikutnya adalah oversampling, teknik oversampling dilakukan guna menyeimbangkan distribusi kelas dengan menambah sampel yang terdapat pada kelas minoritas. Pendekatan ini bertujuan mengurangi bias model terhadap kelas mayoritas dan meningkatkan kemampuan algoritma dalam mengenali kelas minoritas, sehingga performa klasifikasi menjadi lebih optimal [37]. Oversampling juga membantu model menjadi lebih peka dalam mengenali kasus yang jarang muncul [38]. Pada penelitian ini, oversampling dilakukan menggunakan tiga teknik, yaitu SMOTE, ROS, dan ADASYN. Untuk lebih lanjut mengenai masing-masing teknik akan dijelaskan sebagai berikut:

1. Synthetic Minority Oversampling Technique (SMOTE)

SMOTE menghasilkan sampel sintetis baru dengan melakukan interpolasi antara data minoritas dan tetangga terdekatnya. Pendekatan ini membantu memperluas representasi kelas minoritas sekaligus menjaga struktur ruang fitur. Penelitian Elreedy dan Atiya menunjukkan bahwa SMOTE mampu menghasilkan distribusi sintetis yang lebih stabil dan mengurangi bias terhadap kelas mayoritas [39].

2. Random Oversampling (ROS)

ROS menambahkan sampel baru dengan cara menduplikasi data pada kelas minoritas secara acak. Metode ini mudah diterapkan dan efektif untuk dataset berukuran kecil atau ketika variasi kelas minoritas sudah cukup representatif. Namun, karena hanya menyalin data yang sama, ROS memiliki potensi meningkatkan overfitting apabila tidak disertai dengan langkah pembersihan data [40].

3. Adaptive Synthetic Sampling (ADASYN)

ADASYN merupakan Teknik oversampling yang mirip dengan SMOTE namun bersifat adaptif. ADASYN meningkatkan jumlah sampel sintetis terutama pada area kelas minoritas yang relatif sulit dipelajari atau berada dekat dengan kelas mayoritas. Dengan pendekatan ini, model dapat mempelajari pola minoritas secara lebih detail. Penelitian oleh Tekkali [41] menunjukkan bahwa pendekatan adaptif ADASYN mampu meningkatkan performa model pada dataset sangat tidak seimbang dibandingkan metode oversampling tradisional.

F. Model Machine Learning

Setelah proses oversampling dilakukan, pada tahap ini data training digunakan untuk melatih model machine

learning. Untuk memastikan setiap algoritma mencapai performa optimal, penelitian ini menerapkan GridSearchCV sebagai metode hyperparameter tuning. Teknik ini melakukan pencarian sistematis terhadap kombinasi hyperparameter terbaik melalui skema 5-fold cross-validation. Pendekatan tersebut dipilih karena mampu meningkatkan kinerja model secara konsisten sebagaimana dilaporkan pada beberapa studi sebelumnya [42], [43], [44]. Proses pelatihan dimulai dengan menetapkan ruang hyperparameter untuk masing-masing algoritma yang diuji. Rincian konfigurasi parameter yang diuji ditampilkan pada Tabel 1.

TABEL 1
RUANG PENCARIAN HYPERPARAMETER UNTUK OPTIMASI MODEL

Algoritma	Hyperparameter	Ruang Pencarian
Logistic Regression	C	0.1, 1.0, 10.0
	solver	liblinear, lbfgs
SVM	C	0.1, 1.0, 10.0
	gamma	scale, 0.1, 0.01
LDA	solver	svd, lsqr
	hidden_layer_size	(100,), (50, 25), (5, 2)
ANN	solver	'adam', 'lbfgs'
	alpha	0.0001, 1e-5

Dari Tabel 1, dapat dijelaskan bahwa pada algoritma Logistic Regression (LR), model dioptimalkan menggunakan parameter C (0.1, 1.0, 10.0), yang berfungsi mengatur tingkat regularisasi, serta dua jenis solver, yaitu liblinear dan lbfgs. Variasi ini memungkinkan model mencapai keseimbangan antara kompleksitas dan generalisasi, lalu pada Support Vector Machine (SVM). Hyperparameter utama yang diuji meliputi C (0.1, 1.0, 10.0) untuk mengendalikan kesalahan klasifikasi, serta gamma (scale, 0.1, 0.01) untuk menentukan pengaruh radius kernel RBF. Kombinasi nilai ini membantu menentukan margin yang optimal pada ruang fitur. Selanjutnya pada Linear Discriminant Analysis (LDA), model dievaluasi menggunakan dua solver, svd dan lsqr. Pemilihan solver yang tepat dapat menjaga kestabilan perhitungan matriks kovarians ketika memproyeksikan data ke ruang diskriminatif. Terakhir pada Artificial Neural Network (ANN). Algoritma diuji dengan tiga arsitektur hidden layer (100,), (50,25), dan (5,2) untuk melihat pengaruh kompleksitas jaringan terhadap performa. Dua metode optimasi adam dan lbfgs serta nilai regularisasi alpha (0.0001 dan 1e-5) digunakan untuk mengontrol konvergensi dan mengurangi overfitting.

Setelah menentukan pilihan parameter, langkah berikutnya adalah melihat bagaimana proses pencarian parameter tersebut dijalankan. Tabel 2 menunjukkan pengaturan yang digunakan oleh GridSearchCV selama melakukan tuning, seperti model yang dievaluasi, daftar parameternya, jumlah fold, hingga metrik yang digunakan.

TABEL 2
KONFIGURASI METODE OPTIMASI GRIDSEARCHCV DAN SKEMA VALIDASI

Komponen Optimasi	Detail Konfigurasi
estimator	base_model
param_grid	param_grids[name]
cv	5
scoring	f1_macro
n_jobs	-1

Dari Tabel 2 kita bisa melihat pengaturan yang digunakan dalam proses GridSearchCV saat melakukan pencarian parameter terbaik. Setiap argumen memiliki fungsi yang berbeda dalam memastikan proses tuning berjalan optimal. Argumen estimator menentukan model yang sedang diuji, sedangkan param_grid berisi daftar kombinasi parameter yang akan dicoba. Nilai cv menunjukkan bahwa pencarian dilakukan menggunakan 5-fold cross-validation. Selanjutnya, argumen scoring menggunakan metrik F1-macro agar penilaian model tetap seimbang pada dataset yang sempat imbalanced. Terakhir, n_jobs = -1 memungkinkan seluruh inti prosesor bekerja secara paralel sehingga proses tuning menjadi lebih cepat.

Untuk algoritma machine learning yang digunakan dalam penelitian ini adalah Logistic Regression (LR), Support Vector Machine (SVM), Linear Discriminant Analysis (LDA), dan Artificial Neural Network (ANN). Penjelasan lebih lanjut mengenai masing-masing Teknik akan dijelaskan sebagai berikut:

1. Logistic Regression (LR)

LR bekerja dengan mempelajari hubungan linier antar fitur, kemudian mengonversinya menjadi probabilitas melalui fungsi sigmoid. Selama pelatihan, model mencari koefisien terbaik dengan meminimalkan log-loss menggunakan metode optimasi seperti gradient descent. LR telah digunakan dalam analisis risiko penyakit kardiovaskular dan terbukti mampu menghasilkan performa prediksi yang baik [45].

2. Support Vector Machine (SVM)

SVM adalah algoritma klasifikasi margin-based yang bekerja dengan mencari hyperplane terbaik yang memisahkan kelas dengan margin terbesar. Dengan bantuan kernel, SVM dapat memetakan data ke ruang berdimensi lebih tinggi sehingga pola non-linier tetap dapat dipisahkan. Dalam penelitian ini, SVM dipilih karena kemampuannya dalam menangani variasi distribusi fitur dan ketidakteraturan pola dengan mencari hyperplane optimal melalui mekanisme margin maximization. Kernel Radial Basis Function (RBF) dengan parameter gamma digunakan untuk memodelkan hubungan non-linear yang kompleks antar fitur. Sebagaimana dilaporkan oleh Nuryani et al. [46], SVM menunjukkan performa yang stabil dan kompetitif dalam deteksi hipertensi. Stabilitas ini secara teoretis dapat dijelaskan melalui mekanisme margin maximization, di mana posisi hyperplane ditentukan terutama oleh support vectors, yaitu sampel-

sampel yang berada di sekitar batas kelas. Karakteristik ini menyebabkan SVM relatif robust terhadap penambahan data sintetis melalui teknik oversampling, karena penambahan sampel di luar area margin tidak secara signifikan memengaruhi posisi batas keputusan selama struktur data di sekitar support vectors tetap terjaga.

