Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 4, Juni 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i4.9711

Evaluasi Komparatif Random Forest, XGBoost, dan Logistic Regression untuk Prediksi Stroke Menggunakan Teknik SMOTE

Anggita Nur Holifah[1], Imam Tahyudin[2,*]

[1]Sistem Informasi, Fakultas Ilmu Komputer, Universitas Amikom Purwokerto, Purwokerto, Indonesia
[2] Magister Ilmu Komputer, Fakultas Ilmu Komputer, Universitas Amikom Purwokerto, Purwokerto, Indonesia
Email: [1]anggitanur32@gmail.com, [2,*]imam.tahyudin@amikompurwokerto.ac.id
Email Penulis Korespondensi: imam.tahyudin@amikompurwokerto.ac.id[*]
Submitted: **19/04/2026**; Accepted: **05/06/2026**; Published: **30/06/2026**

Abstrak– Stroke merupakan salah satu penyakit dengan tingkat kematian dan kecacatan yang tinggi sehingga diperlukan metode prediksi yang mampu mendukung deteksi dini secara lebih akurat. Penelitian ini bertujuan untuk menganalisis dan membandingkan performa algoritma Random Forest, XGBoost, dan Logistic Regression dalam memprediksi risiko stroke berdasarkan data kesehatan pasien. Dataset yang digunakan terdiri dari 5.110 data dengan 12 variabel, seperti usia, riwayat hipertensi, penyakit jantung, kadar glukosa darah, indeks massa tubuh, dan kebiasaan merokok. Ketidakseimbangan data ditangani menggunakan metode SMOTE, sedangkan hyperparameter tuning diterapkan pada model Random Forest untuk mengevaluasi pengaruhnya terhadap performa klasifikasi. Evaluasi model dilakukan menggunakan metrik accuracy, precision, recall, dan F1-score. Hasil penelitian menunjukkan bahwa berdasarkan nilai weighted average, Random Forest memperoleh performa keseluruhan terbaik dengan accuracy sebesar 94%, precision 90%, recall 94%, dan F1-score 91%. XGBoost menghasilkan performa yang hampir setara dengan accuracy 93%, precision 90%, recall 93%, dan F1-score 91%, sedangkan Logistic Regression memperoleh accuracy 74%, precision 93%, recall 74%, dan F1-score 81%. Hasil evaluasi juga menunjukkan bahwa hyperparameter tuning pada Random Forest tidak memberikan peningkatan performa yang signifikan dibandingkan model sebelum tuning. Meskipun Random Forest dan XGBoost menghasilkan performa keseluruhan yang lebih tinggi, Logistic Regression menunjukkan kemampuan yang lebih baik dalam mendeteksi kasus stroke berdasarkan nilai recall pada kelas stroke. Oleh karena itu, pemilihan model perlu disesuaikan dengan tujuan penggunaan, baik untuk memperoleh performa klasifikasi secara keseluruhan maupun untuk memaksimalkan deteksi kasus stroke.

Kata Kunci: SMOTE; Stroke; Machine Learning; Logistic Regression; Random Forest; XGBoost.

Abstract– Stroke is one of the leading causes of death and disability, making accurate prediction methods essential for supporting early detection. This study aims to analyze and compare the performance of Random Forest, XGBoost, and Logistic Regression algorithms in predicting stroke risk based on patient health data. The dataset consisted of 5,110 records with 12 variables, including age, hypertension history, heart disease, blood glucose level, body mass index, and smoking status. Data imbalance was addressed using the Synthetic Minority Oversampling Technique (SMOTE), while hyperparameter tuning was applied to the Random Forest model to evaluate its impact on classification performance. Model evaluation was conducted using accuracy, precision, recall, and F1-score. The results showed that, based on weighted average values, Random Forest achieved the best overall performance with 94% accuracy, 90% precision, 94% recall, and 91% F1-score. XGBoost delivered comparable performance with 93% accuracy, 90% precision, 93% recall, and 91% F1-score, while Logistic Regression achieved 74% accuracy, 93% precision, 74% recall, and 81% F1-score. The findings also indicate that hyperparameter tuning did not significantly improve the performance of the Random Forest model. Although Random Forest and XGBoost demonstrated better overall classification performance, Logistic Regression showed superior capability in detecting stroke cases based on the recall value of the stroke class. Therefore, model selection should be aligned with the intended objective, whether to achieve overall classification performance or to maximize stroke case detection.

Keywords: SMOTE; Stroke; Machine Learning; Logistic Regression; Random Forest; XGBoost.

1. PENDAHULUAN

Stroke merupakan salah satu penyebab kematian dan kecacatan terbesar di dunia. Diagnosis stroke memerlukan identifikasi fitur klinis serta pencitraan otak untuk membedakan stroke iskemik dari perdarahan *intraserebral*. Stroke, yang juga dikenal sebagai kecelakaan *serebrovaskular*, menyebabkan kerusakan pada sistem saraf pusat akibat gangguan *vaskular* dan menjadi salah satu penyebab kecacatan utama secara global [1]. Menurut penelitian [2], stroke merupakan salah satu penyebab utama kematian dan kecacatan dengan faktor risiko utama seperti hipertensi, diabetes, dan stroke. Deteksi dini serta prediksi risiko stroke sangatlah krusial untuk meningkatkan prospek pemulihan pasien dan meminimalkan komplikasi jangka panjang. Menurut Organisasi Kesehatan Dunia (WHO), penyakit *kardiovaskular* merupakan penyebab utama kematian di seluruh dunia dengan jumlah kematian mencapai sekitar 17,9 juta jiwa setiap tahunnya. Penyakit ini mencakup penyakit jantung koroner dan stroke, di mana sebagian besar kasus kematian disebabkan oleh serangan jantung dan stroke [3]. Oleh karena itu, deteksi dini serta penilaian risiko yang akurat, khususnya terhadap stroke, menjadi sangat penting untuk meningkatkan peluang kesembuhan pasien dan mencegah terjadinya komplikasi yang lebih serius.

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 4, Juni 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i4.9711

Menurut laporan tenaga kesehatan di Indonesia, prevalensi stroke tercatat 7 per 1.000 penduduk, dengan gejala stroke mencapai 12,1 per 1.000 penduduk. Berdasarkan hasil Riset Kesehatan Dasar (Riskesdas), angka prevalensi stroke di Provinsi Sulawesi Utara adalah 10,8%, sementara di D.I. Yogyakarta sebesar 10,3%. DKI Jakarta dan Bangka Belitung masing-masing memiliki prevalensi stroke 9,7 per mil. Prevalensi gejala stroke tertinggi tercatat di Sulawesi Selatan dengan 17,9%, diikuti D.I. Yogyakarta 16,9%, serta Sulawesi Tengah 16,6%. Di Jawa Timur, prevalensi stroke dilaporkan 16 per mil, sedangkan di Kalimantan Selatan, kasus stroke yang didiagnosis tenaga kesehatan mencapai 9,2%; jika ditambah gejala stroke, angka tersebut naik menjadi 14,5%[4].

