Stroke Prediction using Machine Learning

Yusra Erlangga Putra[1], Sheryl Anastasya[2], Resky Auliyah Kartini Askin[3], Ivan Betrandi[4], Amaliah Diah[5]
Departemen Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Hasanuddin, Makassar, Indonesia
[1]yusraerlangg@gmail.com, [2]sherylanastasya2809@gmail.com

Abstrak

Stroke adalah penyebab utama kecacatan jangka panjang dan kematian di seluruh dunia, dengan risiko yang meningkat seiring bertambahnya usia serta adanya faktor risiko seperti hipertensi dan diabetes. Tujuan penelitian ini adalah mengembangkan model prediksi dini yang akurat untuk stroke sehingga memungkinkan intervensi perawatan kesehatan preventif yang efektif. Penelitian ini menggunakan algoritma Random Forest dan Support Vector Machine (SVM). Karena ketidakseimbangan pada himpunan data, teknik Synthetic Minority Oversampling Technique (SMOTE) diterapkan untuk meningkatkan representasi data minoritas. Model dioptimalkan melalui penyesuaian hiperparameter menggunakan Optimasi Bayesian. Evaluasi model dilakukan dengan metrik akurasi, presisi, sensitivitas, dan skor-F1, menggunakan validasi silang K-Fold yang dimodifikasi untuk memastikan keandalan pada data yang belum terlihat. Hasil eksperimen menunjukkan bahwa algoritma SVM dengan SMOTE dan Optimasi Bayesian mencapai akurasi tertinggi. Temuan ini menunjukkan bahwa model machine learning yang dioptimalkan dapat memberikan kontribusi signifikan dalam prediksi dini stroke dan mendukung pengambilan keputusan dalam sistem perawatan kesehatan preventif.

Kata kunci: *Prediksi Stroke, machine learning, Random Forest, Support Vector Machine, SMOTE, Bayesian Optimization*

1. Introduction

1.1 Background

Stroke merupakan masalah kesehatan global yang serius, menempati posisi kedua sebagai penyebab kematian tertinggi dan posisi ketiga sebagai penyebab kecacatan di dunia. Menurut data WHO, satu dari empat orang berisiko mengalami stroke dalam masa hidupnya, dengan 70% kasus terjadi di negara-negara berpenghasilan rendah dan menengah[1]. Di Indonesia sendiri, angka kejadian stroke mencapai 10,9 per 1000 penduduk, dengan sekitar 137 orang per 100.000 penduduk meninggal akibat stroke setiap tahunnya[2]. Beban ekonomi yang ditimbulkan stroke juga sangat besar, mencakup biaya perawatan medis langsung dan hilangnya produktivitas akibat kecacatan jangka panjang.

Kompleksitas faktor risiko stroke yang meliputi tekanan darah tinggi, kolesterol, diabetes, obesitas, dan gaya hidup, menjadikan prediksi stroke sebagai tantangan yang signifikan dalam dunia kesehatan[3]. Namun, deteksi dini risiko stroke memiliki potensi besar dalam mencegah kejadian stroke dan menurunkan tingkat kematian serta kecacatan. Studi menunjukkan bahwa hingga 80% kasus stroke dapat dicegah melalui manajemen faktor risiko yang tepat[4]. Oleh karena itu, pengembangan sistem prediksi stroke yang akurat menjadi sangat penting dalam upaya pencegahan.

Prediksi dini stroke menggunakan metode konvensional seringkali terbatas dalam kemampuannya mengintegrasikan berbagai faktor risiko secara simultan. Perkembangan *machine learning* membuka peluang baru dalam meningkatkan akurasi prediksi stroke. Dengan kemampuannya menganalisis data kompleks dalam jumlah besar, teknik *machine learning* dapat mengidentifikasi pola-pola tersembunyi dan korelasi antar berbagai faktor risiko yang sulit dideteksi melalui metode konvensional.

Penelitian ini mengusulkan pendekatan inovatif dengan algoritma Random Forest dan Support Vector Machine (SVM) untuk prediksi stroke. Kedua algoritma ini dipilih karena kemampuannya dalam menangani data kesehatan yang kompleks dan menghasilkan model prediksi yang akurat. Untuk mengatasi masalah ketidakseimbangan data yang umum dalam kasus medis, diterapkan teknik SMOTE (Synthetic Minority Oversampling Technique). Selanjutnya, optimasi Bayesian digunakan untuk meningkatkan performa model melalui penyetelan hiperparameter yang optimal. Pendekatan komprehensif ini diharapkan dapat menghasilkan sistem prediksi stroke yang lebih andal dan aplikatif dalam praktik klinis.

1.2 Literature Review

1.3 Research Rationale

Penelitian ini didasari oleh beberapa pertimbangan penting dalam konteks prediksi stroke menggunakan *machine learning*. Pertama, meskipun telah banyak penelitian yang menggunakan berbagai algoritma seperti Logistic Regression[5], Decision Tree, dan Neural Network[6], masih terdapat tantangan dalam menangani ketidakseimbangan data pada kasus medis. Ketidakseimbangan ini dapat menyebabkan bias dalam model prediksi yang berpotensi menghasilkan kesalahan diagnosis serius dalam praktik klinis.

