OPTIMASI ALGORITMA SUPPORT VECTOR MACHINES(SVM) MENGGUNAKAN ADAPTIVE BOOSTING(ADABOOST) UNTUK MENINGKATKAN AKURASI PREDIKSI PENYAKIT BRAIN STROKE

Azka Maulana, Sarwido, Adi Sucipto
Teknik Informatika, Universitas Islam Nahdlatul Ulama Jepara, Jepara
Jl. Taman Siswa, Pekeng, Kauman, Tahunan, Kec. Tahunan, Kabupaten Jepara, Jawa Tengah
azka32232@gmail.com

ABSTRAK

Stroke merupakan salah satu penyakit kronis dengan tingkat kematian dan kecacatan yang tinggi, sehingga deteksi dini sangat diperlukan untuk meminimalkan risiko yang ditimbulkan. Penelitian ini bertujuan untuk meningkatkan akurasi prediksi penyakit stroke dengan mengoptimalkan algoritma Support Vector Machines (SVM) menggunakan pendekatan Adaptive Boosting (AdaBoost). Algoritma SVM dikenal efektif untuk klasifikasi data non-linear, namun memiliki keterbatasan saat menangani data yang tidak seimbang. Oleh karena itu, dalam penelitian ini dilakukan preprocessing data meliputi imputasi data hilang, encoding variabel kategorikal, normalisasi, serta penyeimbangan data menggunakan teknik undersampling ClusterCentroids. Dataset yang digunakan diperoleh dari situs Kaggle dengan jumlah 2000 entri dan 12 fitur. Model SVM tanpa optimasi menghasilkan akurasi sebesar 76%, dengan nilai precision, recall, dan f1-score yang seimbang pada kedua kelas. Setelah dioptimasi dengan AdaBoost, performa model meningkat secara signifikan dengan akurasi mencapai 87% serta peningkatan pada nilai precision dan recall, khususnya dalam mendeteksi kelas stroke. Hasil penelitian ini menunjukkan bahwa kombinasi SVM dan AdaBoost mampu mengatasi ketidakseimbangan data dan meningkatkan akurasi prediksi, sehingga metode ini berpotensi diterapkan dalam sistem pendukung keputusan untuk deteksi dini penyakit stroke secara lebih akurat dan andal.

Kata kunci : *Stroke, Support Vector Machines, Adaptive Boosting, Prediksi Penyakit, Undersampling, Machine Learning*

1. PENDAHULUAN

Menjaga kesehatan adalah hal yang sangat penting dalam kehidupan manusia, namun sering kali kurang diperhatikan hingga akhirnya muncul penyakit secara mendadak[1]. Penyakit degeneratif, yaitu gangguan kesehatan yang terjadi akibat menurunnya fungsi organ tubuh, sering kali dikaitkan dengan gaya hidup modern dan tekanan psikologis yang tinggi. Situasi ini dapat menyebabkan munculnya penyakit serius seperti penyakit jantung, stroke, diabetes, hipertensi, dan sebagainya. Beberapa faktor yang berperan besar dalam hal ini antara lain kebiasaan merokok, konsumsi minuman beralkohol, pola makan yang tidak sehat, kurangnya aktivitas fisik, dan stres berlebihan, serta paparan terhadap polusi lingkungan[2].

Penyakit ini dikenal sebagai penyakit kronis, atau *chronic disease*, merupakan jenis penyakit yang berkembang secara perlahan dan berlangsung dalam jangka waktu panjang. Kondisi ini dapat berdampak signifikan terhadap kualitas hidup serta menurunkan tingkat produktivitas seseorang[3]. Perubahan sosial ekonomi dan preferensi makanan yang semakin menjauh dari prinsip gizi seimbang memberi dampak buruk terhadap kesehatan. Asupan makanan tinggi lemak dan gula, serta minim serat dan mikronutrien, dapat memicu masalah seperti obesitas, kelebihan gizi, dan peningkatan radikal bebas. Hal ini turut mendorong pergeseran pola penyakit dari yang bersifat infeksi menjadi penyakit kronis tidak menular, seperti stroke[4].