3. Linear Discriminant Analysis (LDA)

LDA adalah generalisasi dari sudut diskriminan linier. adalah metode yang digunakan dalam statistik, pattern recognition, dan machine learning untuk menemukan kombinasi fitur linier, atau untuk memisahkan dua atau lebih objek atau peristiwa. Kombinasi yang dihasilkan dapat digunakan sebagai pengklasifikasi linier atau umum digunakan dalam reduksi dimensi awal. Tujuan linear discriminant analysis (LDA) adalah untuk mengklasifikasikan objek ke dalam sejumlah kelas sesuai dengan karakteristik yang dijelaskan. Dalam analisis diskriminan linier, suatu objek memiliki dua variabel: kelas/variabel terkait (variabel dependen) dan atribut/variabel tidak terkait (variabel independen). Variabel dependen dikaitkan dengan variabel independen yang menggambarkan variabel tersebut. Penelitian Rizal et al. [47] telah menggunakan LDA untuk mengklasifikasi penyakit jantung dan mendapatkan performa yang baik dengan akurasi 81,22%.

4. Artificial Neural Network (ANN)

ANN terdiri dari lapisan neuron yang saling terhubung dan mampu mempelajari pola non-linier. Setiap neuron melakukan operasi linier yang dilanjutkan dengan fungsi aktivasi, memungkinkan model menangkap hubungan yang lebih kompleks antar fitur. Bobot antar neuron diperbarui melalui proses backpropagation untuk meminimalkan error. ANN dipilih karena kemampuannya menangkap pola non-linier pada fitur-fitur sinyal PPG yang mungkin tidak dapat ditangani oleh model linier. Bukhari et al. melaporkan bahwa ANN dapat mencapai akurasi lebih tinggi dibanding metode lain dalam tugas diagnosis medis, mendukung penerapannya pada prediksi penyakit seperti hipertensi [48].

G. Evaluasi

Pada tahap evaluasi, kinerja model yang telah dilatih dengan kombinasi hyperparameter tuning GridSearch CV akan diuji menggunakan metrik evaluasi untuk menilai efektivitasnya dalam mengidentifikasi kelas hipertensi. Tujuan utama evaluasi adalah guna mengidentifikasi teknik oversampling yang paling unggul dalam meningkatkan model klasifikasi, dengan menganalisis berbagai kelas hipertensi untuk mengamati dampak dari masing-masing teknik oversampling terhadap hasil yang didapatkan. Evaluasi dilakukan dengan mempertimbangkan empat komponen utama dalam analisis klasifikasi, yaitu True Positive (TP) yang menunjukkan jumlah kasus ketika model benar dalam mengklasifikasikan pasien mengidap hipertensi, True Negative (TN) yang mengindikasikan jumlah kasus ketika model dengan benar mengklasifikasikan pasien tidak mengidap hipertensi, False Positive (FP) yang terjadi saat

model salah mengklasifikasikan pasien mengidap hipertensi padahal sebenarnya tidak, serta False Negative (FN) yang terjadi ketika model salah mengklasifikasikan pasien tidak mengidap hipertensi padahal sebenarnya mengidap [49]. Keempat komponen ini menjadi dasar dalam menghitung berbagai metrik evaluasi yang digunakan untuk mengukur kinerja berbagai algoritma Logistic Regression, SVM, LDA, dan ANN secara menyeluruh. Hasil evaluasi mencakup berbagai metrik yang digunakan untuk mengukur kinerja model yang akan dijelaskan sebagai berikut [50]:

1. Accuracy

Metrik accuracy mengukur sejauh mana model dapat melakukan klasifikasi yang benar dibandingkan dengan total klasifikasi. Akurasi memberikan gambaran umum tentang performa model, tetapi kurang berguna pada data yang tidak seimbang. Akurasi dihitung dengan rumus:

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

2. Precision

Metrik ini mengukur akurasi klasifikasi positif yang dilakukan oleh model. Metrik ini digunakan untuk mengetahui seberapa banyak klasifikasi positif yang akurat atau sesuai dengan kondisi yang sebenarnya. Precision dihitung dengan rumus:

$$\text{Precision} = \frac{TP}{TP + FP}$$

3. Recall

Recall mengukur kemampuan model dalam menemukan semua kasus positif yang sebenarnya. Metrik ini penting ketika sangat penting untuk mendeteksi sebanyak mungkin kasus positif. Recall dihitung dengan rumus:

$$\text{Recall} = \frac{TP}{TP + FN}$$

4. F1-Score

F1-Score adalah rata-rata harmonik antara precision dan recall. Metrik ini penting untuk memberikan gambaran keseimbangan antara kemampuan model dalam mendeteksi kelas positif dan negatif, terutama pada dataset yang tidak seimbang. F1-score dihitung dengan rumus:

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

5. Specificity

Specificity mengukur kemampuan model dalam mengidentifikasi data negatif dengan benar. Metrik ini penting untuk memastikan model tidak salah mengklasifikasikan data negatif sebagai positif. Specificity dihitung dengan rumus:

$$\text{Specificity} = \frac{TN}{TN + FP}$$

6. ROC AUC

ROC-AUC mengukur kemampuan model untuk membedakan antara dua kelas (positif dan negatif) di seluruh

rentang nilai threshold yang mungkin. Nilai AUC yang lebih tinggi menunjukkan bahwa model memiliki kemampuan yang lebih baik dalam membedakan antara kelas positif dan negatif, bahkan ketika threshold berubah.

III. HASIL DAN PEMBAHASAN

Pada bab ini, akan dijelaskan hasil yang diperoleh dari setiap tahapan yang telah dijelaskan pada bagian metode penelitian. Berikut adalah penjelasan mengenai hasil yang didapat dari proses yang telah dilakukan:

A. Preprocessing

Pada tahap awal preprocessing, dilakukan pemeriksaan tipe data untuk memastikan bahwa setiap fitur dalam dataset berada dalam format yang sesuai. Pada algoritma Logistic Regression, SVM, LDA, dan ANN, memiliki karakteristik komputasi yang bergantung pada perhitungan angka matematis. Oleh karena itu, jika dataset berisi data non-numerik, algoritma-algoritma ini tidak dapat melakukan proses perhitungan dan pemodelan dengan baik. Oleh karena itu, dengan memastikan bahwa seluruh fitur berada dalam format numerik, algoritma dapat bekerja dengan optimal dan memberikan performa klasifikasi yang lebih akurat, selain itu juga dilakukan penyesuaian dan penyederhanaan nama kolom agar lebih mudah diproses pada tahap pemodelan berikutnya.

TABEL 3
TIPE DATA DALAM FITUR DATASET

Fitur	Tipe Data
Num	int64
Subject_ID	int64
Sex	object
Age	int64
Height	int64
Weight	int64
SBP	int64
DBP	int64
Heart Rate	int64
BMI	float64
Hypertension	object

Dari Tabel 3, terlihat bahwa sebagian besar fitur memiliki tipe data int64 dan float64 yang tergolong tipe numerik, namun masih terdapat juga fitur kategorikal object yaitu pada 'Sex' yang terdiri dari Male dan Female. Serta pada 'Hypertension' yang terdiri dari Normal, Prehypertension, Stage 1 Hypertension, Stage 2 hypertension. Keberadaan fitur kategorikal dalam dataset ini harus dikonversi ke dalam bentuk numerik yang prosesnya akan dilakukan pada akhir tahap preprocessing untuk memastikan pembersihan data berlangsung pada data asli dan tanpa gangguan dari nilai hasil encoding. Langkah berikutnya adalah menghapus fitur tidak relevan agar model tidak terbebani oleh noise dan fitur yang tidak memberikan

kontribusi penting. Selanjurnya dilakukan pengecekan dan penanganan pada missing values. Data numerik dengan missing values akan diimputasi menggunakan nilai median, sementara itu data kategorikal akan diimputasi menggunakan label ‘unknown’. Imputasi dipilih karena ukuran dataset yang digunakan pada penelitian ini relatif kecil, sehingga penghapusan baris berpotensi mengurangi representativitas data. Setelah itu data duplikat akan dihapus guna mencegah pengaruhnya terhadap hasil analisis, yang dapat menyebabkan model menghasilkan hasil yang tidak akurat.

TABEL 4
MISSING VALUES DAN DATA DUPLIKAT DALAM FITUR DATASET

Fitur	Missing Values	Data Duplikat
Sex(M/F)	0	0
Age(year)	0	0
Height(cm)	0	0
Weight(kg)	0	0
SBP	0	0
DBP	0	0
Heart Rate	0	0
BMI	0	0
Hypertension	0	0

Dapat dilihat dari Tabel 4, Fitur yang tidak relevan telah dihapus, serta tidak ada fitur dalam dataset yang memiliki

missing values maupun data duplikat. Oleh karena itu penghapusan dan penanganan lebih lanjut terkait missing values dan data duplikat tidak perlu dilakukan. Langkah terakhir adalah mengonversi fitur kategorikal menjadi numerik, fitur ‘Sex’ diubah nilainya menjadi format numerik menggunakan LabelEncoder, sehingga jika male menjadi 0 dan female menjadi 1. Serta Pada fitur ‘Hypertension’, kelas Normal diubah menjadi angka 0, Prehypertension menjadi 1, Stage 1 hypertension menjadi 2 dan Stage 2 hypertension menjadi 3.