Kedua, penelitian-penelitian sebelumnya[7, 8, 9] cenderung berfokus pada penggunaan algoritma tunggal tanpa mempertimbangkan optimasi parameter secara sistematis. Pendekatan tersebut berpotensi menghasilkan model yang suboptimal. Oleh karena itu, penelitian ini mengusulkan integrasi teknik Optimasi Bayesian untuk mengoptimalkan parameter model.

1.4 Research Questions and Objectives

Berdasarkan pertimbangan tersebut, penelitian ini bertujuan menjawab beberapa pertanyaan kunci: (1) bagaimana perbandingan efektivitas Random Forest dan SVM linear dalam memprediksi risiko stroke?, (2) seberapa signifikan pengaruh optimasi parameter menggunakan Optimasi Bayesian terhadap performa model?, dan (3) bagaimana dampak penerapan teknik SMOTE terhadap kinerja prediksi pada data yang tidak seimbang?

Untuk menjawab pertanyaan-pertanyaan tersebut, penelitian ini memiliki tiga sasaran utama:

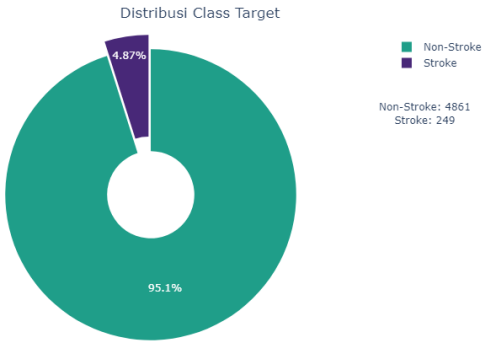
1. Mengembangkan dan membandingkan model prediksi stroke menggunakan Random Forest dan SVM linear
2. Mengoptimalkan parameter model menggunakan Optimasi Bayesian untuk meningkatkan akurasi prediksi
3. Mengevaluasi efektivitas teknik SMOTE dalam menangani ketidakseimbangan data medis

Hasil penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan sistem pendukung keputusan klinis untuk deteksi dini risiko stroke.

2. Research Methods

2.1 Dataset Description

Dataset yang digunakan dalam penelitian ini berasal dari *Kaggle Stroke Prediction Dataset*[10], yang berisi informasi kesehatan dari 5110 pasien dan memiliki 12 variabel. Dataset ini mencakup berbagai parameter demografis dan faktor risiko kesehatan yang berpotensi terkait dengan kejadian stroke. Data dikumpulkan dari berbagai fasilitas kesehatan dan mencakup informasi seperti usia, jenis kelamin, berbagai penyakit, dan gaya hidup pasien. Deskripsi fitur dataset dapat dilihat pada Tabel 1. Output kolom stroke adalah variabel target biner yang menunjukkan status stroke pasien, dengan nilai 1 yang berarti pasien pernah mengalami stroke dan nilai 0 yang berarti tidak pernah mengalami stroke. Dari total 5110 sampel dalam dataset, hanya terdapat 249 kasus stroke (4,87%) dan 4861 kasus non-stroke (95,13%) yang bisa dilihat pada Gambar 1. Hal ini menunjukkan bahwa dataset memiliki sebaran yang sangat tidak seimbang. Ketidakseimbangan yang signifikan ini merupakan karakteristik umum dalam dataset medis dan menjadi tantangan khusus dalam pengembangan model prediksi yang akurat. Untuk mengatasi ketidakseimbangan tersebut, penelitian ini menerapkan teknik SMOTE untuk menghasilkan sampel sintetis dari kelas minoritas sehingga dapat meningkatkan kinerja model dalam mendeteksi kasus stroke.



Gambar 1: Distribusi Kelas pada Dataset Stroke Prediction

Tabel 1: Deskripsi Fitur Dataset Stroke Prediction

Fitur	**Deskripsi**	**Tipe Data**
id	Identifier unik setiap pasien	Numerik
gender	Jenis kelamin pasien (*Male*, *Female*, *Other*)	Kategorikal
age	Usia pasien	Numerik
hypertension	0 jika pasien tidak memiliki hipertensi, 1 jika memiliki hipertensi	Biner
heart_disease	0 jika pasien tidak memiliki penyakit jantung, 1 jika memiliki penyakit jantung	Biner
ever_married	Status pernikahan (*Yes*, *No*)	Kategorikal
work_type	Tipe pekerjaan (*children*, *Govt_job*, *Never_worked*, *Private*, *Self-employed*)	Kategorikal
Residence_type	Tipe tempat tinggal (*Rural*, *Urban*)	Kategorikal
avg_glucose_level	Level glukosa rata-rata dalam darah (*Average Glucose Level*)	Numerik
bmi	Body Mass Index	Numerik
smoking_status	Status merokok (*formerly smoked*, *never smoked*, *smokes*, *Unknown*)	Kategorikal
stroke	1 jika pasien pernah stroke, 0 jika tidak	Biner