Stroke adalah penyakit yang sering kali berujung pada kecacatan permanen atau kematian. Berdasarkan data dari Organisasi Stroke Dunia, setiap tahun tercatat sekitar 13,7 juta kasus stroke di seluruh dunia, dengan 5,5 juta di antaranya berakhir fatal. Secara global, stroke menempati posisi ketiga sebagai faktor penyebab kematian tertinggi, berada di bawah penyakit jantung koroner dan kanker [5]. Stroke memiliki risiko kematian yang tinggi dan dapat menimbulkan berbagai komplikasi, seperti hilangnya kemampuan berbicara atau melihat, kelumpuhan, dan kebingungan. Penyakit ini disebut stroke karena serangannya yang datang secara mendadak. Selain itu, risiko terjadinya serangan stroke kedua juga meningkat secara signifikan pada mereka yang pernah mengalaminya sebelumnya [6].

Support Vector Machine (SVM) dipilih dalam penelitian ini karena merupakan salah satu algoritma klasifikasi yang efektif dalam menangani data berdimensi tinggi dan memiliki kompleksitas hubungan antar variable. Kemampuannya dalam membentuk pemisah kelas yang optimal menjadikannya unggul, terutama pada data medis yang umumnya memiliki struktur kompleks dan jumlah yang terbatas. Selain itu, SVM juga memiliki ketahanan terhadap overfitting, sehingga tetap dapat menghasilkan model yang generalisasi dengan baik. Dengan kelebihan tersebut, SVM berpotensial untuk digunakan dalam membangun model prediktif yang mampu mengidentifikasi risiko penyakit stroke secara lebih akurat dan andal.

Penelitian yang berjudul "perbandingan algoritma data mining untuk prediksi penyakit stroke" menunjukkan bahwa algoritma Support Vector Machine (SVM) dengan data yang telah diseimbangkan mampu mencapai akurasi sebesar 76,52%, precision 74,47%, recall 76,12%, dan f-measure 75,29 [7]. Meskipun kinerja tersebut tergolong baik, namun masih terdapat keterbatasan, performanya yang masih dipengaruhi oleh kompleksitas data, pemilihan parameter, serta teknik penyeimbangan data yang kurang efektif yang berdampak pada penurunan akurasi model. Untuk mengatasi permasalahan tersebut, pendekatan ensemble learning seperti Adaptive Boosting (AdaBoost) dapat diterapkan guna mengoptimalkan performa SVM. AdaBoost bekerja dengan meningkatkan bobot pada kesalahan prediksi, sehingga model menjadi lebih fokus pada data yang sulit diklasifikasikan secara benar. Dengan demikian, kombinasi antara SVM dan AdaBoost diharapkan dapat meningkatkan akurasi prediksi serta mendukung pengembangan sistem deteksi dini penyakit stroke yang lebih efektif dan akurat.

Tujuan penelitian ini adalah menerapkan algoritma Adaptive Boosting (AdaBoost) pada algoritma Support Vector Machines (SVM) untuk mengoptimalkan performa algoritma SVM dalam memprediksi penyakit stroke otak. Optimalisasi ini ditujukan untuk meningkatkan kemampuan model dalam melakukan klasifikasi secara lebih akurat melalui pendekatan ensemble learning. Dalam penelitian ini, AdaBoost berfungsi memperbaiki kelemahan model SVM dengan cara menggabungkan beberapa model pembelajar lemah menjadi satu model prediktif yang lebih kuat. Mekanisme pembobotan yang diterapkan oleh AdaBoost membuat model lebih fokus pada sampel data yang sebelumnya sulit diklasifikasikan dengan benar, sehingga proses pelatihan menjadi lebih adaptif terhadap kesalahan. Dengan demikian, integrasi AdaBoost diharapkan dapat meningkatkan akurasi, sensitivitas, dan kemampuan generalisasi SVM dalam mengidentifikasi risiko stroke otak berdasarkan karakteristik data medis yang tersedia.

2. TINJAUAN PUSTAKA

2.1. Stroke

Stroke yaitu gangguan kesehatan serius yang dapat menyebabkan dampak jangka panjang dan dikenal sebagai salah satu penyebab kematian terbanyak. Risiko stroke semakin tinggi jika penderita juga mengalami masalah jantung, karena hal tersebut dapat memperburuk kondisi. Penyakit stroke dapat terjadi akibat terganggunya proses aliran darah menuju otak, yang dapat menyebabkan penurunan fungsi sistem saraf pusat (SSP) [8].

2.2. Data Mining

Data mining merupakan proses yang melibatkan berbagai teknik untuk menemukan informasi penting yang tersembunyi di dalam sekumpulan data[7]. Tujuan utama dari data mining yaitu untuk mengidentifikasi informasi baru yang bernilai dan mendukung pengambilan keputusan yang lebih tepat [9].