B. Pembagian Data

Pada tahap ini, dataset akan dibagi menjadi dua bagian, yaitu data training dan data testing. Pembagian data ini dilakukan dengan proporsi 80% untuk data training dan 20% untuk data testing. Data training digunakan untuk melatih model, sedangkan data testing digunakan untuk menguji performa model setelah dilatih. Dataset yang telah dibagi disajikan dalam Tabel 6. Data training terdiri dari 175 baris dan 9 kolom, sedangkan untuk data testing terdiri dari 44 baris, 9 kolom, dengan demikian, dataset sudah siap untuk dilakukan normalisasi.

C. Normalisasi

Pada tahap normalisasi, data yang telah dibagi menjadi data training dan data testing akan dilakukan normalisasi. Penyajian data asli akan ditampilkan pada Tabel 5 dan Tabel 6.

TABEL 5
DATA TRAINING SEBELUM DINORMALISASI

Sex	Age	Height	Weight	SBP	DBP	Heart Rate	BMI	Hypertension (label)
1	61	165	66	173	100	89	24.242424	3
1	45	168	69	117	74	59	24.447279	0
0	67	156	56	113	55	61	23.011177	0
1	46	165	56	116	56	72	20.569330	0
1	51	170	65	123	78	72	22.491349	1

TABEL 6
DATA TESTING SEBELUM DINORMALISASI

Sex	Age	Height	Weight	SBP	DBP	Heart Rate	BMI	Hypertension (label)
0	42	152	49	100	63	67	21.208449	0
0	55	155	50	174	104	96	20.811655	3
0	50	157	50	160	93	76	20.284799	3
0	24	155	38	96	67	82	15.816857	0
0	72	157	39	122	65	83	15.822143	1

Berdasarkan Tabel 5 dan Tabel 6, terlihat bahwa setiap fitur memiliki rentang nilai yang berbeda-beda secara signifikan. Fitur seperti Systolic Blood Pressure dan Diastolic Blood Pressure memiliki nilai yang jauh lebih besar

dibandingkan fitur lain seperti BMI atau variabel kategori yang telah dikodekan secara biner seperti Hypertension. Ketidakseimbangan skala antar variabel ini dapat menyebabkan fitur dengan rentang besar mendominasi proses

pembelajaran model, sehingga memengaruhi kestabilan dan kinerja dari algoritma. Temuan ini menunjukkan bahwa proses normalisasi diperlukan agar seluruh fitur berada pada skala yang sebanding, sehingga model dapat menghasilkan prediksi yang lebih akurat. Pada penelitian ini normalisasi dilakukan menggunakan metode Z-Score. Metode ini menormalisasi data dengan cara menstandarisasi fitur menggunakan nilai mean dan standar deviasi yang dihitung dari data training sehingga memiliki distribusi dengan nilai mean 0 dan standar

deviasi 1. Parameter tersebut lalu diimplementasikan pada data testing tanpa menghitung ulang statistik yang baru. Skala fitur pada data uji akan disesuaikan menggunakan parameter yang diperoleh dari data training guna memastikan konsistensi dari kedua dataset tersebut, sehingga distribusi pada data testing menjadi serupa dengan data training. Hasil normalisasi dari Z-Score dapat dilihat secara rinci pada Tabel 7 untuk data training dan Tabel 8 untuk data testing.

TABEL 7
DATA TRAINING SESUDAH DINORMALISASI

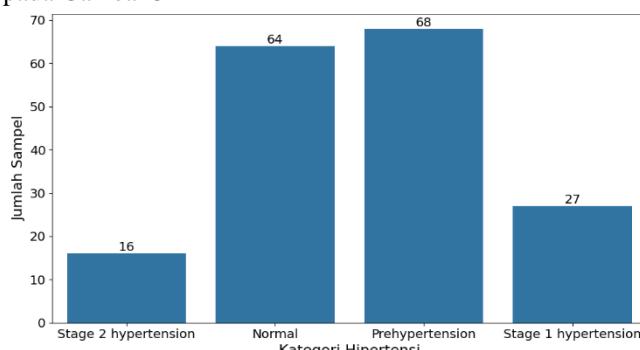
Sex	Age	Height	Weight	SBP	DBP	Heart Rate	BMI	Hypertension (label)
1.028992	0.247488	0.431693	0.479316	2.204512	2.612500	1.556332	0.291002	0
1.028992	-0.742464	0.795479	0.733499	-0.543417	0.216919	-1.390203	0.343378	3
-0.971825	0.618720	-0.659666	-0.367960	-0.739698	-1.533698	-1.193768	-0.023798	3
1.028992	-0.680592	0.431693	-0.367960	-0.592487	-1.441561	-0.113371	-0.648120	0
1.028992	-0.371232	1.038004	0.394589	-0.248996	0.585469	-0.113371	-0.156706	1

TABEL 8
DATA TESTING SESUDAH DINORMALISASI

Sex	Age	Height	Weight	SBP	DBP	Heart Rate	BMI	Hypertension (label)
-0.971825	-0.928081	-1.144714	-0.961053	-1.377610	-0.796597	-0.604461	-0.484713	0
-0.971825	-0.123744	-0.780928	-0.876326	2.253582	2.981051	2.243857	-0.586163	3
-0.971825	-0.433104	-0.538404	-0.876326	1.566600	1.967535	0.279500	-0.720868	3
-0.971825	-2.041777	-0.780928	-1.893057	-1.573890	-0.428046	0.868807	-1.863213	0
-0.971825	0.928081	-0.538404	-1.808330	-0.298066	-0.612321	0.967025	-1.861861	1

D. Oversampling

Setelah split dan normalisasi, selanjutnya akan dilakukan oversampling pada data training. Distribusi kelas pada fitur Hypertension sebelum dilakukan oversampling dapat dilihat pada Gambar 3



Gambar 3. Distribusi Kelas Target 'Hypertension' Sebelum Oversampling

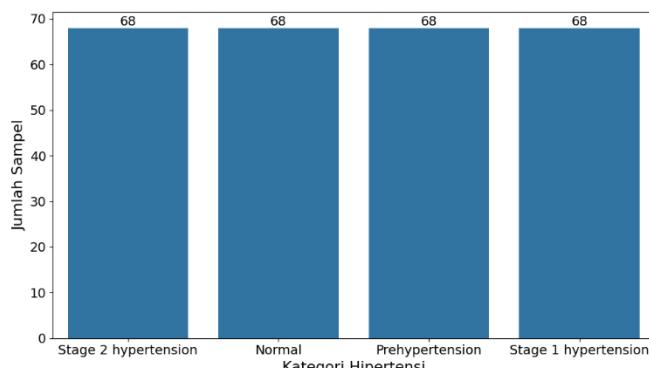
Gambar 3 memberikan gambaran bahwa data pada fitur Hypertension memiliki jumlah yang beragam di tiap kelasnya, Normal 64, Pre-Hypertension 68, Stage 1 Hypertension 27, dan Stage 2 Hypertension 16. Dengan demikian penanganan ketidakseimbangan kelas akan

dilakukan pada fitur Hypertension menggunakan teknik oversampling. Selanjutnya, akan dibahas lebih lanjut mengenai teknik oversampling yang digunakan:

1. Synthetic Minority Oversampling Technique (SMOTE)

SMOTE melakukan oversampling dengan cara menghasilkan sampel sintetis baru pada kelas minoritas melalui proses interpolasi. Metode ini membuat data baru dengan memilih sebuah sampel dari kelas minoritas, kemudian menentukan k-nearest neighbors terdekatnya, dan membentuk titik baru pada ruang fitur berdasarkan selisih antara sampel tersebut dengan salah satu tetangganya. Pendekatan ini bertujuan memperluas wilayah representasi kelas minoritas sehingga distribusinya menjadi lebih seimbang bukan sekedar menggandakan data yang sama. Berdasarkan hasil eksekusi teknik SMOTE, jumlah sampel pada data training mengalami peningkatan. Sebelum dilakukan resampling, data training memiliki dimensi (175, 8), yang berarti terdapat 175 sampel pasien dengan 8 fitur. Setelah proses SMOTE diterapkan, ukuran data latih berubah menjadi (272, 8). Artinya, ada penambahan 97 sampel sintetis baru untuk menyeimbangkan seluruh kelas. Jumlah akhir 272 sampel ini diperoleh dengan menyamakan jumlah data pada

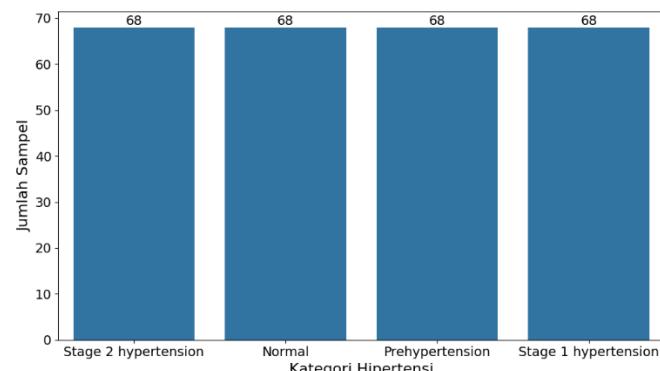
empat kategori hipertensi (Normal, Prehypertension, Stage 1, dan Stage 2), sehingga masing-masing kelas memiliki jumlah sampel yang setara dengan kelas mayoritas, yaitu 68 sampel per kelas, dengan jumlah fitur yang tetap sama, yakni 8 fitur. Hasil oversampling dari SMOTE dapat dilihat pada Gambar 4.