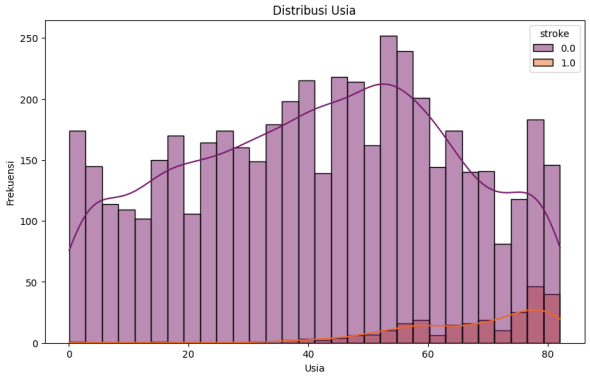
2.2 Data Preprocessing

Preprocessing data merupakan tahapan fundamental dalam pengembangan model *machine learning* yang bertujuan untuk meningkatkan kualitas dan kesiapan data sebelum proses pemodelan. Dalam penelitian ini, preprocessing data dilakukan melalui empat tahapan utama: pembersihan dan penyaringan data, penanganan nilai yang hilang, rekayasa fitur, dan encoding variabel kategorikal. Setiap tahapan dirancang secara sistematis untuk mengatasi berbagai tantangan dalam dataset, seperti ketidaklengkapan data, variasi format data, dan kebutuhan standardisasi untuk pemrosesan algoritmik.

2.2.1 *Data Cleaning and Filtering*

Proses preprocessing data diawali dengan analisis dan pembersihan data untuk memastikan kualitas serta keandalan data yang akan digunakan dalam pemodelan. Tahap pertama adalah eliminasi kolom identifikasi (id) dari dataset. Kolom ini dihapus karena bersifat unik untuk setiap pasien dan tidak memiliki kontribusi terhadap proses prediksi stroke. Keputusan ini diambil berdasarkan prinsip reduksi dimensi yang bertujuan meningkatkan efisiensi komputasi tanpa mengurangi informasi yang bermakna dalam dataset.

Selanjutnya, dilakukan penyaringan data berdasarkan kategori usia dewasa dengan membatasi subjek berusia 18 tahun ke atas. Pembatasan ini dilakukan berdasarkan pertimbangan epidemiologis yang menunjukkan bahwa pola dan faktor risiko stroke pada populasi dewasa memiliki karakteristik yang berbeda secara signifikan dibandingkan dengan populasi anak-anak. Sebagaimana ditunjukkan pada Gambar 2, fokus pada populasi dewasa tidak hanya meningkatkan homogenitas data tetapi juga meminimalkan potensi bias dalam prediksi, mengingat prevalensi stroke yang secara signifikan lebih tinggi pada kelompok usia dewasa.

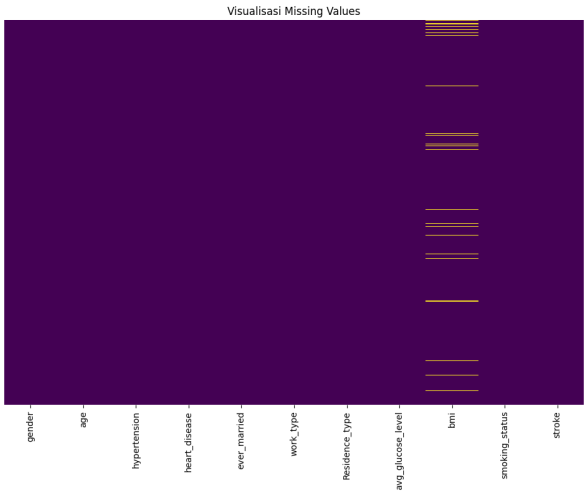


Gambar 2: Distribusi Usia pada Dataset Stroke Prediction

2.2.2 *Handling Missing Value*

Penanganan nilai yang hilang merupakan tahap kritis dalam proses preprocessing, terutama pada variabel BMI (Body Mass Index) sebagaimana tervisualisasi pada Gambar 3. Untuk mengatasi permasalahan ini, diterapkan metode K-Nearest Neighbors (KNN) Imputer dengan parameter `n_neighbors=5`. Pemilihan metode ini didasarkan pada kemampuannya dalam mempertimbangkan pola dan kemiripan karakteristik antara data saat melakukan estimasi nilai yang hilang.