2.3. Python

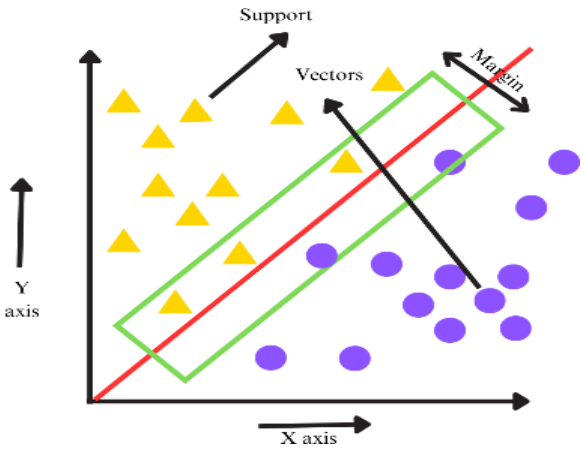
Python adalah bahasa pemrograman tingkat tinggi yang dikembangkan oleh Guido Van Rossum dan resmi dirilis pada tahun 1991. Hingga kini, Python telah menjadi salah satu bahasa yang paling diminati karena kemampuannya yang fleksibel serta aplikatif di berbagai bidang. Bahasa ini banyak digunakan dalam pengembangan teknologi seperti *Machine Learning*, *Deep Learning*, dan *data mining* [10].

2.4. Algoritma Ensemble Learning Adaptive Boosting (AdaBoost)

Adaptive Boosting (Adaboost) merupakan algoritma klasifikasi yang bekerja dengan menggabungkan beberapa klasifier lemah menjadi satu klasifier kuat untuk meningkatkan akurasi prediksi. Algoritma ini bertujuan untuk memfokuskan perhatian lebih pada data-data yang sulit diklasifikasikan dengan benar, yaitu dengan menaikkan bobotnya, sementara bobot data yang sudah diklasifikasikan dengan benar akan dikurangi [12].

2.5. Algoritma Support Vector Machines (SVM)

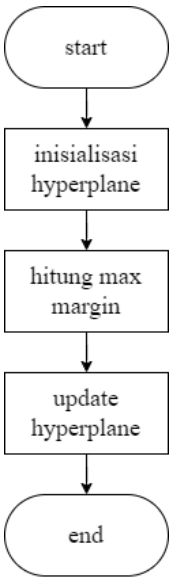
Support Vector Machines (SVM) adalah algoritma yang digunakan untuk mengklasifikasikan data, baik yang memiliki pola linear maupun non-linear. Algoritma ini mampu bekerja secara efektif pada data berpola non-linear dengan memanfaatkan konsep kernel, yang berfungsi untuk mentransformasikan data ke dalam ruang berdimensi lebih tinggi. Penjelasan rumus SVM dapat dilihat pada bagian berikut [11].



Gambar 1. Gambar Algoritma Svm

$$f(x) = \text{sign}\left(\sum_{i=1}^{n} a_i\ y_i\ K(x_i,\ x) + b\right) \quad (1)$$

(x) : data yang akan diklasifikasikan
x_i : data latih
y_i : label kelas
a_i : bobot
$K(x_i, x)$: kernel yang menghitung jarak antara x_i dan x.
b = bias



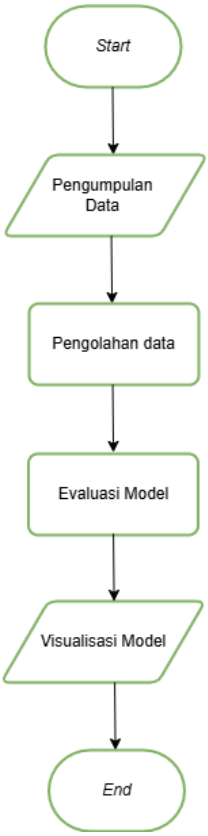
Gambar 2. Gambar Flowchart Metode Svm

Flowchart tersebut menggambarkan proses pelatihan algoritma Support Vector Machine (SVM) secara iteratif yang dimulai dengan inisialisasi hyperplane sebagai batas pemisah awal antar kelas. Selanjutnya, dilakukan perhitungan margin maksimum antara hyperplane dan data terdekat dari masing-masing kelas (support vectors) untuk memperoleh pemisahan yang optimal. Berdasarkan hasil tersebut, hyperplane diperbarui secara bertahap hingga mencapai konvergensi, yaitu kondisi terbaik di mana perubahan tidak lagi signifikan. Proses diakhiri ketika model telah mencapai konfigurasi optimal.