Gambar 4. Distribusi Kelas Target 'Hypertension' Setelah SMOTE

2. Random Oversampling (ROS)

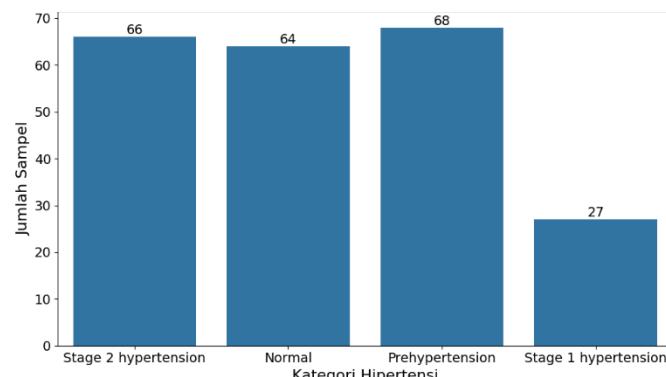
ROS melakukan oversampling dengan cara menambahkan jumlah sampel pada kelas minoritas melalui proses duplikasi acak terhadap instance yang sudah ada. Tidak seperti SMOTE, ROS tidak membentuk sampel baru, melainkan memperbanyak data minoritas secara langsung sampai proporsinya seimbang dengan kelas mayoritas. Metode ini termasuk pendekatan yang paling dasar dalam oversampling dan sering digunakan karena implementasinya sederhana serta mampu mengatasi ketidakseimbangan kelas dengan cepat. Namun, karena hanya menggandakan data, ROS memiliki potensi meningkatkan overfitting pada model yang sensitif terhadap duplikasi, terutama pada algoritma berbasis pohon atau jarak. Berdasarkan hasil eksekusi teknik ROS, memberikan perubahan pada jumlah data training yang secara kuantitatif serupa dengan hasil SMOTE. Sebelum dilakukan oversample dengan ROS, data training hanya berjumlah 175 sampel dan menunjukkan perbedaan yang cukup jelas antar kelas. Melalui ROS, dataset kemudian diseimbangkan dengan menjadikan jumlah sampel pada kelas mayoritas yaitu 68 sampel sebagai acuan. Setelah proses tersebut, dimensi data latih bertambah dari (175, 10) menjadi (272, 10). Walaupun jumlah total sampel setelah oversampling ROS sama dengan hasil dari Teknik SMOTE, perlu diperhatikan bahwa ROS menambahkan data dengan cara menggandakan sampel yang sudah ada. Pendekatan ini memberikan bobot tambahan bagi kelas minoritas agar model dapat mempelajari pola dari kelas tersebut dengan lebih baik selama proses pelatihan. Hasil oversampling dari ROS dapat dilihat pada Gambar 5.



Gambar 5. Distribusi Kelas Target 'Hypertension' Setelah ROS

3. Adaptive Synthetic Sampling (ADASYN)

ADASYN melakukan oversampling dengan cara menekankan pembuatan sampel sintetis pada area yang dianggap lebih sulit dipelajari. Teknik ini menghitung tingkat kesulitan setiap sampel minoritas berdasarkan proporsi tetangga dari kelas mayoritas di sekitarnya. Sampel yang berada pada wilayah dengan ketidakseimbangan tinggi atau batas keputusan yang kompleks akan memperoleh jumlah sintesis yang lebih besar. Dengan demikian, ADASYN mendorong model untuk lebih fokus pada area kelas dengan memiliki data yang sedikit, sekaligus meningkatkan kemampuan generalisasi terhadap pola yang jarang muncul. Penerapan Adaptive Synthetic Sampling (ADASYN) pada data latih menghasilkan distribusi kelas yang lebih "selektif" dibandingkan metode oversampling lainnya. Tidak seperti SMOTE atau ROS yang menambah data secara merata, ADASYN hanya menambahkan sampel sintetis pada kelas yang dianggap paling sulit dipelajari oleh model. Hal ini terlihat dari hasil resampling, di mana kelas Stage 2 Hypertension meningkat signifikan hingga 66 sampel, sementara kelas Stage 1 Hypertension tetap pada 27 sampel dan kelas Normal berada di 64 sampel tanpa penambahan data apa pun. Pola ini muncul karena ADASYN bekerja berdasarkan weighted density distribution, yaitu algoritma akan fokus menghasilkan sampel baru pada kelas yang berada di wilayah sulit. Dengan kata lain, ADASYN menilai bahwa kelas Stage 2 merupakan extreme minority dengan tingkat kesulitan belajar paling tinggi, sehingga membutuhkan lebih banyak data sintetis. Sebaliknya, tidak ada penambahan sampel pada kelas Stage 1 karena algoritma memandang kelas ini sudah berada pada area yang relatif aman atau terklaster dengan baik. Artinya, penambahan data di kelas tersebut tidak akan memberikan dampak berarti terhadap kemampuan model membangun batas keputusan. Hasil ini menunjukkan bahwa cara kerja ADASYN tidak sekedar menyamakan jumlah sampel tiap kelas, tetapi juga menyeimbangkan data secara adaptif sesuai kebutuhan topologi dataset. Hasil oversampling dari ROS dapat dilihat pada Gambar 6.



Gambar 6. Distribusi Kelas Target 'Hypertension' Setelah ADASYN

E. Model Machine Learning

Dalam pemodelan machine learning, pembahasan lebih lanjut mengenai masing-masing teknik akan dijelaskan sebagai berikut:

1. Logistic Regression (LR)

Pada penelitian ini, Logistic Regression digunakan sebagai model klasifikasi berbasis fungsi logit yang memetakan hubungan antara fitur masukan dengan probabilitas kelas. Model ini dilatih menggunakan seluruh data hasil oversampling dengan proses optimasi hyperparameter melalui GridSearchCV. Parameter yang dituning meliputi nilai regularisasi C (0.1, 1.0, 10.0) serta jenis solver (liblinear dan lbfgs). Proses ini memungkinkan LR memilih konfigurasi terbaik untuk memaksimalkan performa F1-Macro pada data yang telah diseimbangkan.

2. Support Vector Machine (SVM)

Algoritma SVM bertujuan mencari hyperplane terbaik yang memisahkan kelas secara optimal. Pada penelitian ini, SVM dioptimalkan menggunakan GridSearchCV dengan kombinasi parameter C (0.1, 1.0, 10.0) dan gamma (scale, 0.1, 0.01). Penggunaan kernel RBF memungkinkan model menangkap pola non-linear pada dataset hipertensi, sementara mekanisme tuning membantu menentukan margin pemisahan paling efektif pada data hasil resampling.

3. Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis digunakan sebagai model pengklasifikasi dengan batas keputusan linear, dihasilkan dengan menyesuaikan kepadatan kondisional kelas dengan data dan menggunakan aturan Bayes. LDA dilatih menggunakan GridSearchCV dengan pilihan solver yakni svd dan lsqr. Pemilihan solver terbaik memberikan stabilitas komputasi dan akurasi maksimum, terutama setelah dataset diseimbangkan melalui teknik oversampling yang meningkatkan representasi kelas minoritas.

4. Artificial Neural Network (ANN)

Artificial Neural Network diterapkan sebagai model berbasis struktur jaringan saraf yang mampu mempelajari hubungan kompleks pada data. ANN dilatih dan dituning menggunakan GridSearchCV dengan beberapa konfigurasi hidden layer ((100,), (50, 25), (5, 2)), metode optimasi solver (adam dan lbfgs), serta nilai regulasi alpha (0.0001 dan 1e-5). Proses ini memastikan ANN memperoleh arsitektur yang paling sesuai untuk mempelajari pola klasifikasi hipertensi pada dataset yang telah di-oversample.

