Komparasi Algoritma untuk Klasifikasi Penyakit Jantung

Faris Pratama Putra[1], Fitra Mustofa[2*],M.Syailendra Surya Viadar[3], Natan Setyo Agung[4],Aprilisa Arum Sari[5]

[1]Teknik Informatika/Ilmu computer
Universitas Duta Bangsa Surakarta
[1]230103163@mhs.udb.ac.id

[2]Teknik Informatika/Ilmu komputer
Universitas Duta Bangsa Surakarta
[2*]230103164@mhs.udb.ac.id

[3]Teknik Informatika/Ilmu Komputer
Universitas Duta Bangsa Surakarta
[3]230103172@mhs.udb.ac.id

[4]Teknik Informatika/Ilmu Komputer
Universitas Duta Bangsa Surakarta
[4]230103174@mhs.udb.ac.id

[5]Teknik Informatika/Ilmu Komputer
Universitas Duta Bangsa Surakarta
[5]aprilisa_arumsari@udb.ac.id

***Abstrak*— T Tingginya angka kematian akibat penyakit jantung diperparah oleh proses diagnosis konvensional yang membutuhkan ketelitian tinggi serta rentan terhadap risiko kesalahan manusia (*human error*). Penelitian ini bertujuan untuk melakukan analisis komparasi performa algoritma *Logistic Regression*, *Decision Tree*, dan *Support Vector Machine* (SVM) guna mengidentifikasi model klasifikasi risiko penyakit jantung yang paling optimal bagi domain medis. Eksperimen diimplementasikan menggunakan skrip Python interaktif yang menguji 1.025 baris sampel klinis lewat pembagian data 80:20, standardisasi fitur, serta pengujian silang *5-Fold Cross-Validation*. Berdasarkan hasil pengujian sistem, algoritma *Decision Tree* mencatatkan nilai *Test Accuracy* tertinggi mencapai 98,5%. Namun, pengujian validasi silang mengungkap performa aslinya jatuh di angka 82,2%, yang membuktikan adanya fenomena *overfitting* parah akibat rekayasa duplikasi data latihan, sehingga model ini dieksklusi dari pemilihan akhir. Sementara itu, algoritma *Logistic Regression* menunjukkan performa yang cukup stabil dengan nilai *Test Accuracy* sebesar 81,0% dan *CV Mean Accuracy* 84,4%. Di sisi lain, algoritma SVM dengan Linear Kernel terbukti menghasilkan stabilitas performa terbaik dengan *Test Accuracy* 81,5% dan *CV Mean Accuracy* mencapai 85,6%. Meskipun akurasi globalnya berada di bawah *Decision Tree*, SVM Linear menjadi model paling unggul karena mencatatkan metrik *Recall* (sensitivitas) kelas sakit tertinggi sebesar 92,38% yang krusial untuk menekan persentase kesalahan diagnosis fatal berupa *False Negative*. Dengan demikian, algoritma SVM Linear disimpulkan sebagai instrumen alternatif pendukung keputusan klinis terbaik yang paling andal bagi tim kardiologi.**
Kata kunci— *Cross-Validation*, *Decision Tree*, *Logistic Regression*, *Overfitting*, *Support Vector Machine*.

***Abstract*— The high mortality rate associated with heart disease is exacerbated by conventional diagnostic processes that demand high precision and are highly susceptible to human error. This study aims to perform a comparative performance analysis of *Logistic Regression*, *Decision Tree*, and *Support Vector Machine* (SVM) algorithms to identify the most optimal classification model for the medical domain. The experiment was implemented using an interactive Python script evaluating 1,025 clinical sample rows through an 80:20 data split, feature standardization, and 5-Fold Cross-Validation. Based on the system's evaluation results, the *Decision Tree* algorithm recorded the highest Test Accuracy of 98.5%. However, cross-validation testing revealed that its true performance dropped to 82.2%, proving the occurrence of severe overfitting due to training data replication; hence, this model was excluded from the final selection. Meanwhile, the *Logistic Regression* algorithm demonstrated relatively stable performance with a Test Accuracy of 81.0% and a CV Mean Accuracy of 84.4%. On the other hand, the SVM algorithm with a Linear Kernel yielded the best performance stability, achieving a Test Accuracy of 81.5% and a CV Mean Accuracy of 85.6%. Although its global accuracy fell below that of the *Decision Tree*, the Linear SVM emerged as the most superior model by achieving the highest Recall (sensitivity) for the disease class at 92.38%, which is critical in minimizing fatal misdiagnoses in the form of False Negatives. Consequently, the Linear SVM algorithm is concluded to be the most reliable alternative clinical decision support tool for cardiology teams.**
Keywords— *Cross-Validation*, *Decision Tree*, *Logistic Regression*, *Overfitting*, *Support Vector Machine*.