3. METODE PENELITIAN

Jenis penelitian yang dilakukan dalam studi ini adalah penelitian eksperimental dengan pendekatan kuantitatif. Penelitian eksperimental bertujuan untuk mengidentifikasi dampak suatu variabel terhadap variabel lainnya dalam kondisi yang dikendalikan secara ketat [13].

Dalam pendekatan kuantitatif, terdapat beberapa jenis metode yang umum digunakan, di antaranya adalah metode eksperimen dan metode survei, yang bertujuan untuk mengumpulkan data secara objektif dan terukur. Sebaliknya, pendekatan kualitatif cenderung menggunakan metode naturalistik, yang menekankan pada pengamatan fenomena dalam konteks alami tanpa intervensi langsung dari peneliti. Metode eksperimen sendiri diarahkan untuk menguji pengaruh suatu variabel independen terhadap variabel dependen, dengan pelaksanaan dalam kondisi yang dikendalikan secara ketat, guna memastikan validitas hubungan kausal antarvariabel yang diteliti [13]. Adapun skema penelitian adalah seperti yang ditunjukan pada Gambar 3.



Gambar 3. Gambar Skema Penelitian

Tahap pertama dalam penelitian ini adalah pengumpulan data, pengolahan data, evaluasi model, dan visualisasi model. Berikut penjelasan mengenai tahapan skema penelitian.

3.1. Pengumpulan data

Studi literatur dalam penelitian ini dimanfaatkan sebagai metode untuk mengumpulkan, menganalisis, dan merangkum informasi dari berbagai sumber literatur yang relevan dengan topik yang dikaji. Proses ini dilakukan dengan mengacu pada referensi terpercaya, seperti jurnal ilmiah, buku, artikel, serta sumber digital lainnya yang berkaitan.

3.2. Pengolahan Data

Pengolahan data dalam penelitian ini dilakukan menggunakan teknik preprocessing data dengan melalui beberapa tahap sebagai betikut :

a. Imputasi
   Proses pengisian nilai yang hilang (missing value) dalam dataset dilakukan dengan mengganti nilai kosong tersebut menggunakan nilai tertentu, seperti rata-rata, median, modus, atau dengan menggunakan prediksi yang dihasilkan oleh model.

b. Encoding
Proses konversi data kategorikal ke dalam format numerik dilakukan agar data tersebut dapat diproses oleh algoritma machine learning, seperti Support Vector Machines (SVM), yang hanya dapat bekerja dengan data dalam bentuk angka. Fitur kategorikal seperti "gender" (laki-laki, perempuan) atau "smoking_status" (tidak merokok, merokok sebelumnya, merokok) perlu diubah menjadi angka agar model dapat memahaminya dan menggunakannya dalam analisis.

c. Scaling
Proses ini bertujuan untuk menyesuaikan rentang nilai fitur numerik agar memiliki skala yang konsisten. Tujuannya adalah agar semua fitur memberikan kontribusi yang seimbang saat digunakan dalam algoritma machine learning, terutama algoritma yang sensitif terhadap jarak antar data, seperti Support Vector Machines (SVM).

d. Blancing
Proses ini bertujuan untuk menyeimbangkan jumlah data yang berasal dari kelas dominan dan kelas yang jumlahnya lebih sedikit dalam suatu dataset, terutama ketika terdapat ketidak seimbangan yang signifikan. Pada penelitian ini, metode yang digunakan untuk blancing adalah metode undersampling, yang mengurangi jumlah data pada kelas mayoritas agar seimbang dengan kelas minoritas.

e. Train-Test Split
Proses ini mencakup pembagian dataset menjadi dua bagian utama: data latih (training data) dan data uji (testing data). Dataset latih dipergunakan untuk melatih metode machine learning agar dapat mengenali pola serta hubungan dalam data, sedangkan dataset uji dipakai untuk menguji kinerja model pada dataset yang belum pernah dilihat sebelumnya.

f. Model Building
Proses ini melibatkan pembangunan dan pelatihan model machine learning dengan tujuan agar model dapat mempelajari pola dari data pelatihan (training data) dan kemudian mampu menghasilkan prediksi untuk data yang belum diketahui.

3.3. Confusion Matrix

Confusion Matrix adalah salah satu metode evaluasi yang digunakan untuk mengukur performa algoritma klasifikasi, baik untuk kasus dua kelas maupun multiclass. Metode ini disajikan dalam bentuk tabel yang menggambarkan empat kemungkinan hasil berdasarkan perbandingan antara prediksi model dan data sebenarnya. Tabel confusion matrix ditunjukan pada tabel 1 [12].