F. Evaluasi

Setelah proses pelatihan selesai, evaluasi kinerja dilakukan terhadap setiap model machine learning yang telah dilatih. Evaluasi menggunakan enam metrik utama, yaitu akurasi, precision, recall, F1-score, specificity, dan ROC AUC. Metrik tersebut dipilih karena memiliki kemampuan untuk memberikan gambaran yang komprehensif mengenai kinerja model dalam membedakan kelas target secara akurat, baik pada kelas positif maupun kelas negatif. Pengujian dilakukan pada untuk menilai sensitivitas dan konsistensi performa model terhadap perubahan jumlah tetangga. Hasil evaluasi dari seluruh kombinasi parameter Kdan teknik normalisasidisajikan pada Tabel 9 untuk dilakukan analisis perbandingan:

TABEL 9
HASIL EVALUASI

Teknik Oversampling	Label (Dalam Numerik)	Accuracy	Precision	Recall	F1-Score	F1-Macro Average	Specificity	ROC AUC
LR Original	0	0.9091	1.00	1.00	1.00	0.8143	1.0000	0.9930
	1		0.94	1.00	0.97		0.9630	
	2		0.71	0.71	0.71		0.9459	
	3		0.67	0.50	0.57		0.9750	
LR SMOTE	0	0.9545	1.00	1.00	1.00	0.9018	1.0000	0.9949
	1		1.00	1.00	1.00		1.0000	
	2		0.86	0.86	0.86		0.9730	
	3		0.75	0.75	0.75		0.9750	
LR ROS	0	0.9773	1.00	1.00	1.00	0.9530	1.0000	0.9930
	1		1.00	1.00	1.00		1.0000	

	2		1.00	0.86	0.92		1.0000	
			0.80	1.00	0.89		0.9750	
LR ADASYN	0	0.9545	1.00	1.00	1.00	0.9234	1.0000	0.9940
	1		0.94	1.00	0.97		0.9630	
	2		1.00	0.71	0.83		1.0000	
	3		0.80	1.00	0.89		0.9750	
SVM Original	0	0.8409	0.88	0.94	0.91	0.8260	0.9286	0.9538
	1		0.82	0.82	0.82		0.8889	
	2		0.71	0.71	0.71		0.9459	
	3		1.00	0.75	0.86		1.0000	
SVM SMOTE	0	0.8409	0.88	0.94	0.91	0.7693	0.9643	0.9776
	1		0.88	0.88	0.88		0.9286	
	2		0.71	0.71	0.71		0.959	
	3		0.67	0.50	0.57		0.9750	
SVM ROS	0	0.8182	0.88	0.88	0.88	0.7723	0.9286	0.9430
	1		0.79	0.88	0.83		0.8519	
	2		0.71	0.71	0.71		0.9459	
	3		1.00	0.50	0.67		1.0000	
SVM ADASYN	0	0.8182	0.88	0.88	0.88	0.7937	0.9286	0.9681
	1		0.75	0.88	0.81		0.8148	
	2		1.00	0.43	0.60		1.0000	
	3		0.80	1.00	0.89		0.9750	
LDA Original	0	0.9545	1.00	0.94	0.97	0.9588	1.0000	1.0000
	1		0.89	1.00	0.94		0.9259	
	2		1.00	0.86	0.92		1.0000	
	3		1.00	1.00	1.00		1.0000	
LDA SMOTE	0	0.9773	1.00	0.94	0.97	0.9848	1.0000	1.0000
	1		0.94	1.00	0.97		0.9630	
	2		1.00	1.00	1.00		1.0000	
	3		1.00	1.00	1.00		1.0000	
LDA ROS	0	0.8409	0.87	0.81	0.84	0.8486	0.9286	0.9786
	1		0.79	0.88	0.83		0.8519	
	2		1.00	0.71	0.83		1.0000	
	3		0.80	1.00	0.89		0.9750	
LDA ADASYN	0	0.8864	1.00	0.94	0.97	0.8217	1.0000	0.9930
	1		0.85	1.00	0.92		0.8889	
	2		1.00	0.43	0.60		1.0000	
	3		0.67	1.00	0.80		0.9500	
ANN Original	0	0.8864	0.88	0.94	0.91	0.8563	0.9286	0.9863
	1		0.94	0.88	0.91		0.9630	
	2		0.86	0.86	0.86		0.9730	

	3		0.75	0.75	0.75		0.9750	
ANN SMOTE	0	0.8864	0.88	0.94	0.91	0.8563	0.9286	0.9896
	1		0.94	0.88	0.91		0.9630	
	2		0.86	0.86	0.86		0.9730	
	3		0.75	0.75	0.75		0.9750	
ANN ROS	0	0.9091	1.00	0.94	0.97	0.8579	1.0000	0.9831
	1		0.89	1.00	0.74		0.9259	
	2		0.83	0.71	0.77		0.9730	
	3		0.75	0.75	0.75		0.9750	
ANN ADASYN	0	0.8864	0.88	0.94	0.91	0.8784	0.9286	0.9920
	1		0.88	0.88	0.88		0.9259	
	2		1.00	0.71	0.83		1.0000	
	3		0.80	1.00	0.89		0.9750	

1. Logistic Regression (LR)

Pada kondisi tanpa oversampling, LR mencapai akurasi 0.9091, F1-macro 0.8143, dan ROC AUC 0.9930. Performa ini menunjukkan bahwa model cukup baik untuk kelas mayoritas yaitu Normal dan Prehypertension, namun masih memiliki keterbatasan pada kelas minoritas seperti Stage 2 Hypertension dengan recall hanya 0.50. Situasi ini mengindikasikan bahwa ketidakseimbangan kelas menyebabkan model lebih condong pada pola mayoritas. Setelah SMOTE diterapkan, kinerja LR meningkat. Akurasi naik menjadi 0.9545, F1-macro meningkat ke 0.9018, dan ROC AUC juga bergerak naik ke 0.9949. Sampel sintetis hasil interpolasi membantu LR mengenali pola pada kelas minoritas tanpa mengganggu sifat linear model. Random Oversampling menghasilkan peningkatan terbesar bagi LR. Akurasi mencapai 0.9773, F1-macro naik menjadi 0.9530, dan ROC AUC tetap tinggi pada 0.9930. Duplikasi sampel minoritas memperkuat pola asli yang relevan bagi model linear sehingga sensitivitas pada Stage 2 meningkat drastis dengan recall mencapai 1.00. Pada kasus ini, ROS menjadi metode paling efektif bagi LR karena tidak memasukkan variasi sintetis yang berpotensi mengubah batas keputusan linear. ADASYN menghasilkan akurasi 0.9545, F1-macro 0.9234, dan ROC AUC 0.9940. Meskipun meningkatkan performa, hasilnya masih berada di bawah ROS. ADASYN membantu memperkaya area sulit pada kelas minoritas, tetapi struktur sintetisnya tidak sesederhana pola duplikasi ROS, sehingga penguatannya tidak setajam kombinasi LR dan ROS. Secara keseluruhan, semua teknik oversampling meningkatkan LR, dengan ROS sebagai konfigurasi paling optimal.

2. Support Vector Machine (SVM)

SVM tanpa oversampling menghasilkan akurasi 0.8409, F1-macro 0.8260, dan ROC AUC 0.9538. Model mampu menjaga keseimbangan antar kelas dan menunjukkan kemampuan diskriminatif yang cukup baik terhadap kelas minoritas dengan recall 0.75. SMOTE. Penerapan SMOTE mempertahankan akurasi di angka 0.8409 tetapi menurunkan

F1-macro menjadi 0.7693, sementara ROC AUC meningkat ke 0.9776. Kondisi ini menunjukkan bahwa SMOTE membantu memperluas pemisahan antar kelas secara global (ditunjukkan lewat kenaikan AUC), namun menciptakan ketidakseimbangan baru pada precision dan recall, sehingga F1-macro menurun. ROS. Dengan ROS, performa SVM mengalami penurunan. Akurasi turun ke 0.8182, F1-macro menjadi 0.7723, dan ROC AUC berkurang ke 0.9430. Duplikasi sampel minoritas memengaruhi struktur margin optimal SVM dan menurunkan kualitas generalisasi model. ADASYN. ADASYN menunjukkan hasil yang sedikit lebih baik dibanding ROS, terutama pada F1-macro (0.7937) dan ROC AUC (0.9681), meskipun akurasi tetap 0.8182. Tambahan sampel sintetis adaptif kadang membantu, tetapi tidak secara konsisten. Secara keseluruhan, performa SVM pada dataset ini cenderung paling stabil (robust) ketika digunakan tanpa resampling. Secara keseluruhan, SVM menunjukkan performa paling stabil ketika digunakan tanpa resampling. Stabilitas SVM tanpa resampling ini dijelaskan melalui prinsip margin maximization, di mana batas keputusan ditentukan oleh support vectors. Jadi, penambahan sampel sintetis di luar area margin tidak mengubah posisi hyperplane secara signifikan, sehingga performa tetap konsisten. Penggunaan kernel RBF dengan parameter gamma memperkuat stabilitas ini dengan memetakan pola non-linear pada area kemiripan yang terlokalisasi di sekitar support vectors. Didukung oleh karakteristik fitur klinis seperi tekanan darah dan usia yang sudah informatif sejak awal, dengan ini dapat disimpulkan bahwa SVM mampu mencapai konfigurasi pemisahan optimal tanpa memerlukan penambahan volume data sintetis.