Teknologi *machine learning* telah berkembang pesat dalam beberapa tahun belakangan, dengan penerapan yang semakin meluas di sektor medis, khususnya untuk deteksi dan prediksi penyakit [5]. *Exploratory Data Analysis* (EDA) menjadi langkah awal yang penting untuk memahami karakteristik data dan mempersiapkan data bagi pengembangan model prediktif yang andal. Melalui EDA, kita dapat melihat pola dan hubungan antar variabel kunci, mengidentifikasi data yang hilang serta *outlier*, dan melakukan rekayasa fitur untuk meningkatkan kinerja model prediksi [6]. Teknologi *machine learning* dapat dimanfaatkan untuk mendeteksi berbagai penyakit. Namun, pada proses penerapannya, sering muncul masalah ketidakseimbangan kelas dalam dataset yang dipakai. Ketidakseimbangan tersebut timbul ketika jumlah sampel pada kelas mayoritas jauh melebihi kelas minoritas, sehingga berdampak pada akurasi hasil serta performa model secara keseluruhan [7].

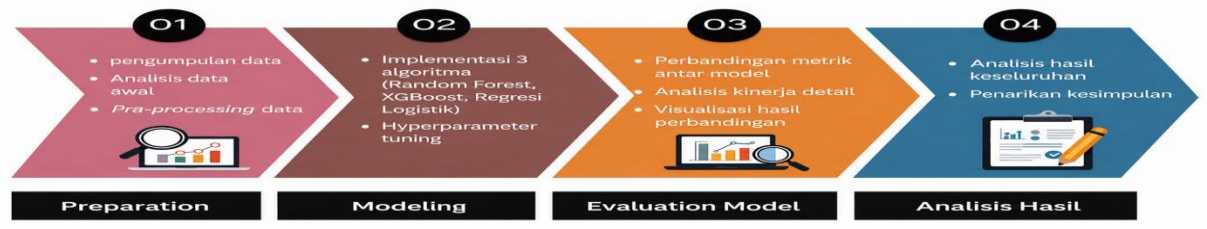
Pada aplikasi *machine learning* ini, pendekatan prediksi stroke menggunakan *Random Forest*, *XGBoost*, dan *Logistic Regression* untuk mengukur akurasi dan sensitivitas model. *Random Forest* menghasilkan prediksi stabil dengan mengurangi *overfitting* melalui banyak pohon keputusan, sedangkan *XGBoost* meningkatkan akurasi dengan pendekatan *boosting* yang cepat dan adaptif terhadap fitur data. Sebagai *baseline*, *logistic regression* digunakan untuk memprediksi probabilitas stroke melalui hubungan linear antar variabel, menyediakan interpretasi yang sederhana. Kinerja ketiga model ini dievaluasi untuk menentukan metode terbaik dalam prediksi risiko stroke [8].

Beberapa penelitian terdahulu telah menerapkan algoritma *machine learning* untuk prediksi stroke. Penelitian oleh Banjar dkk. menggunakan algoritma Random Forest dengan penerapan SMOTE pada *dataset* stroke sebanyak 5.110 data dan memperoleh akurasi terbaik sebesar 86,82% menggunakan 100 *trees* [9]. Penelitian lain membandingkan beberapa algoritma *machine learning* untuk memprediksi stroke, yakni *Decision Tree*, *Naive Bayes*, dan *Random Forest* dalam klasifikasi stroke menggunakan RapidMiner Temuan dari studi tersebut mengindikasikan bahwa *Decision Tree* mencapai akurasi 95,13%, yang merupakan nilai tertinggi di antara algoritma lain, sehingga dianggap paling optimal untuk klasifikasi risiko stroke pada dataset terkait [10]. Beberapa studi tersebut telah memanfaatkan metode *ensemble learning* dan *boosting* untuk meningkatkan performa prediksi stroke dengan memanfaatkan berbagai atribut kesehatan pasien. Namun, sebagian besar penelitian sebelumnya masih berfokus pada penggunaan satu algoritma tertentu atau hanya membandingkan model klasifikasi konvensional tanpa melakukan evaluasi secara komprehensif terhadap model *ensemble* modern seperti XGBoost. Selain itu, beberapa penelitian hanya menitikberatkan pada nilai akurasi tanpa memperhatikan metrik evaluasi lain seperti *precision*, *recall*, dan *F1-score* yang penting dalam kasus *imbalanced dataset*. Penanganan ketidakseimbangan data juga masih menjadi tantangan karena dapat memengaruhi kemampuan model dalam mengidentifikasi pasien stroke secara tepat [9], [10].

Penelitian ini tidak hanya membandingkan performa model *Random Forest*, *XGBoost*, dan *Logistic Regression*, tetapi juga mengevaluasi pengaruh penerapan SMOTE dan *hyperparameter tuning* terhadap peningkatan performa klasifikasi stroke pada dataset yang tidak seimbang. Oleh karena itu, penelitian ini bertujuan untuk menganalisis dan membandingkan performa algoritma *Random Forest*, *XGBoost*, dan *Logistic Regression* dalam prediksi stroke menggunakan metode SMOTE untuk mengatasi ketidakseimbangan data. Proses evaluasi model dilakukan menggunakan metrik akurasi, *precision*, *recall*, dan *F1-score* guna menentukan model yang memiliki performa terbaik dalam klasifikasi risiko stroke.

2. METODOLOGI PENELITIAN

Penelitian ini dilakukan melalui empat tahapan utama, yaitu *preparation*, *modeling*, *model evaluation*, dan analisis hasil. Setiap tahapan dilakukan secara sistematis mulai dari pengumpulan dan preprocessing data, proses pembangunan model, evaluasi performa model, hingga analisis hasil untuk menentukan model terbaik dalam prediksi stroke.



Gambar 1. Tahapan Penelitian

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 4, Juni 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i4.9711

Pada gambar 1 dijelaskan bahwa, Penelitian ini menggunakan pendekatan *machine learning* yang kompleks dalam pengembangan model prediktif untuk klasifikasi penyakit stroke. Metode ini melibatkan beberapa tahap utama: pengumpulan data, *preprocessing* data, analisis eksplorasi data, pemilihan fitur, optimasi *hyperparameter*, pelatihan, analisis hasil, dan evaluasi hasil.