Tabel 1. Tabel Confusion Matrix

		True Valse	
		True	False
Prediction	True	TP Correct result	FP Unexpected result
	False	FN Missing result	TN Correct absence of result

Empat komponen utama dalam confusion matrix mencakup: True Positive (TP), yaitu kondisi ketika data positif berhasil diprediksi dengan benar; True Negative (TN), ketika data negatif juga berhasil diklasifikasikan secara tepat; False Positive (FP), yaitu kesalahan saat data negatif diklasifikasikan sebagai positif; serta False Negative (FN), saat data positif justru diprediksi sebagai negatif. Dari hasil proses klasifikasi pada confusion matrix, dihitung tiap nilai dengan accuracy, precision, recall, dan F-1 score [14].

a. Accuracy, menggambarkan seberapa akurat model yang digunakan dalam klasifikasi.
Rumus:

$$accuracy = \left(\frac{TP+TN}{TP+FP+TN+FN}\right) \times 100\% \quad (2)$$

b. Precision, menggambarkan akurasi antara data yang diminta dengan hasil prediksi yang diberikan oleh model.
Rumus:

$$precision = \left(\frac{TP}{TP+FP}\right) \times 100\% \quad (3)$$

c. Recall, menggambarkan keberhasilan model dalam menemukan kembali informasi.
Rumus:

$$recall = \left(\frac{TP}{TP+FN}\right) \times 100\% \quad (4)$$

d. F-1 score, menggambarkan perbandingan rata-rata precision dan recall yang dibobotkan.
Rumus:

$$f1 - score = \left(\frac{2 \times precision \times recall}{(precision+recall)}\right) \times 100\% \quad (5)$$

4. HASIL DAN PEMBAHASAN

4.1. Pengumpulan data

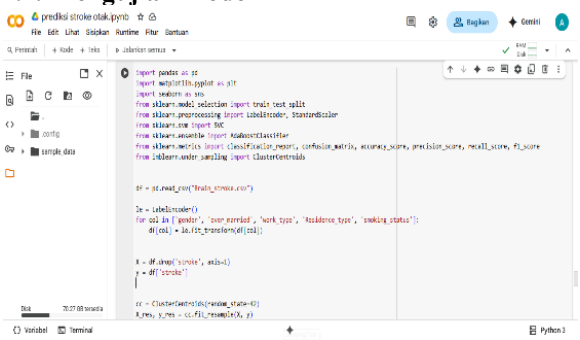
Dataset dalam penelitian ini dikumpulkan menggunakan metode studi literatur yang diperoleh dari situs Kaggle.com dengan judul "Brain Stroke Prediction Dataset". Data ini memuat catatan medis pasien yang meliputi sejumlah atribut penting seperti umur, jenis kelamin, tekanan darah, kadar glukosa, indeks massa tubuh (BMI), serta kebiasaan merokok. Dengan jumlah 12 variabel dan 2000 dataset. Lalu diolah dan analisis menggunakan berbagai tools dalam data science. Data yang diperoleh dapat dilihat pada tabel 2 berikut:

Tabel 2. Tabel Dataset Stroke

gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
Male	67	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
Male	80	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
Female	49	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
Female	79	1	0	Yes	Self-employed	Rural	174.12	24	never smoked	1

Dari Tabel 2 dapat dilihat bahwa, setiap entri data merepresentasikan karakteristik individu pasien yang terdiri atas sejumlah variabel independen, antara lain jenis kelamin, usia, riwayat hipertensi dan penyakit jantung, status pernikahan, jenis pekerjaan, jenis tempat tinggal, kadar glukosa darah rata-rata (*average glucose level*), indeks massa tubuh (*Body Mass Index*), serta status merokok. Kolom *stroke* berfungsi sebagai variabel dependen atau label klasifikasi, dengan nilai 1 yang mengindikasikan bahwa pasien telah mengalami stroke. Data tersebut merefleksikan keberagaman karakteristik demografis dan klinis yang relevan untuk mendukung proses pelatihan dan pengujian model prediktif dalam konteks identifikasi risiko stroke secara dini.