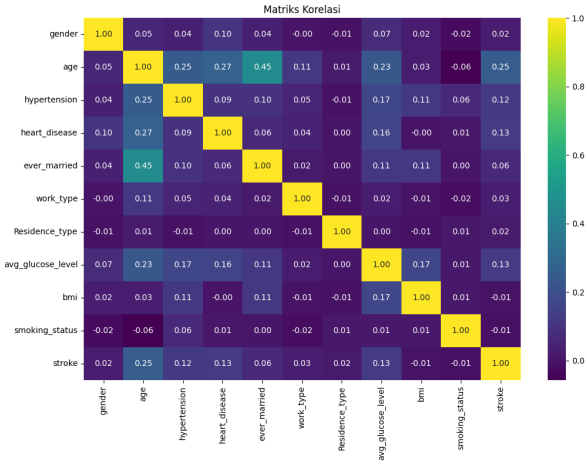
KNN Imputer menunjukkan keunggulan dibandingkan dengan metode imputasi konvensional seperti mean atau median, karena kemampuannya dalam menghasilkan estimasi yang lebih presisi dengan mempertimbangkan konteks dan interaksi antarvariabel dalam dataset. Metode ini bekerja berdasarkan prinsip bahwa sampel-sampel dengan karakteristik serupa cenderung memiliki nilai BMI yang mirip, sehingga menghasilkan estimasi yang lebih representatif dan mempertahankan integritas hubungan antarvariabel dalam data.



Gambar 3: Distribusi Nilai Hilang pada Variabel BMI

2.2.3 *Feature Analysis*

Tahap akhir preprocessing meliputi analisis mendalam terhadap hubungan antarvariabel melalui matriks korelasi yang ditampilkan pada Gambar 4. Analisis korelasi ini memberikan pemahaman komprehensif tentang kekuatan dan arah hubungan antarvariabel dalam dataset. Nilai korelasi yang berkisar antara -1 hingga 1 mengindikasikan tingkat dan sifat hubungan antarvariabel, dengan nilai mendekati 1 atau -1 menunjukkan korelasi yang kuat dan nilai mendekati 0 menandakan hubungan yang lemah.



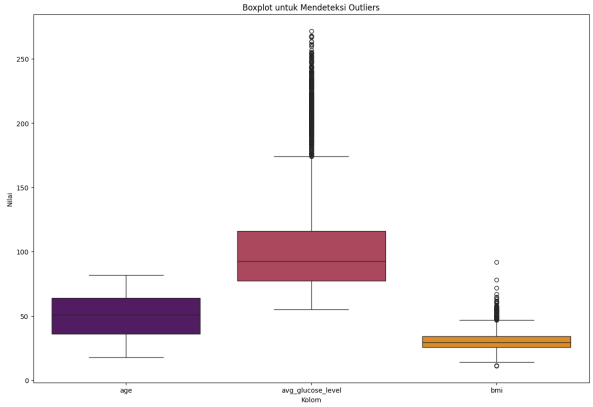
Gambar 4: Matriks Korelasi Variabel pada Dataset Stroke Prediction

Hasil analisis matriks korelasi menunjukkan bahwa faktor usia memiliki korelasi tertinggi dengan kejadian stroke, yakni sebesar 0,25. Meskipun nilai ini tergolong rendah, temuan ini mengonfirmasi bahwa usia merupakan faktor yang paling berpengaruh dibandingkan dengan variabel lainnya. Rendahnya nilai ko-

relasi secara keseluruhan mengindikasikan kompleksitas dalam memprediksi kejadian stroke dan menegaskan pentingnya pendekatan *machine learning* yang mampu menangkap pola-pola kompleks dalam data. Temuan ini juga memperkuat pemahaman bahwa kelompok usia lanjut merupakan segmen populasi yang memerlukan perhatian khusus dalam upaya pencegahan stroke.

2.2.4 *Outlier Analysis*

Analisis outlier dilakukan untuk mengidentifikasi nilai-nilai ekstrem dalam dataset, khususnya pada variabel numerik seperti usia, BMI, dan level glukosa rata-rata, sebagaimana ditunjukkan pada Gambar 5. Meskipun terdeteksi beberapa nilai yang secara statistik dapat dikategorikan sebagai outlier, keputusan diambil untuk mempertahankan nilai-nilai tersebut dalam dataset. Pertimbangan ini didasarkan pada dua alasan utama: pertama, dalam konteks data medis, nilai-nilai ekstrem seringkali merepresentasikan kasus-kasus klinis yang valid dan penting; kedua, outlier tersebut dapat mengandung informasi yang bermakna tentang faktor risiko stroke, mengingat kondisi ekstrem seperti kadar glukosa yang sangat tinggi atau BMI yang jauh di atas normal memang berkorelasi dengan peningkatan risiko stroke.



Gambar 5: Visualisasi Outlier pada Variabel Numerik

2.2.5 *Encoding*

Sebelum data dapat diproses oleh algoritma *machine learning*, seluruh variabel kategorikal dikonversi menjadi format numerik melalui proses Label Encoding. Metode ini diterapkan pada semua variabel kategorikal dalam dataset, termasuk *gender*, *ever_married*, *Residence_type*, *work_type*, dan *smoking_status*.