2.1 Pengumpulan Data

Dataset stroke diambil dari situs web *Kaggle*, yang dapat diakses melalui tautan https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset. Dataset terdiri dari 5.110 data dengan 12 variabel. Variabel yang digunakan meliputi usia, hipertensi, penyakit jantung, kadar glukosa darah, indeks massa tubuh, dan kebiasaan merokok. Dataset terdiri dari distribusi kelas yang tidak seimbang karena jumlah kasus stroke lebih sedikit dibandingkan non-stroke. Oleh karena itu, penelitian ini menerapkan metode SMOTE untuk menangani ketidakseimbangan data sebelum proses klasifikasi dilakukan.

Tabel 1. Variabel dataset stroke

No	Variabel	Deskripsi	Tipe Data
1.	*Age*	Usia/umur pasien [tahun]	*Numerikal*
2.	*Gender*	Jenis Kelamin [M: *Male*, F: *Female*]	*Kategorikal*
3.	*Hypertension*	Riwayat hipertensi	*Kategorikal*
4.	*Heart_disease*	Penyakit jantung (1 = ya, 0 = tidak)	*Numerikal*
5.	*Ever_married*	Status pernikahan	*Numerikal*
6.	*Work_type*	Jenis pekerjaan (seperti *Private*, *Self-employed*, dan *Govt job*).	*Kategorikal*
7.	*Residence_type*	mengidentifikasi zona tempat tinggal (*Urban* atau *Rural*)	*Kategorikal*
8.	*Avg_glucose_level*	Glukosa rata-rata dalam darah	*Numerikal*
9.	*BMI*	Indeks massa tubuh	*Numerikal*
10.	*Smoking_status*	Status merokok.	*Kategorikal*
11.	*Stroke*	Individu pernah mengalami stroke	*Numerikal*

2.2 *Pengolahan Data*

Pengolahan data (*Pre-processing)* merupakan serangkaian teknik yang dilakukan untuk mempersiapkan dan mengolah data sebelum digunakan dalam proses analisis oleh model atau algoritma. Tahap ini bertujuan untuk meningkatkan kualitas data dengan menangani nilai hilang, menyamakan format data, mengatasi ketidakseimbangan kelas, melakukan encoding pada variabel kategorikal, serta menormalisasi data numerik. Dengan adanya proses *pre-processing*, data yang digunakan dalam pelatihan model menjadi lebih bersih, terstruktur, dan optimal sehingga dapat membantu meningkatkan akurasi hasil prediksi [11].

2.3 *Modeling*

Dalam penelitian ini, model prediksi memanfaatkan teknik *machine learning* seperti *Random Forest*, *XGBoost, dan logistic regression*, yang dirancang untuk menilai performanya dalam memprediksi penyakit stroke. Kinerja ketiga model tersebut akan dievaluasi menggunakan berbagai metrik, termasuk akurasi, *presisi*, *recall*, serta nilai *AUC*, guna menentukan model terbaik untuk deteksi stroke [12].

2.4 *Exploratory Data Analysis (EDA)*

Exploratory Data Analysis (EDA) dilakukan untuk memahami pola dan hubungan antar-fitur. Tahapan ini melibatkan visualisasi distribusi fitur, analisis korelasi, dan pengecekan ketidakseimbangan kelas antara pasien yang menderita stroke dan yang tidak. Jika terdapat ketidakseimbangan kelas yang signifikan, digunakan teknik sampling seperti *oversampling* pada kelas minoritas atau metode SMOTE untuk menyeimbangkan data. Pemilihan dan rekayasa fitur dilakukan untuk mengeliminasi fitur-fitur yang kurang relevan, dengan bantuan algoritma seperti *Recursive Feature Elimination (RFE)* dan *Principal Component Analysis (PCA)* yang berperan dalam proses ini. Menyeleksi fitur yang paling berdampak pada prediksi. Rekayasa fitur juga diterapkan untuk menciptakan fitur baru dari variabel yang ada, dengan harapan meningkatkan kemampuan prediktif model [13].

2.5 *Pemilihan dan Rekayasa Fitur*

Pemilihan fitur dilakukan untuk mengidentifikasi fitur yang paling berpengaruh terhadap prediksi stroke. Proses ini dilakukan menggunakan metode Recursive Feature Elimination (RFE). Selain itu, rekayasa fitur diterapkan untuk meningkatkan kemampuan prediktif model melalui pembentukan fitur baru dari variabel yang tersedia.

2.5.1 Random Forest Classifier

Random Forest merupakan algoritma *ensemble* berbasis pohon keputusan yang mengintegrasikan sejumlah pohon keputusan *(decision trees)* guna menghasilkan prediksi yang lebih stabil dan tepat. Melalui mekanisme *voting* dari berbagai pohon, algoritma ini mampu mengelola data kompleks serta mengurangi risiko *overfitting*. Secara umum,

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 4, Juni 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i4.9711

random forest menghasilkan tingkat akurasi yang tinggi, meskipun cenderung kurang sensitif terhadap kelas minoritas [14]. Pada penelitian ini, model diimplementasikan menggunakan library *Scikit-learn* dengan pembagian data sebesar 80% untuk training dan 20% untuk testing. Proses evaluasi dilakukan menggunakan metode 5-*fold cross validation*, sedangkan optimasi parameter dilakukan melalui *GridSearchCV* pada parameter *n_estimators* dan *max_depth*.

2.5.2 XGBoost

XGBoost (Extreme Gradient Boosting) adalah algoritma *boosting* yang kuat dan efisien. *XGBoost* sangat efektif untuk menangani data yang tidak seimbang, dan sering menghasilkan akurasi tinggi dan *recall* yang baik untuk kelas minoritas, menjadikannya optimal untuk mendeteksi kasus positif dalam dataset medis [15]. Pada penelitian ini, implementasi *XGBoost* dilakukan menggunakan library *XGBoost* dengan pembagian data training dan testing sebesar 80:20. Proses evaluasi model dilakukan menggunakan *5-fold cross validation* serta pengujian beberapa *hyperparameter*, seperti *learning_rate*, *max_depth*, dan *n_estimators*, untuk memperoleh performa model yang optimal.