4.2. Pengujian Model



Gambar 4. Gambar Pengujian Model menggunakan Python

Dari gambar 4 dapat dilihat bahwa, pada penelitian ini model diuji menggunakan bahasa pemrograman python dengan menggunakan beberapa library machine learning yang tersedia pada pemrograman python. Pengujian dilakukan menggunakan tipe data csv lalu data diambil dan diolah menggunakan teknik undersampling sebagai teknik penyeimbang data, preprocesing sebagai teknik analisis atau pengolahan data dan pengujian model.

4.3. Hasil Prediksi Svm Tanpa Adaboost

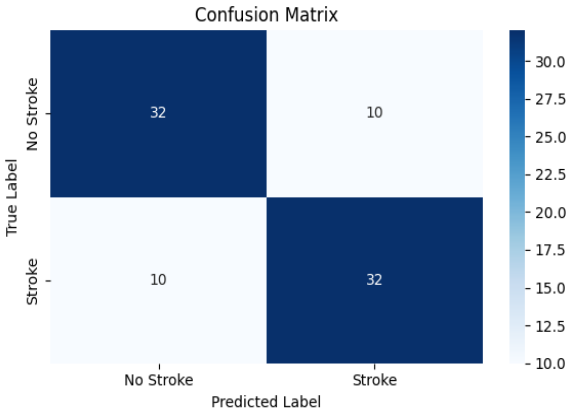
Berdasarkan hasil evaluasi, model Support Vector Machine (SVM) tanpa penerapan metode boosting menunjukkan kinerja yang baik dengan nilai precision, recall, dan f1-score masing-masing sebesar 0,76 untuk kedua kelas, yaitu kelas positif (stroke) dan negatif (non-stroke). Akurasi keseluruhan yang diperoleh sebesar 76% mengindikasikan bahwa model mampu melakukan klasifikasi dengan tingkat ketepatan yang moderat terhadap data uji. Nilai macro average dan weighted average yang identik menunjukkan bahwa distribusi performa model terhadap kedua kelas relatif seimbang, yang selaras dengan penerapan teknik undersampling untuk mengatasi ketidakseimbangan kelas. Meskipun demikian, performa model masih dapat ditingkatkan melalui penerapan algoritma ensembel seperti Adaptive Boosting (AdaBoost) guna memperkuat kemampuan generalisasi model dalam melakukan prediksi.

```
Classification Report:
              precision    recall  f1-score   support

           0       0.76      0.76      0.76        42
           1       0.76      0.76      0.76        42

    accuracy                           0.76        84
   macro avg       0.76      0.76      0.76        84
weighted avg       0.76      0.76      0.76        84
```

Gambar 5. Gambar Prediksi Svm



Gambar 6. Gambar Confuison Matrix Svm

Gambar tersebut merepresentasikan *confusion matrix* yang digunakan sebagai alat evaluasi kinerja model klasifikasi dalam membedakan antara kategori stroke dan non-stroke. Berdasarkan visualisasi tersebut, model mampu mengklasifikasikan secara tepat masing-masing sebanyak 32 data pada kelas *No Stroke* (True Negative) dan *Stroke* (True Positive), namun masih terdapat kesalahan klasifikasi berupa 10 data *False Positive* dan 10 data *False Negative*. Nilai

akurasi, presisi, dan recall yang diperoleh masing-masing sebesar 76,19%, menunjukkan bahwa model memiliki performa klasifikasi yang relatif seimbang. Meskipun demikian, masih terdapat peluang untuk peningkatan akurasi prediktif, yang dapat dilakukan melalui optimalisasi parameter, pengayaan fitur, atau penerapan teknik penanganan ketidakseimbangan data guna meningkatkan kemampuan generalisasi model secara keseluruhan.

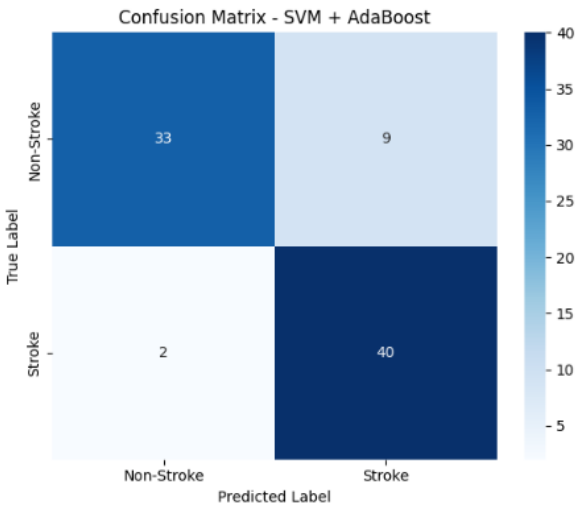
4.4. Hasil Optimasi Svm Dengan Adaboost

Classification Report:

	precision	recall	f1-score	support
0	0.94	0.79	0.86	42
1	0.82	0.95	0.88	42
accuracy			0.87	84
macro avg	0.88	0.87	0.87	84
weighted avg	0.88	0.87	0.87	84