3. Linear Discriminant Analysis (LDA)

Pada kondisi tanpa oversampling, LDA menunjukkan performa tinggi dengan akurasi 0.9545, F1-macro 0.9588, dan ROC AUC 1.0000 menunjukkan bahwa model mampu memisahkan kelas dengan sangat baik berdasarkan asumsi Gaussian linear. Sensitivitas terhadap kelas kritis juga sudah

optimal. SMOTE. Penerapan SMOTE menghasilkan performa terbaik dari seluruh eksperimen. Akurasi meningkat menjadi 0.9773, F1-macro mencapai 0.9848, dan ROC AUC tetap sempurna di angka 1.0000. Interpolasi SMOTE selaras dengan asumsi linear LDA sehingga menghasilkan estimasi kovarian yang lebih stabil dan batas pemisahan yang lebih presisi. Kombinasi LDA dan SMOTE menjadi konfigurasi terbaik dari seluruh model, hal ini dikarenakan karakteristik SMOTE yang melakukan interpolasi linier antar sampel kelas minoritas. Proses ini menghasilkan distribusi data sintetis yang cenderung mempertahankan struktur kovarians pada kelas asli. Mengingat LDA sangat bergantung pada asumsi distribusi Gaussian dan pemisahan linier, data sintetis yang dihasilkan SMOTE membantu LDA dalam menentukan decision boundary yang lebih stabil dan akurat dibandingkan ADASYN yang cenderung menciptakan data di area borderline yang mungkin menyebabkan overlap dan dianggap sebagai noise oleh model linier seperti LDA. Sementara itu, ROS menyebabkan penurunan performa signifikan. Akurasi turun ke 0.8409, F1-macro menjadi 0.8486, dan ROC AUC berkurang ke 0.9786. Duplikasi sampel minoritas mengubah struktur distribusi dan matriks kovarian, sehingga merusak asumsi dasar LDA. ADASYN. ADASYN memberikan performa lebih baik dibanding ROS tetapi tetap di bawah SMOTE. Akurasi berada pada 0.8864, F1-macro turun ke 0.8217, dan ROC AUC menjadi 0.9930. Karena ADASYN menghasilkan sampel yang lebih beragam pada area sulit, distribusi kelas menjadi kurang sesuai dengan asumsi Gaussian yang diperlukan LDA. Secara ringkas, SMOTE adalah teknik paling cocok bagi LDA, sementara ROS dan ADASYN kurang sesuai secara statistik.

4. Artificial Neural Network (ANN)

Pada kondisi ANN tanpa oversampling. Model menghasilkan akurasi 0.8864, F1-macro 0.8563, dan ROC AUC 0.9863 pada kondisi awal serta mampu menangkap pola non-linear, namun di sisi lain masih kesulitan dalam mengenali kelas minoritas dengan recall hanya 0.50 pada Stage 2 hypertension. Setelah SMOTE diterapkan, akurasi sedikit turun ke 0.8864, namun F1-macro meningkat menjadi 0.8563 dan ROC AUC naik ke 0.9896. Hal ini menunjukkan bahwa SMOTE dapat membantu dalam menyeimbangkan pembelajaran antar kelas walaupun ada sedikit penurunan akurasi keseluruhan. Dengan ROS, akurasi kembali ke 0.9091, F1-macro mencapai 0.8579, dan ROC AUC berada pada 0.9831. ANN cukup toleran terhadap duplikasi sehingga pola minoritas dapat diperkuat tanpa mengorbankan performa pada kelas mayoritas. ADASYN menghasilkan F1-macro tertinggi untuk ANN, yaitu 0.8784, dengan ROC AUC 0.9920 dan akurasi 0.8864. Sampel adaptif dari ADASYN menambah variasi pada area yang sulit sehingga jaringan mampu mempelajari representasi kelas minoritas dengan lebih baik. Secara umum ANN paling diuntungkan oleh teknik oversampling sintetis adaptif seperti ADASYN.

5. Analisis Signifikansi dan Generalisasi Model

Untuk memastikan keandalan hasil, dilakukan uji signifikansi statistik menggunakan Uji Wilcoxon Signed-

Rank terhadap hasil 5-fold cross-validation. Hasil uji menunjukkan nilai $p < 0,05$ untuk kombinasi metode unggulan seperti LDA-SMOTE dan ANN-ADASYN, yang mengonfirmasi bahwa peningkatan performa tersebut signifikan secara statistik dan bukan merupakan variasi acak. Selanjutnya, potensi overfitting akibat teknik oversampling dimitigasi melalui optimasi hyperparameter GridSearchCV dan penerapan regularisasi (parameter C dan alpha). Stabilitas model pada data testing tercermin dari konsistensi skor performa di seluruh fold cross-validation, yang menunjukkan bahwa model mampu mempelajari pola distribusi kelas hipertensi secara robust dan mempertahankan kemampuan generalisasi tanpa terjebak pada penghafalan sampel sintetis.

Hasil penelitian ini berimplikasi langsung pada pengembangan teknologi deteksi dini hipertensi otomatis. Pada model unggulan seperti LDA-SMOTE memiliki potensi besar untuk diintegrasikan ke dalam Clinical Decision Support System (CDSS) guna mendukung proses skrining yang lebih objektif dan efisien, terutama pada fasilitas kesehatan dengan keterbatasan sumber daya. Kemampuan model dalam mengidentifikasi pasien Stage 2 hypertension dengan akurat sangat penting untuk meminimalkan risiko false negative, sehingga penanganan medis dapat dilakukan lebih awal guna mencegah komplikasi kardiovaskular yang lebih berat. Dengan demikian, model ini berpotensi menjadi alat bantu yang efektif dalam mendukung pengambilan keputusan klinis yang lebih tepat dan berorientasi pada pencegahan.

IV. KESIMPULAN

Hasil evaluasi menunjukkan bahwa teknik oversampling memberikan pengaruh yang berbeda pada setiap algoritma, sehingga pemilihan metode resampling tidak dapat digeneralisasi dan harus disesuaikan dengan karakteristik matematis dari masing-masing model. Secara keseluruhan, kombinasi Linear Discriminant Analysis (LDA) dengan SMOTE merupakan konfigurasi paling unggul dalam penelitian ini. Pasangan ini menghasilkan akurasi tertinggi sebesar 0.9773, F1-macro mencapai 0.9848. Hal ini menjadikannya sebagai model pemenang dalam penelitian ini. Keberhasilan SMOTE pada LDA disebabkan oleh sifat interpolasi yang selaras dengan asumsi Gaussian linear yang menjadi dasar LDA, sehingga meningkatkan stabilitas pemodelan tanpa mengganggu struktur kovarian. Untuk algoritma lain, performa terbaik ditemukan pada konfigurasi yang berbeda. Logistic Regression (LR) menunjukkan peningkatan paling besar ketika dikombinasikan dengan Random Oversampling (ROS), yang menghasilkan akurasi 0.9773, F1-Macro 0.9530, serta peningkatan sensitivitas yang signifikan pada kelas minoritas. Hal ini membuatnya menjadi konfigurasi terbaik untuk model linear. Berbeda dengan model linear, SVM cenderung tidak responsif terhadap seluruh teknik oversampling karena margin optimalnya sensitif terhadap sampel sintetis pada dataset berukuran kecil, sehingga perubahan performanya relatif minimal. ANN menunjukkan pola yang lebih dinamis dibandingkan algoritma lainnya. Pada kondisi tanpa resampling, ANN

menghasilkan akurasi 0.8864, namun setelah diterapkan ROS akurasinya meningkat menjadi 0.9091. Selain itu, teknik oversampling adaptif seperti ADASYN tetap memberikan keuntungan terbesar secara keseluruhan, karena mampu menambah variasi sampel pada area yang sulit dan meningkatkan kemampuan jaringan dalam mengenali kelas minoritas. Secara keseluruhan, teknik oversampling terbukti efektif dalam menangani ketidakseimbangan kelas, namun tingkat efektivitasnya bergantung pada karakteristik algoritma. SMOTE paling sesuai untuk model berbasis distribusi seperti LDA, ROS optimal untuk model linear seperti Logistic Regression, sementara teknik adaptif seperti ADASYN menawarkan performa terbaik untuk model non-linear seperti ANN. Temuan ini memiliki implikasi klinis penting, di mana model unggulan dapat diintegrasikan ke dalam sistem pendukung keputusan medis seperti Clinical Decision Support System untuk membantu deteksi dini hipertensi secara lebih presisi. Stabilitas seluruh model ini tetap terjaga melalui penerapan regularisasi dan GridSearchCV yang secara efektif memitigasi risiko overfitting. Validitas peningkatan performa ini juga telah dibuktikan melalui uji statistik Wilcoxon Signed-Rank yang menunjukkan hasil signifikan dengan nilai $p < 0,05$. Dengan demikian, konfigurasi SMOTE-LDA muncul sebagai kombinasi paling unggul dalam prediksi multilevel hipertensi pada penelitian ini.