2.5.3 Logistic Regression

Logistic Regression adalah algoritma klasifikasi yang umum dipakai dalam *machine learning*, khususnya untuk tugas klasifikasi *biner*. Algoritma ini beroperasi dengan menghitung probabilitas keanggotaan data pada kelas tertentu melalui fungsi logistik *(sigmoid*). Kelebihannya meliputi kesederhanaan model, kecepatan komputasi, serta kemudahan interpretasi hasil, menjadikannya pilihan populer dalam penelitian kesehatan untuk memprediksi risiko penyakit. Meski demikian, pada dataset yang kompleks, kinerjanya cenderung kalah dibandingkan metode *ensemble* seperti *random forest* atau algoritma *boosting* seperti *XGBoost* [16]**.**

2.6 *Hyperparameter Tuning*

Hyperparameter tuning adalah proses memilih kombinasi terbaik dari *hyperparameter* (parameter yang tidak dipelajari langsung dari data oleh model) untuk memaksimalkan kinerja model. *Grid search* adalah teknik untuk melakukan *hyperparameter tuning* pada model *machine learning*, di mana tujuan utamanya adalah untuk menemukan kombinasi parameter terbaik yang memberikan performa terbaik pada model. Pada penelitian ini digunakan metode *GridSearchCV* untuk mencari kombinasi *hyperparameter* terbaik pada algoritma *random forest* [17]. Parameter yang diuji meliputi jumlah *decision tree (n_estimators)*, kedalaman pohon *(max_depth)*, dan minimum jumlah sampel pada proses splitting *(min_samples_split)*. Proses tuning dilakukan menggunakan metode 5-*fold cross validation* untuk memastikan kestabilan performa model.

2.7 Teknik SMOTE

Ketidakseimbangan kelas dalam dataset medis sering kali mengganggu kinerja model klasifikasi, sebab jumlah data kelas mayoritas jauh lebih dominan daripada kelas minoritas. Untuk mengatasi permasalahan tersebut, digunakan metode *Synthetic Minority Oversampling Technique* (SMOTE) yang berfungsi menyeimbangkan distribusi data dengan cara menghasilkan sampel buatan pada kelas yang jumlahnya lebih sedikit. Sejumlah penelitian menunjukkan bahwa penerapan SMOTE dapat membantu meningkatkan kinerja model machine learning dalam melakukan prediksi stroke [18].

2.8 *Evaluasi*

Proses evaluasi menjadi tahap penting untuk mengukur performa model yang digunakan. Penelitian ini menerapkan sejumlah metrik utama guna menilai kemampuan model dalam memprediksi stroke, meliputi presisi, *recall*, *F1 score*, dan akurasi. Kinerja model dinilai dengan membandingkan proporsi prediksi benar terhadap total prediksi secara keseluruhan, menggunakan rumus yang sesuai untuk setiap metrik [19].

a. Akurasi mengukur persentase prediksi benar dari seluruh prediksi yang dibuat, mencakup kelas positif maupun negatif. Metrik ini mencerminkan seberapa sering model menghasilkan hasil tepat secara menyeluruh.

$$Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{TP + FP + FN + TN} \quad (1)$$

b. Presisi mengukur ketepatan prediksi positif yang dihasilkan model. Semakin tinggi presisi, semakin minim kesalahan mengklasifikasikan kasus negatif sebagai positif

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

c. *Recall* menilai kemampuan model dalam mendeteksi semua kasus positif secara benar. Semakin tinggi nilai *recall*, semakin efektif model menangkap kasus positif yang ada.

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 4, Juni 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i4.9711

$$Recall = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3)$$

d. *F1-Score* merupakan rata-rata harmonis dari presisi dan *recall*, yang menyediakan keseimbangan di antara keduanya. Metrik ini sangat berguna ketika perlu mempertimbangkan kedua aspek secara seimbang, terutama pada kondisi ketidakseimbangan kelas.

$$F1 - Score = 2 \times \frac{precision \times recall}{precision + recall} \quad (4)$$

Penjelasan rumus adalah sebagai berikut: *True Positive* (TP) mengacu pada jumlah data positif yang berhasil diklasifikasikan secara benar oleh model. *True Negative* (TN) adalah data negatif yang diprediksi dengan akurat. *False Positive* (FP) terjadi ketika data negatif salah dikenali sebagai positif, sedangkan *False Negative* (FN) adalah data positif yang keliru dikategorikan sebagai negatif [20].

3. HASIL DAN PEMBAHASAN

3.1 Pengolahan Data

Gambar 3 menggambarkan detail atribut dalam dataset stroke, meliputi id (identitas unik), *gender* (jenis kelamin), *age* (usia), *hypertension* (status hipertensi), *heart disease* (status penyakit jantung), *ever_married (status pernikahan)*, *work type* (jenis pekerjaan), *residence_type* (tipe tempat tinggal), *avg_glucose_level* (rata-rata kadar glukosa darah), *BMI* (indeks massa tubuh), *smoking status* (status merokok), serta stroke (indikator riwayat stroke pada individu). Dataset tersebut dimanfaatkan untuk menganalisis faktor risiko yang berpotensi terkait dengan terjadinya stroke.

```
import pandas as pd
data = pd.read_csv('/content/dataset penyakit stroke.csv')
from IPython.display import display
display(data.head())
```

	id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
0	9046	Male	67.0	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
1	51676	Female	61.0	0	0	Yes	Self-employed	Rural	202.21	NaN	never smoked	1
2	31112	Male	80.0	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
3	60182	Female	49.0	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
4	1665	Female	79.0	1	0	Yes	Self-employed	Rural	174.12	24.0	never smoked	1

Gambar 2. Variabel dataset stroke

3.2 *Missing Values*

Deteksi *missing values* dilakukan agar proses analisis atau pemodelan dapat dilakukan secara akurat dan tanpa bias. Berdasarkan gambar 3, semua kolom, kecuali *BMI*, tidak memiliki nilai yang hilang (jumlah *missing values* = 0). Namun, kolom *BMI* memiliki 201 nilai yang hilang.

	0
id	0.000000
gender	0.000000
age	0.000000
hypertension	0.000000
heart_disease	0.000000
ever_married	0.000000
work_type	0.000000
Residence_type	0.000000
avg_glucose_level	0.000000
bmi	3.933464
smoking_status	0.000000
stroke	0.000000

dtype: float64

Gambar 3. Analisis *missing values*

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 4, Juni 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i4.9711

Gambar 4 menunjukkan hasil perhitungan jumlah *missing values* (nilai yang hilang) setelah dilakukan penanganan *missing values* pada setiap kolom dalam dataset. Berdasarkan hasil ini, semua kolom tidak memiliki nilai yang hilang, termasuk bmi yang sebelumnya teridentifikasi memiliki *missing values* pada gambar sebelumnya. Jumlah *missing values* untuk setiap kolom adalah 0, yang berarti data dalam dataset ini telah lengkap di setiap kolomnya. Informasi ini menandakan bahwa dataset siap untuk analisis lebih lanjut tanpa perlu penanganan khusus terkait *missing values.*