Gambar 7. Gambar Hasil Prediksi Svm+Adaboost

Gambar tersebut menampilkan *classification report* dari hasil prediksi model Support Vector Machine (SVM) yang telah dioptimasi menggunakan algoritma Adaptive Boosting (AdaBoost). Pada kelas 0 (*No Stroke*), model menghasilkan nilai precision sebesar 0,94, recall sebesar 0,79, dan f1-score sebesar 0,86. Sementara itu, pada kelas 1 (*Stroke*), diperoleh precision sebesar 0,82, recall sebesar 0,95, dan f1-score sebesar 0,88. Secara keseluruhan, model menunjukkan performa yang baik dengan akurasi sebesar 87%, serta nilai *macro average* dan *weighted average* masing-masing sebesar 0,88 untuk precision, dan 0,87 untuk recall serta f1-score. Nilai recall yang tinggi pada kelas 1 menunjukkan bahwa integrasi AdaBoost mampu meningkatkan sensitivitas model dalam mendeteksi kasus stroke, yang merupakan kelas minoritas. Dengan demikian, pendekatan ensemble ini terbukti efektif dalam meningkatkan performa klasifikasi, khususnya dalam mengurangi kesalahan prediksi terhadap data yang sebelumnya sulit diidentifikasi dengan benar oleh model dasar.



Gambar 8. Gambar Confusion Matrix Svm+Adaboost

Gambar tersebut menunjukkan *confusion matrix* dari hasil klasifikasi model Support Vector Machine (SVM) yang telah dioptimasi menggunakan algoritma Adaptive Boosting (AdaBoost). Berdasarkan matriks tersebut, model mampu mengklasifikasikan dengan benar sebanyak 33 data pada kelas *Non-Stroke* (True Negative) dan 40 data pada kelas *Stroke* (True Positive). Adapun kesalahan klasifikasi yang terjadi terdiri atas 9 data *Non-Stroke* yang diprediksi sebagai *Stroke* (False Positive) dan 2 data *Stroke* yang diprediksi sebagai *Non-Stroke* (False Negative). Hasil ini mencerminkan kinerja model yang sangat baik, khususnya dalam mendeteksi kasus stroke, yang ditunjukkan oleh nilai *False Negative* yang sangat rendah. Kemampuan ini sangat krusial dalam konteks prediksi medis, mengingat kesalahan dalam mendeteksi pasien dengan risiko stroke dapat berimplikasi serius. Oleh karena itu, integrasi AdaBoost pada SVM secara empiris terbukti efektif dalam meningkatkan akurasi dan sensitivitas model terhadap kelas positif (*Stroke*) tanpa mengorbankan keseimbangan klasifikasi antar kelas.

Tabel 3. Tabel Perbandingan Hasil Akurasi

Algoritma	Hasil Akurasi
Support Vector Machinet Tanpa Adaboost	76%
Support Vector Machinet Dengan Adaboost	87%

Berdasarkan tabel 3 dapat dilihat bahwa, akurasi svm tanpa adaboost sebesar 76%, sedangkan akurasi svm dengan adaboost meningkat menjadi 87%. peningkatan ini menunjukkan bahwa penggunaan teknik ensemble adaboost mampu meningkatkan performa model dalam melakukan klasifikasi, khususnya dalam konteks prediksi penyakit stroke.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian, penerapan metode Adaptive Boosting (AdaBoost) pada algoritma Support Vector Machines (SVM) terbukti mampu meningkatkan performa model dalam melakukan prediksi penyakit stroke otak. Model SVM tanpa optimasi menunjukkan akurasi sebesar 76% dengan nilai precision, recall, dan f1-score yang seimbang di angka 0,76. Setelah dioptimasi menggunakan AdaBoost dan teknik undersampling ClusterCentroids, kinerja model meningkat secara signifikan dengan akurasi sebesar 87%, precision sebesar 0,82, recall 0,95, dan f1-score 0,88. Hasil ini menunjukkan bahwa integrasi AdaBoost dapat mengatasi kelemahan SVM dalam menghadapi ketidakseimbangan data, sekaligus meningkatkan kemampuan generalisasi model secara keseluruhan. Penelitian selanjutnya disarankan untuk mengeksplorasi metode balancing lain seperti SMOTE atau ADASYN serta membandingkan kinerja algoritma ensemble lainnya seperti XGBoost atau LightGBM. Selain itu, penggunaan validasi silang juga direkomendasikan guna meningkatkan reliabilitas

dan generalisasi model terhadap data yang lebih kompleks dan beragam.