DAFTAR PUSTAKA

- [1] World Health Organization, "Hypertension," Apr. 2024. [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/hypertension>. [Accessed: 08-Feb-2025].
- [2] L. R. Putri, M. Azam, A. A. Nisa, A. I. Fibriana, P. Kanthawee, and S. A. Shabbir, "Prevalence and Risk Factors of Hypertension among Young Adults: An Indonesian Basic Health Survey," *The Open Public Health Journal*, vol. 18, no. 1, Jan. 2025, doi: <https://doi.org/10.2174/0118749445361291241129094132>.
- [3] Giuseppe Mancia et al., "2023 ESH Guidelines for the management of arterial hypertension The Task Force for the management of arterial hypertension of the European Society of Hypertension Endorsed by the European Renal Association (ERA) and the International Society of Hypertension (ISH)," *Journal of Hypertension*, vol. Publish Ahead of Print, no. 12, Jun. 2023, doi: <https://doi.org/10.1097/hjh.0000000000003480>.
- [4] R. Kurniawan et al., "Hypertension prediction using algorithm among Indonesian adults," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 12, no. 2, 2023, doi: <https://doi.org/10.11591/ijai.v12.i2.pp776-784>.
- [5] S. Lip et al., "BASED MODELS FOR PREDICTING WHITE-COAT AND MASKED PATTERNS OF BLOOD PRESSURE," *Journal of Hypertension*, vol. 39, no. Supplement 1, p. e69, Apr. 2021, doi: <https://doi.org/10.1097/01.hjh.0000745092.07595.a5>.
- [6] S. M. S. Islam et al., "Approaches for Predicting Hypertension and Its Associated Factors Using Population-Level Data From Three South Asian Countries," *Frontiers in Cardiovascular Medicine*, vol. 9, Mar. 2022, doi: <https://doi.org/10.3389/fcvm.2022.839379>.
- [7] N. Novianti, S. P. A. Alkadri, and I. Fakhruzi, "Klasifikasi Penyakit Hipertensi Menggunakan Metode Random Forest," *Progresif: Jurnal Ilmiah Komputer*, vol. 20, no. 1, pp. 165-174, Feb. 2024, doi: [10.35889/progresif.v20i1.1663](https://doi.org/10.35889/progresif.v20i1.1663).
- [8] J. M. Kurniawan, I. Bagus, S. Yong, and Mediana Aryuni, "Handling Imbalanced Dataset in Online Learning Performance Prediction Using Resampling Techniques," 2024 7th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), pp. 900–906, Dec. 2024, doi: <https://doi.org/10.1109/isriti64779.2024.10963560>.
- [9] U. Hasanah, A. M. Soleh, and K. Sadik, "Effect of Random Under sampling, Oversampling, and SMOTE on the Performance of Cardiovascular Disease Prediction Models," *Jurnal Matematika, Statistika dan Komputasi*, vol. 21, no. 1, pp. 88–102, Sep. 2024, doi: <https://doi.org/10.20956/j.v21i1.35552>.
- [10] M. Khushi et al., "A Comparative Performance Analysis of Data Resampling Methods on Imbalance Medical Data," *IEEE Access*, vol. 9, pp. 109960–109975, 2021, doi: <https://doi.org/10.1109/access.2021.3102399>.
- [11] A. J. Albert, R. Murugan, and T. Sripriva, "Diagnosis of heart disease using oversampling methods and decision tree classifier in cardiology," *Research on Biomedical Engineering*, Dec. 2022, doi: <https://doi.org/10.1007/s42600-022-00253-9>.
- [12] I. Dey and V. Pratap, "A Comparative Study of SMOTE, Borderline-SMOTE, and ADASYN Oversampling Techniques using Different Classifiers," 2023 3rd International Conference on Smart Data Intelligence (ICSDMI), Mar. 2023, doi: <https://doi.org/10.1109/icsmdi57622.2023.00060>.
- [13] F. Handayani, "Komparasi Support Vector Machine, Logistic Regression Dan Artificial Neural Network Dalam Prediksi Penyakit Jantung," *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, vol. 7, no. 3, p. 329, Dec. 2021, doi: <https://doi.org/10.26418/jp.v7i3.48053>.
- [14] None Dewi Nasien et al., "Perbandingan Implementasi Menggunakan Metode KNN, Naive Bayes, dan Logistik Regression Untuk Mengklasifikasi Penyakit Diabetes," *JEKIN - Jurnal Teknik Informatika*, vol. 4, no. 1, pp. 10–17, Feb. 2024, doi: <https://doi.org/10.58794/jekin.v4i1.640>.
- [15] A. Jibril, K. Haruna, and Z. Jiangsheng, "Feature Selection and Parameter Optimization of Support Vector Machine (Svm) and Logistic Regression (Lr) Algorithms Using Particle Swarm Optimization (Pso) In Prediction of Diabetes," *Journal of Computer Science and Information Technology*, vol. 11, no. 1, pp. 21–47, Jun. 2023, doi: <https://doi.org/10.15640/jcsit.v11n1a3>.
- [16] S. Vamshi Kumar, T. V. Rajinikanth, and S. Viswanadha Raju, "Heart Attack Classification Using SVM with LDA and PCA Linear Transformation Techniques," *Algorithms for Intelligent Systems*, pp. 99–112, 2021, doi: https://doi.org/10.1007/978-981-33-4046-6_10.
- [17] O. Samarina, A. Chuykov, and O. Kovtun, "PREDICTION OF EARLY ONSET OF ARTERIAL HYPERTENSION BY DISCRIMINANT ANALYSIS IN CHILDREN," *Journal of Hypertension*, vol. 39, no. Supplement 1, p. e188, Apr. 2021, doi: <https://doi.org/10.1097/01.hjh.0000746532.50623.d4>.
- [18] S. Ebrahimi and H. Farshid, "The efficiency of artificial neural network (ANN) for diagnosis of obesity and hypertension," *Academic Journal of Health Sciences: Medicina Balear* 202, pp. 47–51, 2023.
- [19] R. Muzayyah, D. A. A. Pertwi, M. Ali, and M. A. Muslim, "Comparison of gridsearchcv and bayesian hyperparameter optimization in random forest algorithm for diabetes prediction," *Journal of Soft Computing Exploration*, vol. 5, no. 1, pp. 86–91, Apr. 2024, doi: <https://doi.org/10.52465/josce.v5i1.308>.
- [20] R. Waluyo and A. S. Munir, "Optimasi Prediksi Kematian pada Gagal Jantung Analisis Perbandingan Algoritma Pembelajaran Ensemble dan Teknik Penyeimbangan Data pada Dataset," *Jurnal Sistem dan Teknologi Informasi (JustIN)*, vol. 12, no. 2, p. 365, Apr. 2024, doi: <https://doi.org/10.26418/justin.v12i2.75158>.
- [21] M. Lokanan, "Exploring Resampling Techniques in Credit Card Default Prediction," *Research Square (Research Square)*, Mar. 2024, doi: <https://doi.org/10.21203/rs.3.rs-4087259/v1>.
- [22] G. M. Foody, "Challenges in the real world use of classification accuracy metrics: From recall and precision to the Matthews correlation coefficient," *PLOS ONE*, vol. 18, no. 10, pp. e0291908–e0291908, Oct. 2023, doi: <https://doi.org/10.1371/journal.pone.0291908>.
- [23] W. Cullerne Bown, "Sensitivity and Specificity versus Precision and Recall, and Related Dilemmas," *Journal of Classification*, vol. 41, no. 2, pp. 402–426, Jun. 2024, doi: <https://doi.org/10.1007/s00357-024-09478-y>.