Label Encoding mengonversi setiap kategori menjadi nilai numerik berurutan. Sebagai contoh, untuk variabel *gender*, kategori "Female" dikonversi menjadi 0, "Male" menjadi 1, dan "Other" menjadi 2. Pendekatan serupa diterapkan pada variabel kategorikal lainnya, dengan setiap kategori unik diberikan nilai numerik yang berbeda. Meskipun pendekatan ini dapat menghasilkan asumsi ordinal implisit antarkategori, pemilihan Label Encoding untuk seluruh variabel kategorikal didasarkan pada pertimbangan efisiensi komputasi dan kesederhanaan model.

Namun untuk kategori "Other" yang ada pada variabel gender, kami menghapusnya karena nilai tersebut sangat tidak relevan dan hanya terdapat pada satu sampel saja.

2.2.6 *Data Splitting and K-Fold Cross Validation modified*

Dataset dibagi menjadi beberapa subset menggunakan metode stratifikasi untuk memastikan distribusi kelas yang seimbang pada setiap bagian data. Setelah preprocessing, dataset akhir terdiri dari 247 kasus stroke dan 4006 kasus non-stroke. Pemisahan data ini dirancang khusus untuk mencegah kebocoran data sintetis ke dalam set pengujian yang dapat mengakibatkan overfitting dan bias pada estimasi performa model.

Tabel 2: Jumlah Data dalam Setiap Fold

Fold	Stroke	Non-Stroke
Fold 1	49	801
Fold 2	49	801
Fold 3	49	801
Fold 4	50	801
Fold 5	50	802

Pembagian data untuk *training* dan *testing* dilakukan secara terpisah untuk kelas stroke dan non-stroke. Untuk kasus stroke, satu fold digunakan sebagai data *testing* (sekitar 49-50 sampel), sementara empat fold lainnya digabungkan sebagai data *training* (sekitar 197-198 sampel). Sedangkan untuk kasus non-stroke, dari satu fold yang sama dengan *testing* stroke, diambil 100 sampel secara acak sebagai data *testing*, dan sisa data dalam fold tersebut (sekitar 701-702 sampel) digunakan sebagai data *training*. Data non-stroke dari fold lainnya tidak digunakan dalam proses pemodelan untuk menghindari ketimpangan yang berlebihan dalam data *training*.

Proses validasi dilakukan dengan melakukan iterasi sebanyak 25 kali yang merupakan kombinasi dari 5 fold stroke dan 5 fold non-stroke yang berbeda. Setiap iterasi menggunakan kombinasi fold yang unik untuk memastikan evaluasi yang komprehensif terhadap performa model. Rincian kombinasi fold dan komposisi data untuk setiap iterasi dapat dilihat pada Tabel 3.

2.2.7 *Data Balancing*

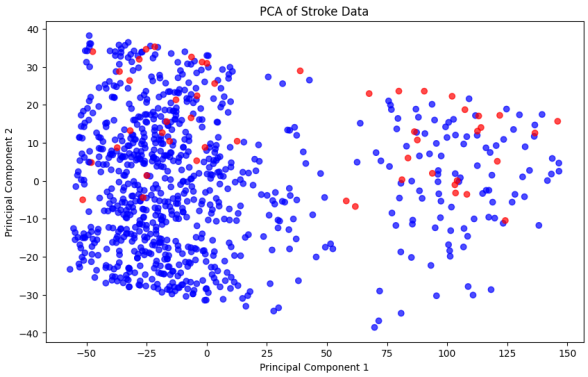
Untuk mengatasi ketidakseimbangan data yang signifikan antara kelas stroke (minoritas) dan non-stroke (mayoritas), penelitian ini menerapkan teknik SMOTE (Synthetic Minority Over-sampling Technique). SMOTE bekerja dengan menciptakan sampel sintetis dari kelas minoritas untuk menyeimbangkan distribusi kelas. Proses ini dilakukan secara sistematis den-

gan menggunakan interpolasi antara sampel-sampel kelas minoritas yang berdekatan dalam ruang fitur.

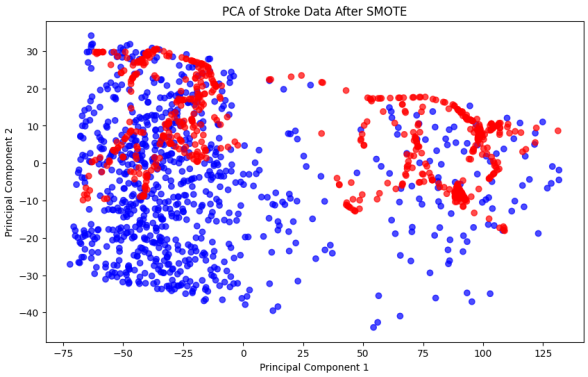
SMOTE menghasilkan sampel sintetis dengan mencari K tetangga terdekat dari kelas yang sama menggunakan jarak Euclidean, memilih secara acak salah satu dari K tetangga terdekat, menghitung vektor perbedaan, dan menghasilkan sampel sintetis menggunakan rumus $x_{new} = x_i + \lambda\delta$, dengan λ adalah bilangan acak antara 0 dan 1.