DAFTAR PUSTAKA

[1] S. Wulandari, Y. I. Mukti, and T. Susanti, "OPTIMALISASI PREDIKSI PENYAKIT STROKE MENGGUNAKAN," vol. 8, no. 2, pp. 1826–1833, 2024.

[2] W. H. A. Susanto *et al.*, "Dietika penyakit degeneratif," no. July, p. 228, 2023.

[3] N. Nurbidayah, F. Y. Afra, M. Saufi, and B. Bandawati, "Upaya Pencegahan Penyakit Degeneratif Melalui Edukasi Kesehatan Pada Lansia di Desa Penggalaman Kecamatan Martapura Barat," *I-Com Indones. Community J.*, vol. 4, no. 1, pp. 101–107, 2024.

[4] F. Fatihaturahmi, Y. Yuliana, and A. Yulastri, "Literature Review : Penyakit Degeneratif : Penyebab, Akibat, Pencegahan Dan Penanggulangan," *JGK J. Gizi dan Kesehat.*, vol. 3, no. 1, pp. 63–72, 2023.

[5] M. N. Maskuri, K. Sukerti, and R. M. Herdian Bhakti, "Penerapan Algoritma K-Nearest Neighbor (KNN) untuk Memprediksi Penyakit Stroke Stroke Desease Predict Using KNN Algorithm," *J. Ilm. Intech Inf. Technol. J. UMUS*, vol. 4, no. 1, pp. 130–140, 2022.

[6] S. H. R. R. Ramadhan, Alpito Gilang; Hidayatullah, "Implementasi Algoritma C4 . 5 pada Sistem Prediksi Stroke Berdasarkan Data Kesehatan," vol. 4, no. 2, pp. 1617–1624, 2025.

[7] Y. Azhar, A. K. Firdausy, and P. J. Amelia, "Perbandingan Algoritma Klasifikasi Data Mining Untuk Prediksi Penyakit Stroke," *SINTECH (Science Inf. Technol. J.*, vol. 5, no. 2, pp. 191–197, 2022.

[8] H. Hendriyansyah, A. Irma Purnamasari, and T. Suprapti, "Penerapan Algoritma Decision Tree Dalam Klasifikasi Penyakit Stroke Otak," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 3, pp. 3038–3043, 2024.

[9] A. K. Hermawan and A. Nugroho, "Analisa Data Mining Untuk Prediksi Penyakit Ginjal Kronik Dengan Algoritma Regresi Linier," vol. 4, no. 1, 2023.

[10] M. R. S. Alfarizi, M. Z. Al-farish, M. Taufiqurrahman, G. Ardiansah, and M. Elgar, "Penggunaan Python Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning," *Karya Ilm. Mhs. Bertauhid (KARIMAH TAUHID)*, vol. 2, no. 1, pp. 1–6, 2023.

[11] R. R. S. Putri Kumala Sari, "Vol 7 No 1 , Februari 2024 KOMPARASI ALGORITMA SUPPORT VECTOR MACHINE DAN RANDOM," vol. 7, no. 1, pp. 31–39, 2024.

[12] Y. Septiawan and Chairani, "Perbandingan Akurasi Metode Deteksi Ujaran Kebencian dalam Postingan Twitter Menggunakan Metode SVM dan Decision Trees yang Dioptimalkan dengan Adaboost," *J. Tek.*, vol. 17, no. 2, pp. 297–299, 2023.

[13] H. Syahrizal and M. S. Jailani, "Jenis-Jenis Penelitian Dalam Penelitian Kuantitatif dan Kualitatif," *J. QOSIM J. Pendidik. Sos. Hum.*, vol. 1, no. 1, pp. 13–23, 2023.

[14] H. S. W. Hovi, A. Id Hadiana, and F. Rakhmat Umbara, "Prediksi Penyakit Diabetes Menggunakan Algoritma Support Vector Machine (SVM)," *Informatics Digit. Expert*, vol. 4, no. 1, pp. 40–45, 2022