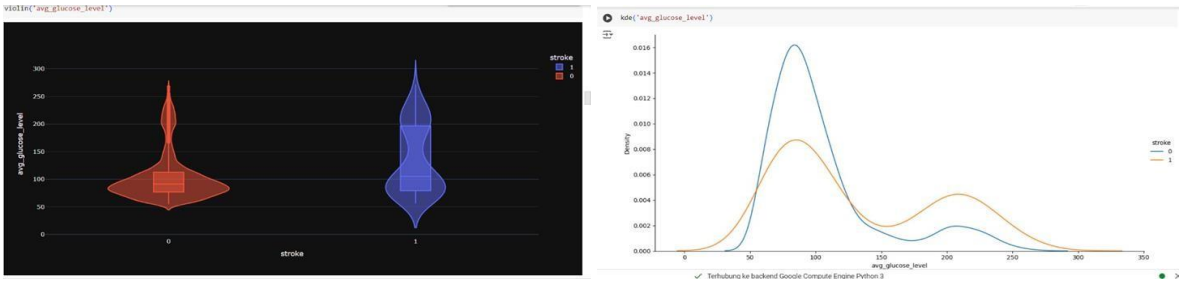
	0
id	0
gender	0
age	0
hypertension	0
heart_disease	0
ever_married	0
work_type	0
Residence_type	0
avg_glucose_level	0
bmi	0
smoking_status	0
stroke	0

dtype: int64

Gambar 4. Setelah penanganan *missing values*

3.3 *Exploratory Data Analysis* (EDA)

Dari kedua hasil *Exploratory Data Analysis* (EDA) di atas, dapat disimpulkan bahwa terdapat perbedaan distribusi kadar glukosa rata-rata *(avg_glucose_level)* antara individu yang mengalami stroke dan yang tidak. Pada *violin plot*, terlihat bahwa individu dengan stroke cenderung memiliki kadar glukosa yang lebih tinggi dibandingkan dengan individu tanpa stroke. Hal ini diperkuat oleh KDE plot, di mana distribusi kadar glukosa untuk kelompok dengan stroke menunjukkan rentang yang lebih luas dan puncak distribusi pada kadar glukosa yang lebih tinggi dibandingkan dengan kelompok tanpa stroke. Analisis ini mengindikasikan bahwa kadar glukosa yang tinggi mungkin berhubungan dengan peningkatan risiko stroke, yang dapat menjadi faktor penting dalam pemodelan prediktif lebih lanjut.



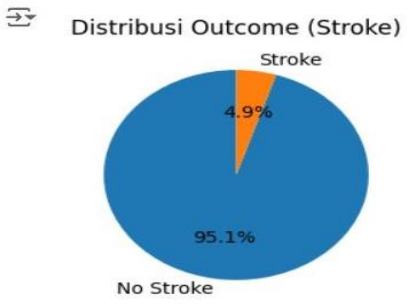
Gambar 5. *Visualisasi EDA Multivariat*

Dari grafik ini, terlihat bahwa individu yang tidak mengalami stroke (garis biru) memiliki puncak distribusi yang lebih tinggi di kisaran kadar glukosa yang lebih rendah, sekitar 80–100 mg/dL. Sementara itu, individu yang mengalami stroke (garis oranye) cenderung memiliki distribusi yang lebih menyebar dan puncak yang lebih rendah, dengan beberapa distribusi pada kadar glukosa yang lebih tinggi, khususnya di atas 200 mg/dL. Hal ini menunjukkan adanya kecenderungan bahwa individu dengan kadar glukosa lebih tinggi mungkin memiliki risiko stroke yang lebih besar.

3.4 SMOTE

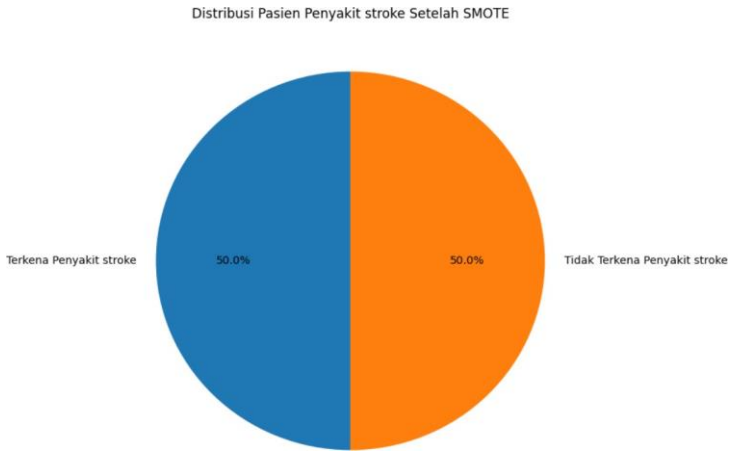
Dari hasil eksplorasi dataset stroke, diketahui bahwa jumlah data pasien yang mengalami stroke hanya 249, sementara pasien yang tidak mengalami stroke mencapai 4.861. Pada Gambar 6, hal ini menunjukkan bahwa data pasien tanpa stroke jauh lebih dominan dibandingkan data pasien dengan stroke. Sehingga atribut stroke dalam dataset ini memiliki ketidakseimbangan kelas. Untuk mengatasi ketidakseimbangan ini, kita akan menggunakan teknik SMOTE untuk menyeimbangkan dataset.

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 4, Juni 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i4.9711



Gambar 6. *Outcome* sebelum dilakukan teknik SMOTE

Setelah menerapkan SMOTE, seperti yang terlihat pada Gambar 7, distribusi data menjadi seimbang dengan proporsi 50% untuk masing-masing kelas, baik untuk kelas Stroke maupun Tidak Stroke. Penerapan SMOTE ini membantu menambahkan sampel sintetis pada kelas minoritas, sehingga model dapat belajar secara lebih seimbang dari kedua kelas dan mengurangi bias terhadap kelas mayoritas. Perbandingan antara kedua grafik menunjukkan bahwa SMOTE berhasil mengatasi masalah ketidakseimbangan kelas dengan menghasilkan data pelatihan yang lebih seimbang, yang diharapkan dapat meningkatkan kinerja model dalam memprediksi kelas minoritas (stroke).