Dalam implementasinya, SMOTE diterapkan hanya pada data *training* untuk menghindari kebocoran data. Parameter yang digunakan adalah rasio oversampling 0.7 untuk meminimalisir risiko overfitting, jumlah tetangga terdekat 5, dan random state 42.

Setelah penerapan SMOTE, distribusi kelas dalam data *training* menjadi lebih seimbang, membantu model dalam mempelajari pola dari kedua kelas secara lebih efektif. Penting untuk dicatat bahwa data *testing* tetap mempertahankan distribusi aslinya untuk memastikan evaluasi yang realistis terhadap performa model.



Gambar 6: Distribusi Kelas Sebelum Penerapan SMOTE pada Data *training*



Gambar 7: Distribusi Kelas Setelah Penerapan SMOTE pada Data *training*

Tabel 3: Kombinasi Fold dan Komposisi Data pada Setiap Iterasi

Iterasi	Testing Fold		Data Testing		Data Training		Data Training with SMOTE	
	Stroke	Non-stroke	Stroke	Non-stroke	Stroke	Non-stroke	Stroke	Non-stroke
1	S1	N1	49	100	198	701	491	701
2	S1	N2	49	100	198	701	491	701
3	S1	N3	49	100	198	701	491	701
4	S1	N4	49	100	198	701	491	701
5	S1	N5	49	100	198	702	491	702
6	S2	N1	49	100	198	701	491	701
7	S2	N2	49	100	198	701	491	701
8	S2	N3	49	100	198	701	491	701
9	S2	N4	49	100	198	701	491	701
10	S2	N5	49	100	198	702	491	702
11	S3	N1	49	100	198	701	491	701
12	S3	N2	49	100	198	701	491	701
13	S3	N3	49	100	198	701	491	701
14	S3	N4	49	100	198	701	491	701
15	S3	N5	49	100	198	702	491	702
16	S4	N1	50	100	197	701	490	701
17	S4	N2	50	100	197	701	490	701
18	S4	N3	50	100	197	701	490	701
19	S4	N4	50	100	197	701	490	701
20	S4	N5	50	100	197	702	490	702
21	S5	N1	50	100	197	701	490	701
22	S5	N2	50	100	197	701	490	701
23	S5	N3	50	100	197	701	490	701
24	S5	N4	50	100	197	701	490	701
25	S5	N5	50	100	197	702	490	702

2.3 Machine Learning Classification Methods

Pengembangan model prediksi stroke dalam penelitian ini menggunakan dua algoritma *machine learning* yang berbeda, yaitu Random Forest dan Support Vector Machine. Pemilihan kedua algoritma ini didasarkan pada keunggulan masing-masing dalam menangani data medis yang kompleks serta kemampuannya dalam menghasilkan model prediksi yang akurat dan interpretabel[11].

2.3.1 Random Forest

Random Forest merupakan algoritma *machine learning* ensemble yang mengombinasikan kekuatan dari multiple pohon keputusan untuk menghasilkan prediksi yang lebih akurat dan stabil[11]. Algoritma ini dipilih karena beberapa keunggulan: (1) kemampuannya menangani data dengan dimensi tinggi, (2) ketahanannya terhadap overfitting, dan (3) kemampuannya memberikan informasi tentang pentingnya setiap fitur dalam proses prediksi. Setiap pohon keputusan dalam Random Forest dibangun menggunakan subset acak dari data pelatihan melalui teknik bootstrap aggregating (bagging), yang meningkatkan variasi antar pohon dan mengurangi variance prediksi akhir.

Random Forest merupakan algoritma *machine learning* ensemble yang terdiri atas kumpulan pohon keputusan. Setiap pohon keputusan dibangun menggunakan subset acak dari data pelatihan melalui teknik bootstrap aggregating (bagging). Untuk suatu dataset dengan n sampel dan m fitur, Random Forest menggunakan formula berikut untuk membangun setiap pohon:

$$f_{RF}(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x) \tag{1}$$

dengan $f_{RF}(x)$ adalah prediksi akhir Random Forest, B adalah jumlah pohon, dan $T_b(x)$ adalah prediksi dari pohon ke-b. Setiap pohon dibangun dengan mempertimbangkan subset fitur m_{try} yang dipilih secara acak, dengan:

$$m_{try} = \lfloor \sqrt{m} \rfloor \tag{2}$$

Pemilihan fitur pada setiap node menggunakan kriteria Gini Impurity:

$$Gini(t) = 1 - \sum_{i=1}^{c} p(i|t)^2 \tag{3}$$

dengan $p(i|t)$ adalah proporsi kelas i pada node t, dan c adalah jumlah kelas.

Dalam implementasinya, ruang pencarian hiperparameter Random Forest mencakup beberapa parameter kunci: jumlah pohon (`n_estimators`) dengan rentang 10 hingga 200, kedalaman maksimum pohon (`max_depth`) dari 1 hingga 50, jumlah minimum sampel untuk pemisahan internal (`min_samples_split`) dari 2 hingga 20, jumlah minimum sampel pada setiap daun (`min_samples_leaf`) dari 1 hingga 20, dan proporsi fitur yang digunakan (`max_features`) dengan rentang 0,1 hingga 1,0 yang didistribusikan secara seragam.