- [24] A. M. Carrington et al., "Deep ROC Analysis and AUC as Balanced Average Accuracy to Improve Model Selection, Understanding and Interpretation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 1–1, 2022, doi: <https://doi.org/10.1109/TPAMI.2022.3145392>.
- [25] Y. Liang, Z. Chen, G. Liu, and M. Elgendi, "A new, short-recorded photoplethysmogram dataset for blood pressure monitoring in China," *Scientific Data*, vol. 5, no. 1, p. 180020, Feb. 2018, doi: <https://doi.org/10.1038/sdata.2018.20>.
- [26] P. Purwono, P. Dewi, S. K. Wibisono, and B. P. Dewa, "MODEL PREDIKSI OTOMATIS JENIS PENYAKIT HIPERTENSI DENGAN PEMANFAATAN ALGORITMA ARTIFICIAL NEURAL NETWORK", *insect*, vol. 7, no. 2, pp. 82–90, Mar. 2022..
- [27] E. Alshdaifat, D. Alshdaifat, A. Alsarhan, F. Hussein, and S. M. F. S. El-Salhi, "The Effect of Preprocessing Techniques, Applied to Numeric Features, on Classification Algorithms' Performance," *Data*, vol. 6, no. 2, p. 11, Jan. 2021, doi: <https://doi.org/10.3390/data6020011>.
- [28] R. Hidayat, "Implementasi Algoritma Random Forest Regression Untuk Memprediksi Penjualan Produk di Supermarket", *simkom*, vol. 10, no. 1, pp. 101-109, Jan. 2025, doi: <https://doi.org/10.51717/simkom.v10i1.703>.
- [29] Mochamad Gilang Saputra and B. J. Santoso, "Implementasi Feature Selection Menggunakan Boruta untuk Peningkatan Akurasi Model Lapser Prediction," *MALCOM Indonesian Journal of and Computer Science*, vol. 5, no. 3, pp. 886–895, Jun. 2025, doi: <https://doi.org/10.57152/malcom.v5i3.1992>.
- [30] Y. Yin, Z. Yuan, and I. M. Tanvir, "Electronic medical records imputation by temporal Generative Adversarial Network," *BioData Mining*, vol. 17, p. 19, 2024, doi: 10.1186/s13040-024-00372-2.
- [31] T. T. Wu, L. H. Smith, L. M. Vernooy, E. Patel, and J. W. Devlin, "Data Missingness Reporting and Use of Methods to Address It in Critical Care Cohort Studies," *Critical Care Explorations*, vol. 5, no. 11, p. e1005, Nov. 2023, doi: <https://doi.org/10.1097/CCE.00000000000001005>.
- [32] O. Alotaibi, E. Pardede, dan S. Tomy, "Cleaning Big Data Streams: A Systematic Literature Review," **Technologies**, vol. 11, no. 4, p. 101, Jul. 2023, doi: [10.3390/technologies11040101](https://doi.org/10.3390/technologies11040101).
- [33] D. Dai, Z. Ji, and H. Wang, "Non-Invasive Continuous Blood Pressure Estimation From Single-Channel PPG Based on a Temporal Convolutional Network Integrated With an Attention Mechanism," *Appl. Sci.*, vol. 14, no. 14, Art. 6061, 2024, doi: [10.3390/app14146061](https://doi.org/10.3390/app14146061).
- [34] L. A. Demidova, "Two-Stage Hybrid Data Classifiers Based on SVM and kNN Algorithms," *Symmetry*, vol. 13, no. 4, p. 615, Apr. 2021, doi: <https://doi.org/10.3390/sym13040615>.
- [35] Y. Dimas Pratama and A. Salam, "Comparison of Data Normalization Techniques on KNN Classification Performance for Pima Indians Diabetes Dataset," *Journal of Applied Informatics and Computing*, vol. 9, no. 3, pp. 693–706, Jun. 2025, doi: <https://doi.org/10.30871/jaic.v9i3.9353>.
- [36] M. R. Firmansyah, "Stroke Classification Comparison with KNN through Standardization and Normalization Techniques," *Advance Sustainable Science, Engineering and Technology*, vol. 6, no. 1, p. 02401012, Jan. 2024, doi: <https://doi.org/10.26877/asset.v6i1.17685>.
- [37] Vera, J. Aranda, Ricardo, R. Valle, and J. Luis, "Imbalanced data preprocessing techniques for : a systematic mapping study," vol. 65, no. 1, pp. 31–57, Nov. 2022, doi: <https://doi.org/10.1007/s10115-022-01772-8>.
- [38] Y. Yang, Hadi Akbarzadeh Khorshidi, and Uwe Aickelin, "A review on over-sampling techniques in classification of multi-class imbalanced datasets: insights for medical problems," *Frontiers in Digital Health*, vol. 6, Jul. 2024, doi: <https://doi.org/10.3389/fdgh.2024.1430245>.
- [39] D. Elreedy, A. F. Atiya, and F. Kamalov, "A theoretical distribution analysis of synthetic minority oversampling technique (SMOTE) for imbalanced learning," , vol. 113, Jan. 2023, doi: <https://doi.org/10.1007/s10994-022-06296-4>.
- [40] M. Salmi, D. Atif, D. Oliva, A. Abraham, and S. Ventura, "Handling imbalanced medical datasets: review of a decade of research," *Artificial Intelligence Review*, vol. 57, no. 10, Sep. 2024, doi: <https://doi.org/10.1007/s10462-024-10884-2>.
- [41] C. G. Tekkali and K. Natarajan, "An advancement in AdaSyn for imbalanced learning: An application to fraud detection in digital transactions," *Journal of Intelligent & Fuzzy Systems*, vol. 46, no. 5–6, pp. 11381–11396, Oct. 2024, doi: <https://doi.org/10.3233/jifs-236392>.
- [42] Z. Maisat Eka Darmawan and A. Fauzan Dianta, "Implementasi Optimasi Hyperparameter GridSearchCV Pada Sistem Prediksi Serangan Jantung Menggunakan SVM," *Teknologi*, vol. 13, no. 1, pp. 8–15, Jan. 2023, doi: <https://doi.org/10.26594/teknologi.v13i1.3098>.
- [43] S. Mulyani and T. Arifin, "Prediksi Kelangsungan Hidup Pasien Gagal Jantung Menggunakan Pendekatan dengan Optimasi GridSearchCV," *Jurnal Informatika Polinema*, vol. 11, no. 4, pp. 577–586, Aug. 2025, doi: <https://doi.org/10.33795/jip.v11i4.7938>.
- [44] Masdar Desiawan and Achmad Solichin, "SVM Optimization with Grid Search Cross Validation for Improving Accuracy of Schizophrenia Classification Based on EEG Signal," *JURNAL TEKNIK INFORMATIKA*, vol. 17, no. 1, pp. 10–20, May 2024, doi: <https://doi.org/10.15408/jti.v17i1.37422>.
- [45] Y. Xi, H. Wang, and N. Sun, "Machine learning outperforms traditional logistic regression and offers new possibilities for cardiovascular risk prediction: A study involving 143,043 Chinese patients with hypertension," *Frontiers in Cardiovascular Medicine*, vol. 9, Nov. 2022, doi: <https://doi.org/10.3389/fcvm.2022.1025705>.
- [46] N. Nuryani, T. Utomo, N. Wiyono, A. D. Sutomo, and S. Ling, "Cuffless Hypertension Detection using Swarm Support Vector Machine Utilizing Photoplethysmogram and Electrocardiogram," *Journal of Biomedical Physics and Engineering*, vol. 13, no. 5, 2023, doi: <https://doi.org/10.31661/jbpe.v0i0.2206-1504>.
- [47] R. Rizal Isnanto, I. Rashad, and Catur Edi Widodo, "Classification of Heart Disease Using Linear Discriminant Analysis Algorithm," *E3S Web of Conferences*, vol. 448, pp. 02053–02053, Jan. 2023, doi: <https://doi.org/10.1051/e3sconf/202344802053>.
- [48] M. M. Bukhari, B. F. Alkhamees, S. Hussain, A. Gumaei, A. Assiri, and S. S. Ullah, "An Improved Artificial Neural Network Model for Effective Diabetes Prediction," *Complexity*, vol. 2021, pp. 1–10, Apr. 2021, doi: <https://doi.org/10.1155/2021/5525271>.
- [49] E. Martinez-Ríos, L. Montesinos, M. Alfaro-Ponce, and L. Pecchia, "A review of in hypertension detection and blood pressure estimation based on clinical and physiological data," *Biomedical Signal Processing and Control*, vol. 68, p. 102813, Jul. 2021, doi: <https://doi.org/10.1016/j.bspc.2021.102813>.
- [50] F. López-Martínez, E. R. Núñez-Valdez, R. G. Crespo, and V. García-Díaz, "An artificial neural network approach for predicting hypertension using NHANES data," *Scientific Reports*, vol. 10, no. 1, Jun. 2020, doi: <https://doi.org/10.1038/s41598-020-67640-z>.