Gambar 7. *Outcome* setelah dilakukan teknik smote

3.5 Pemodelan

3.5.1 *Random Forest*

Hasil perhitungan algoritma *Random Forest* pada Gambar 8 menunjukkan bahwa model memperoleh akurasi sebesar 94%. Berdasarkan hasil evaluasi menggunakan *weighted average*, *Random Forest* menghasilkan *precision* sebesar 90%, *recall* sebesar 94%, dan *F1-score* sebesar 91%. Namun, pada kelas 1 (stroke), model hanya memperoleh *precision* 20%, *recall* 2%, dan *F1-score* 3%, yang menunjukkan bahwa sebagian besar kasus stroke belum berhasil terdeteksi. Hasil ini menunjukkan bahwa *Random Forest* memiliki performa keseluruhan yang baik berdasarkan metrik *weighted average*. Namun, kemampuan model dalam mendeteksi kasus stroke masih terbatas karena nilai *recall* pada kelas stroke relatif rendah.

```
Akurasi: 0.9363992172211351

Laporan Klasifikasi:
              precision    recall  f1-score   support

           0       0.94      1.00      0.97       960
           1       0.20      0.02      0.03        62

    accuracy                           0.94      1022
   macro avg       0.57      0.51      0.50      1022
weighted avg       0.90      0.94      0.91      1022


Confusion Matrix:
[[956   4]
 [ 61   1]]
```

Gambar 8. Hasil Evaluasi *Random Forest*

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 4, Juni 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i4.9711

3.5.2 *XGBoost*

```
Akurasi (XGBoost): 0.9324853228962818

Laporan Klasifikasi (XGBoost):
              precision    recall  f1-score   support

           0       0.94      0.99      0.96       960
           1       0.29      0.08      0.13        62

    accuracy                           0.93      1022
   macro avg       0.62      0.53      0.55      1022
weighted avg       0.90      0.93      0.91      1022


Confusion Matrix (XGBoost):
[[948  12]
 [ 57   5]]
```

Gambar 9. Hasil Evaluasi *XGBoost*

Berdasarkan hasil evaluasi pada Gambar 9, model *XGBoost* memperoleh akurasi sebesar 93%. Pada kelas 0 (non-stroke), model menghasilkan *precision* sebesar 94%, *recall* sebesar 99%, dan *F1-score* sebesar 96%, yang menunjukkan kemampuan sangat baik dalam mengklasifikasikan pasien non-stroke. Sementara itu, pada kelas 1 (stroke), model memperoleh *precision* sebesar 29%, *recall* sebesar 8%, dan *F1-score* sebesar 13%. Hasil tersebut menunjukkan bahwa meskipun *XGBoost* memiliki performa keseluruhan yang tinggi, kemampuan model dalam mendeteksi kasus stroke masih terbatas. Secara keseluruhan, nilai *weighted average* menunjukkan *precision* sebesar 90%, *recall* sebesar 93%, dan *F1-score* sebesar 91%. Hal ini mengindikasikan bahwa performa model lebih dipengaruhi oleh keberhasilan klasifikasi kelas mayoritas dibandingkan kemampuan mendeteksi kelas stroke.

3.5.3 *Logistic Regression*

Berdasarkan hasil evaluasi pada Gambar 10, *Logistic Regression* memperoleh akurasi sebesar 74%. Pada kelas 0 (non-stroke), model menghasilkan *precision* sebesar 98%, *recall* sebesar 74%, dan *F1-score* sebesar 85%. Sementara itu, pada kelas 1 (stroke), model memperoleh *precision* sebesar 16%, *recall* sebesar 75%, dan *F1-score* sebesar 27%. Nilai *recall* yang tinggi pada kelas stroke menunjukkan bahwa *Logistic Regression* mampu mengidentifikasi sebagian besar kasus stroke yang terdapat dalam dataset. Berdasarkan *confusion matrix*, model berhasil mendeteksi 60 dari 80 kasus stroke. Secara keseluruhan, nilai *weighted average* menghasilkan *precision* sebesar 93%, *recall* sebesar 74%, dan *F1-score* sebesar 81%. Meskipun akurasi keseluruhan lebih rendah dibandingkan *Random Forest* dan *XGBoost*, *Logistic Regression* menunjukkan kemampuan yang lebih baik dalam mendeteksi kasus stroke sehingga berpotensi digunakan pada skenario yang memprioritaskan identifikasi pasien berisiko stroke.

```
Logistic Regression Accuracy: 0.7449139280125195
Confusion Matrix:
[[892 306]
 [ 20  60]]

Classification Report:
              precision    recall  f1-score   support

           0       0.98      0.74      0.85      1198
           1       0.16      0.75      0.27        80

    accuracy                           0.74      1278
   macro avg       0.57      0.75      0.56      1278
weighted avg       0.93      0.74      0.81      1278
```

Gambar 10. Hasil Evaluasi *Logistic Regression*

3.5.4 *Hyperparameter Tuning*

Hyperparameter tuning dilakukan untuk mengoptimalkan parameter model agar menghasilkan performa klasifikasi yang lebih baik. Pada penelitian ini, *hyperparameter tuning* diterapkan pada model *Random Forest* untuk mengevaluasi pengaruhnya terhadap nilai *accuracy*, *precision*, *recall*, dan *F1-score* dalam prediksi stroke. Berdasarkan hasil evaluasi setelah dilakukan *hyperparameter tuning*, *Random Forest* memperoleh *accuracy* sebesar 93%, precision sebesar 88%, recall sebesar 93%, dan F1-score sebesar 91%. Hasil tersebut menunjukkan bahwa proses *hyperparameter tuning* tidak memberikan peningkatan performa yang signifikan dibandingkan model sebelum tuning. Nilai *accuracy*, precision, dan *recall* mengalami sedikit penurunan, sedangkan *F1-score* tetap berada pada nilai yang sama. Dengan demikian, parameter awal yang digunakan pada *Random Forest* telah mampu menghasilkan performa yang relatif baik pada dataset yang digunakan.

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 4, Juni 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i4.9711

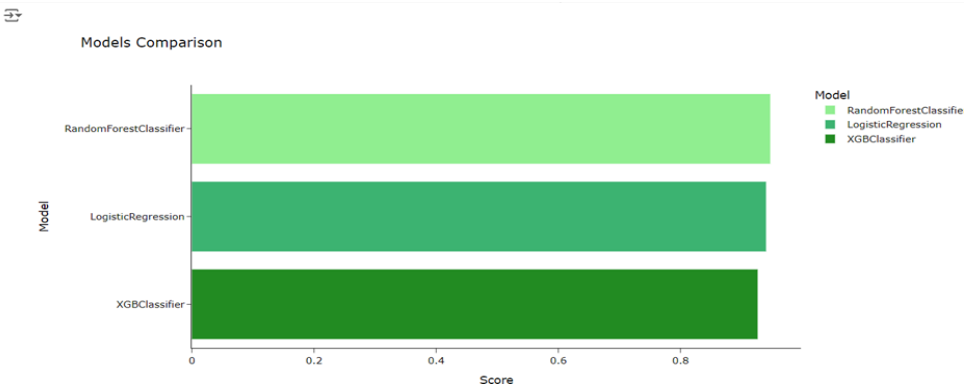

```
Fitting 3 folds for each of 81 candidates, totalling 243 fits
Skor Akurasi Terbaik: 0.9538098103397495
Confusion Matrix:
[[1194    4]
 [  80    0]]

Accuracy: 0.9342723004694836
Recall: 0.0
F1 Score: 0.0

Classification Report:
              precision    recall  f1-score   support

           0       0.94      1.00      0.97      1198
           1       0.00      0.00      0.00        80

    accuracy                           0.93      1278
   macro avg       0.47      0.50      0.48      1278
weighted avg       0.88      0.93      0.91      1278
```

Gambar 11. Hasil Evaluasi *Hyperparameter Tuning*

3.6 Evaluasi

3.6.1 Perbandingan hasil 3 algoritma sebelum dilakukan *hyperparameter tuning*

Setelah proses pemodelan selesai, setiap model dievaluasi menggunakan empat metrik utama, yaitu akurasi, *precision*, *recall*, dan *F1-Score*.