2.3.2 Support Vector Machine

Support Vector Machine (SVM) dipilih karena kemampuannya yang telah terbukti dalam berbagai aplikasi medis, khususnya dalam prediksi penyakit[12]. Dalam penelitian ini, digunakan SVM linear yang bekerja dengan mencari hyperplane optimal yang memaksimalkan margin pemisahan antara kelas stroke dan non-stroke. Menurut kajian terbaru[13], pemilihan SVM linear memiliki keunggulan dalam hal interpretabilitas model dan efisiensi komputasi, terutama untuk dataset medis dengan fitur yang telah diseleksi dengan baik.

Support Vector Machine linear bekerja dengan mencari hyperplane optimal yang memisahkan data ke dalam dua kelas. Fungsi objektif SVM linear dapat diformulasikan sebagai berikut:

$$\min_{\mathbf{w},b,\xi} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \tag{4}$$

dengan kendala:

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \tag{5}$$

Dalam persamaan tersebut, $\mathbf{w}$ merupakan vektor bobot yang menentukan orientasi hyperplane, b adalah bias, ξ_i merupakan variabel slack yang memungkinkan kesalahan klasifikasi, dan C merupakan parameter regularisasi yang mengontrol keseimbangan antara margin maksimum dan error klasifikasi.

Ruang pencarian hiperparameter SVM meliputi parameter regularisasi C dengan rentang dari 10^{-6} hingga 10^6 yang didistribusikan secara log-uniform, parameter `gamma` untuk kernel RBF dengan rentang dari 10^{-6} hingga 10^1 yang juga didistribusikan secara log-uniform, serta penggunaan kernel RBF yang telah ditetapkan. Distribusi log-uniform dipilih untuk parameter C dan `gamma` karena rentang nilai yang mencakup beberapa tingkat besaran, memungkinkan eksplorasi yang lebih efektif pada skala logaritmik.

2.4 Model Optimization

Optimasi model dilaksanakan menggunakan Optimasi Bayesian, suatu pendekatan optimasi berurutan yang memanfaatkan model probabilistik untuk memetakan fungsi objektif dan menentukan parameter optimal. Dalam penelitian ini, fungsi objektif yang dimaksud

adalah nilai F1-score yang dihasilkan model sebagai fungsi dari kombinasi hiperparameter yang digunakan.

Optimasi Bayesian menerapkan dua komponen utama dalam prosesnya. Komponen pertama adalah model pengganti (*surrogate model*) yang menggunakan Gaussian Process untuk memetakan hubungan antara hiperparameter dan performa model. Komponen kedua adalah fungsi akuisisi yang menentukan titik evaluasi berikutnya dengan menyeimbangkan antara eksplorasi ruang parameter yang belum dijelajahi dan eksploitasi area yang menunjukkan hasil menjanjikan.

Proses optimasi dilaksanakan secara sistematis dan berurutan. Diawali dengan pembangunan model probabilistik berdasarkan data hasil evaluasi sebelumnya, dilanjutkan dengan penentuan titik evaluasi berikutnya menggunakan fungsi akuisisi. Setelah itu, dilakukan evaluasi pada kombinasi parameter yang dipilih, dan model probabilistik diperbarui dengan hasil evaluasi terbaru. Proses ini diulang hingga mencapai jumlah iterasi yang telah ditetapkan.

Implementasi optimasi menggunakan BayesSearchCV dengan pengaturan yang telah disesuaikan untuk kasus ini. Jumlah iterasi pencarian ditetapkan sebanyak 32 kali, dengan setiap iterasi mengevaluasi kombinasi hiperparameter yang berbeda. *cross validation* menggunakan 3-lipatan untuk memperoleh estimasi performa yang lebih andal. Untuk memastikan hasil yang dapat direproduksi, nilai `random_state` ditetapkan pada 42, sementara penggunaan sumber daya komputasi dioptimalkan dengan mengatur `n_jobs` ke -1.

Pendekatan ini menghasilkan proses pencarian hiperparameter yang lebih efisien dibandingkan dengan metode Grid Search atau Random Search konvensional. Keunggulan ini diperoleh karena kemampuannya memanfaatkan informasi dari evaluasi sebelumnya untuk mengarahkan pencarian ke area yang lebih menjanjikan, sehingga menghasilkan kombinasi parameter yang optimal dengan lebih cepat dan efektif.

2.5 Performance Metrics

Dalam evaluasi model klasifikasi stroke, kami menggunakan beberapa metrik performa standar yang diperoleh dari *confusion matrix*.

Accuracy mengukur proporsi total prediksi yang benar dibandingkan dengan semua kasus.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

Precision mengukur proporsi prediksi positif yang benar-benar positif.

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

Recall (Sensitivity) mengukur proporsi kasus positif yang berhasil diidentifikasi.