Gambar 12. Perbandingan Kinerja Model Sebelum *Hyperparameter Tuning*

Tabel 2. Hasil Kinerja Model Sebelum *Hyperparameter Tuning (Weighted Average)*

Model	***Accuracy***	***Precision***	***Recall***	***F1-Score***
Random Forest	0,94	0,90	0,94	0,91
XGBoost	0,93	0,90	0,93	0,91
Logistic Regression	0,74	0,93	0,74	0,81

Dari hasil evaluasi model dapat dilihat pada tabel 2 dan visualisasi pada gambar 12, *Random Forest* memperoleh performa keseluruhan terbaik dengan *accuracy* sebesar 94%, *precision* 90%, *recall* 94%, dan *F1-score* 91%. XG*Boost* menunjukkan performa yang hampir setara dengan *accuracy* 93%, *precision* 90%, *recall* 93%, dan F1-score 91%. Sementara itu, *Logistic Regression* memperoleh accuracy sebesar 74%, precision 93%, recall 74%, dan *F1-score* 81%. Nilai pada Tabel 2 merupakan hasil *weighted average* sehingga menggambarkan performa model secara keseluruhan. Namun, kemampuan model dalam mendeteksi kasus stroke secara spesifik dapat dilihat pada Tabel 3 yang menampilkan performa kelas stroke (kelas 1).

Tabel 3. Performa Deteksi Kelas Stroke (Kelas 1)

Model	***Precision***	***Recall***	***F1-Score***
Random Forest	0,20	0,02	0,03
XGBoost	0,29	0,08	0,13
Logistic Regression	0,16	0,75	0,27

Berdasarkan Tabel 3, *Logistic Regression* menunjukkan kemampuan terbaik dalam mendeteksi kasus stroke dengan nilai *recall* sebesar 75%, yang berarti sebagian besar pasien stroke berhasil diidentifikasi oleh model. Sebaliknya, *Random Forest* dan *XGBoost* masing-masing hanya memperoleh *recall* sebesar 2% dan 8%, sehingga sebagian besar kasus stroke masih salah diklasifikasikan sebagai non-stroke. Hasil ini menunjukkan bahwa meskipun *Random Forest* dan *XGBoost* memiliki nilai *accuracy* dan *weighted average* yang lebih tinggi, kedua model tersebut belum optimal dalam mengenali kelas minoritas (stroke). Oleh karena itu, *Logistic Regression* lebih

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 4, Juni 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i4.9711

sesuai digunakan pada skenario yang memprioritaskan deteksi dini kasus stroke, sedangkan *Random Forest* dan *XGBoost* lebih unggul dalam performa klasifikasi secara keseluruhan.

3.6.2 Perbandingan hasil 3 algoritma setelah dilakukan *hyperparameter tuning*

Tabel 4. Hasil Kinerja Model Setelah *Hyperparameter Tuning (Weighted Average)*

Model	*Accuracy*	*Precision*	*Recall*	*F1-Score*
Random Forest	0,93	0,88	0,93	0,91
XGBoost	0,93	0,90	0,93	0,91
Logistic Regression	0,74	0,93	0,74	0,81

Berdasarkan Tabel 4, *Random Forest* setelah *hyperparameter tuning* memperoleh *accuracy* sebesar 93%, *precision* 88%, *recall* 93%, dan *F1-score* 91%. Hasil tersebut menunjukkan bahwa proses *hyperparameter tuning* tidak memberikan peningkatan performa yang signifikan dibandingkan model sebelum tuning, karena nilai *accuracy*, *recall*, dan *F1-score* relatif tetap, bahkan *precision* mengalami sedikit penurunan. *XGBoost* menunjukkan performa yang kompetitif dengan *accuracy* 93%, *precision* 90%, *recall* 93%, dan *F1-score* 91%, sehingga menghasilkan kinerja yang sebanding dengan *Random Forest*. Sementara itu, *Logistic Regression* memperoleh *accuracy* 74%, *precision* 93%, *recall* 74%, dan *F1-score* 81%, yang menunjukkan performa lebih rendah dibandingkan kedua model lainnya. Perlu diperhatikan bahwa nilai pada tabel merupakan hasil *weighted average* yang menggambarkan performa model secara keseluruhan. Meskipun *Random Forest* dan *XGBoost* memiliki nilai *accuracy* dan *F1-score* yang lebih tinggi, kemampuan model dalam mendeteksi kasus stroke tetap perlu dievaluasi berdasarkan performa pada kelas stroke sebagai kelas minoritas. Dengan demikian, *hyperparameter tuning* pada *Random Forest* belum mampu meningkatkan performa model secara signifikan dibandingkan sebelum tuning.

4. KESIMPULAN

Penelitian ini membandingkan tiga algoritma machine learning, yaitu *Random Forest*, *XGBoost*, dan *Logistic Regression*, dalam memprediksi penyakit stroke. Berdasarkan hasil evaluasi, *Random Forest* dan XG*Boost* memberikan performa klasifikasi yang lebih baik secara keseluruhan berdasarkan metrik *weighted average*. Hasil pengujian juga menunjukkan bahwa *hyperparameter tuning* pada *Random Forest* tidak memberikan peningkatan performa yang signifikan dibandingkan model sebelum tuning. Selain itu, *Logistic Regression* menunjukkan kemampuan yang lebih baik dalam mendeteksi kasus stroke berdasarkan nilai *recall* pada kelas stroke. Hasil penelitian ini menunjukkan bahwa metode *ensemble learning* lebih efektif dalam menghasilkan performa klasifikasi secara keseluruhan, sedangkan *Logistic Regression* lebih unggul dalam mendeteksi kasus positif stroke. Oleh karena itu, pemilihan model perlu disesuaikan dengan tujuan penggunaan, baik untuk memperoleh performa klasifikasi yang seimbang maupun untuk memaksimalkan deteksi kasus stroke. Oleh karena itu, penelitian selanjutnya disarankan untuk menggunakan dataset dengan jumlah data yang lebih besar dan distribusi kelas yang lebih seimbang agar performa model dapat meningkat. Selain itu, metode penanganan data tidak seimbang seperti ADASYN atau kombinasi SMOTE dan undersampling dapat diterapkan untuk meningkatkan deteksi kelas minoritas. Penggunaan teknik optimasi hyperparameter yang lebih lanjut, seperti Bayesian Optimization, serta perbandingan dengan algoritma lain seperti LightGBM dan CatBoost juga dapat dilakukan guna memperoleh hasil prediksi stroke yang lebih akurat dan stabil.

REFERENCES

[1] H. Khathimah, N. I. Hanifa, Y. W. Putri, and N. Susanti, "Prevalensi Penyakit Stroke Di Puskesmas Dalu Sepuluh," *Prepotif J. Kesehat. Masy.*, vol. 8, no. 2, pp. 4068–4073, 2024, doi: 10.31004/prepotif.v8i2.30346.

[2] B. K. Kim, S. Park, M. K. Han, J. H. Hong, D. I. Lee, and K. S. Yum, "Deep learning for prediction of mechanism in acute ischemic stroke using brain diffusion magnetic resonance image," *J. Neurocritical Care*, vol. 16, no. 2, pp. 85–93, 2023, doi: 10.18700/jnc.230039.

[3] WHO, *World health statistics 2023: monitoring health for the sdgs, sustainable development goals*, vol. 27, no. 2. 2023.

[4] Balitbang Kemenkes RI, "Laporan Nasional Riset Kesehatan Dasar (Riskesdas)," *Balitbang Kemenkes RI*, p. 124, 2013, doi: 10.1126/science.127.3309.1275.

[5] J. J. Pangaribuan *et al.*, "Machine Learning," vol. 6, no. 2, 2021.

[6] I. N. Rizki, D. Prayoga, M. L. Puspita, and M. Q. Huda, "Implementasi Exploratory Data Analysis Untuk Analisis Dan Visualisasi Data Penderita Stroke Kalimantan Selatan Menggunakan Platform Tableau," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 1, 2024, doi: 10.23960/jitet.v12i1.3856.

[7] L. Mongkau, F. L. F. G. Langi, and A. F. C. Kalesaran, "Studi Ekologi Prevalensi Diabetes Melitus Dengan Stroke Di Indonesia," *PREPOTIF J. Kesehat. Masy.*, vol. 6, no. 2, pp. 1156–1162, 2022, doi: 10.31004/prepotif.v6i2.4027.

[8] S. Arifin and I. Tahyudin, "Optimasi Prediksi Prediabetes dengan Metode Fitur Selection dan Imbalance Learning.," *Techno. com*, vol. 24, no. 1, pp. 68–80, 2025.

[9] M. F. Banjar, I. Irawati, F. Umar, and L. N. Hayati, "Analysis of Stroke Classification Using Random Forest Method,"

Ilk. J. Ilm., vol. 14, no. 3, pp. 186–193, 2022, doi: 10.33096/ilkom.v14i3.1252.186-193.

[10] Y. Aulia, A. Andriyansyah, S. Suharjito, and S. W. Nensi, "Analisis Prediksi Stroke dengan Membandingkan Tiga Metode Klasifikasi Decision Tree, Naïve Bayes, dan Random Forest," *J. Ilmu Komput. dan Inform.*, vol. 3, no. 2, pp. 89–98, 2024, doi: 10.54082/jiki.90.

[11] B. A. Febryanto and I. Tahyudin, "Perbandingan Algoritma CNN, LSTM, FNN untuk Diagnosa Fibrosis Hati dengan Citra Medis," *Techno.Com*, vol. 24, no. 1, pp. 41–55, 2025.

[12] L. R. Sitompul, A. A. Nababan, M. L. Manihuruk, W. A. Ponsen, and S. Supriyandi, "Comparison of Xgboost, Random Forest and Logistic Regression Algorithms in Stroke Disease Classification," *Sinkron*, vol. 9, no. 2, pp. 957–968, 2025, doi: 10.33395/sinkron.v9i2.14794.

[13] U. N. Wisesty, T. A. B. Wirayuda, F. Sthevanie, and R. Rismala, "Analysis of Data and Feature Processing on Stroke Prediction using Wide Range Machine Learning Model," *J. Online Inform.*, vol. 9, no. 1, pp. 29–40, 2024, doi: 10.15575/join.v9i1.1249.

[14] A. Riyadi, J. A. Tambunan, and A. Wijaya, "Perbandingan Akurasi Support Vector Machine dan Random Forest pada Prediksi Diabetes Melitus," vol. 2, pp. 1314–1321, 2025.

[15] M. riki Atsauri, H. Mawengkang, and S. Efendi, "Enhancing Unbalanced Data Classification with Cross-Validation and Extreme Gradient Boosting: A Comprehensive Analysis," *J. Informatics Telecommun. Eng.*, vol. 7, no. 1, pp. 30–42, 2023, doi: 10.31289/jite.v7i1.8690.

[16] N. Nasution, M. A. Hasan, and F. Bakri Nasution, "Predicting Heart Disease Using Machine Learning: An Evaluation of Logistic Regression, Random Forest, SVM, and KNN Models on the UCI Heart Disease Dataset," *IT J. Res. Dev.*, vol. 9, no. 2, pp. 140–150, 2025, doi: 10.25299/itjrd.2025.17941.

[17] N. P. Nur Fauzi, S. Khomsah, and A. D. Putra Wicaksono, "Penerapan Feature Engineering dan Hyperparameter Tuning untuk Meningkatkan Akurasi Model Random Forest pada Klasifikasi Risiko Kredit," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 12, no. 2, pp. 251–262, 2025, doi: 10.25126/jtiik.2025128472.

[18] L. Pasiolo *et al.*, "Penyakit Stroke Dengan Algoritma Support Vector Machine," *J. Sist. Inf.*, vol. 7, no. 1, pp. 61–73, 2025.

[19] N. N. Tahyudin, Imam; Arifin, Samsul; Rizaqi, Hanif; Febryanto, Bagas Aji; Putra, Bernardus Septian Cahya; Holifah, Anggita Nur; Puspitasari, *Data Science: Teori dan Implementasi*. Zahira Media Publisher, 2025.

[20] Z.-H. Zhou, *Machine Learning*. Nanjing, China, 2021. doi: 10.1007/978-981-15-1967-3.