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

F1-Score merupakan rata-rata harmonik dari *precision* dan *recall*.

$$\text{F1-Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9}$$

Dalam konteks ini, TP (*True Positive*) adalah jumlah kasus stroke yang diprediksi benar sebagai stroke. TN (*True Negative*) merupakan jumlah kasus non-stroke yang diprediksi benar sebagai non-stroke. FP (*False Positive*) menunjukkan jumlah kasus non-stroke yang salah diprediksi sebagai stroke. FN (*False Negative*) adalah jumlah kasus stroke yang salah diprediksi sebagai non-stroke.

Pemilihan metrik-metrik ini didasarkan pada karakteristik dataset yang tidak seimbang, sehingga F1-score menjadi sangat penting karena memberikan gambaran yang lebih baik tentang performa model pada kasus yang tidak seimbang dibandingkan dengan *accuracy* saja. *Recall* juga penting dalam kasus prediksi stroke karena fokus utama adalah pada identifikasi kasus positif (stroke) yang sebenarnya.

2.6 Modified K-Fold Cross Validation

Untuk mengatasi ketidakseimbangan data dan memastikan evaluasi model yang akurat, penelitian ini menggunakan validasi silang K-Fold yang dimodifikasi. Teknik ini membagi dataset menjadi K subset dengan distribusi kelas yang seimbang di setiap fold, sehingga performa model pada kelas minoritas dapat dievaluasi secara lebih representatif.

3. Results and Discussion

Hasil dari validasi silang K-Fold yang dimodifikasi menunjukkan bahwa model SVM dengan SMOTE dan Optimasi Bayesian mencapai performa tertinggi secara konsisten di seluruh fold, mengindikasikan kemampuan generalisasi yang baik.

4. Conclusion

Penerapan validasi silang K-Fold yang dimodifikasi terbukti efektif dalam evaluasi model pada dataset yang tidak seimbang, meningkatkan akurasi dan keandalan prediksi risiko stroke.

Acknowledgements

References

[1] World Health Organization and Dr. Poonam Khetrapal Singh, WHO Regional Director for South-East Asia, "World stroke day," 2021. Accessed: 27 November 2024.

[2] Kementerian Kesehatan RI, "Laporan nasional RISKESDAS 2023," technical report, Badan Penelitian dan Pengembangan Kesehatan, Jakarta, Indonesia, 2023. Riset Kesehatan Dasar 2023.

[3] A. K. Boehme, C. Esenwa, and M. S. Elkind, "Stroke risk factors, genetics, and prevention," *Circulation research*, vol. 120, no. 3, pp. 472–495, 2017.

[4] V. L. Feigin, M. Brainin, B. Norrving, P. B. Gorelick, M. Dichgans, W. Wang, J. D. Pandian, S. C. Martins, M. O. Owolabi, and D. A. Wood, "Prevention of stroke: a global perspective," *The Lancet Neurology*, vol. 21, no. 7, pp. 608–617, 2022.

[5] M. Guhdar, A. I. Melhum, and A. L. Ibrahim, "Optimizing accuracy of stroke prediction using logistic regression," *Journal of Technology and Informatics (JoTI)*, vol. 4, no. 2, pp. 41–47, 2023.

[6] S. Dev, H. Wang, C. S. Nwosu, N. Jain, B. Veeravalli, and D. John, "A predictive analytics approach for stroke prediction using machine learning and neural networks," *Healthcare Analytics*, vol. 2, p. 100032, 2022.

[7] Y. Yang, J. Zheng, Z. Du, Y. Li, Y. Cai, *et al.*, "Accurate prediction of stroke for hypertensive patients based on medical big data and machine learning algorithms: retrospective study," *JMIR Medical Informatics*, vol. 9, no. 11, p. e30277, 2021.

[8] S. Zhi, X. Hu, Y. Ding, H. Chen, X. Li, Y. Tao, and W. Li, "An exploration on the machine-learning-based stroke prediction model," *Frontiers in Neurology*, vol. 15, p. 1372431, 2024.

[9] M. S. Azam, M. Habibullah, and H. K. Rana, "Performance analysis of various machine learning approaches in stroke prediction," *International Journal of Computer Applications*, vol. 175, no. 21, pp. 11–15, 2020.

[10] F. Soriano, "Stroke prediction dataset," 2024. Accessed: 15 Oktober 2024.

[11] H. Zhang, X. Wang, and P. Ma, "Random forest for high-dimensional data: Recent advances and future directions," *WIREs Data Mining and Knowledge Discovery*, vol. 13, no. 3, p. e1472, 2023.

[12] D. A. Pisner and D. M. Schnyer, "Support vector machine," in *Machine learning*, pp. 101–121, Elsevier, 2020.

[13] K. A. Bhavsar, A. Abugabah, J. Singla, A. A. AlZubi, A. K. Bashir, *et al.*, "A comprehensive review on medical diagnosis using machine learning," *Computers, Materials and Continua*, vol. 67, no. 2, p. 1997, 2021.