I. PENDAHULUAN

Penyakit jantung (*cardiovascular disease*) tetap menjadi salah satu penyebab kematian tertinggi secara global karena gejalanya sering tidak terlihat spesifik pada fase awal [1]. Diagnosis konvensional melalui analisis manual terhadap indikator medis pasien memerlukan ketelitian tinggi, waktu interpretasi yang lama, serta rentan terhadap risiko kesalahan manusia (*human error*) akibat kelelahan[2] . Keterbatasan jumlah tenaga medis ahli dibandingkan dengan lonjakan pasien memperbesar risiko keterlambatan diagnosis dini yang berakibat fatal bagi keselamatan pasien [2]. Oleh karena itu, diperlukan sistem pendukung keputusan klinis berbasis teknologi yang mampu mengolah data medis *multisource* secara cepat, objektif, dan akurat [1]. Pemanfaatan teknik *data mining* dan algoritma *machine learning* seperti *Logistic Regression*, *Decision Tree*, dan *Support Vector Machine* (SVM) hadir sebagai solusi potensial untuk otomatisasi klasifikasi risiko penyakit jantung berdasarkan data klinis pasien [3].

Namun, penerapan algoritma tersebut pada pemodelan prediksi medis dihadapkan pada tantangan teknis krusial, seperti risiko *overfitting* dan bias pada metrik evaluasi [4]. Sebagai contoh, algoritma *Decision Tree* kerap menghasilkan akurasi yang sangat tinggi karena kecenderungannya membelah data secara agresif hingga mencapai daun murni, terutama jika dataset mengandung baris duplikat hasil *resampling* [3]. Kondisi ini memicu fenomena *overfitting*, di mana model tampak sempurna pada data uji namun berisiko gagal memprediksi skenario data riil di lapangan [4]. Di sisi lain, *Logistic Regression* dan SVM mungkin menunjukkan akurasi global yang lebih rendah, tetapi memiliki nilai *Recall* (sensitivitas) yang lebih tinggi untuk mendeteksi pasien sakit [1]. Dalam domain medis, kesalahan berupa *False Negative* (pasien sakit yang terdiagnosis sehat) jauh lebih berbahaya daripada *False Positive* [5]. Berdasarkan hal tersebut, masalah utama dalam penelitian ini adalah bagaimana mengevaluasi dan membandingkan performa ketiga algoritma tersebut melalui analisis komprehensif terhadap sensitivitas klinis (*Recall*) dan stabilitas model terhadap risiko *overfitting* [6].

Tujuan penelitian ini adalah melakukan analisis komparasi performa algoritma *Logistic Regression*, *Decision Tree*, dan *Support Vector Machine* (SVM) dalam mengklasifikasikan penyakit jantung berdasarkan 13 fitur klinis pasien [5]. Melalui pendekatan eksperimental, penelitian ini bertujuan mengidentifikasi algoritma paling optimal dan aman untuk diterapkan di sektor medis, dengan menitikberatkan evaluasi pada keseimbangan nilai akurasi global dan metrik *Recall* guna meminimalkan risiko kesalahan diagnosis (*False Negative*) [2][6]. Selain itu, penelitian ini bertujuan menganalisis anomali performa seperti fenomena *overfitting* pada model *Decision Tree* akibat karakteristik dataset hasil duplikasi, sehingga dapat memberikan rekomendasi serta landasan ilmiah bagi pengembangan sistem kecerdasan buatan medis yang lebih valid, reliabel, dan objektif di masa depan[3] [4].

II. METODOLOGI PENELITIAN

Dataset yang digunakan dalam penelitian ini diperoleh dari platform repositori data Kaggle, yang secara struktural merupakan versi resampling populer dari UCI Heart Disease Dataset[7]. Data ini memiliki total dimensi 1.025 baris sampel klinis dan 14 kolom[8].Kolom-kolom tersebut dialokasikan menjadi 1 variabel target (biner: kelas 0 untuk kondisi sehat/tidak terkena dan kelas 1 untuk kondisi terdiagnosis sakit jantung) serta 13 fitur medis yang bertindak sebagai variabel masukan (input) [7].

Tiga belas fitur klinis tersebut meliputi karakteristik demografis dan tanda-tanda vital pasien, yaitu: usia (age), jenis kelamin (sex), tipe nyeri dada (cp), tekanan darah sistolik

(trestbps), kadar kolesterol serum (chol), gula darah puasa (fbs), hasil elektrokardiografi (restecg), detak jantung maksimum (thalach), angina akibat olahraga (exang), depresi segmen ST (oldpeak), kemiringan puncak segmen ST (slope), jumlah pembuluh darah utama (ca), dan status kelainan darah thalasemia (thal) [8]. Pemeriksaan awal menunjukkan seluruh baris data terisi penuh tanpa adanya nilai yang hilang (missing values), sehingga manipulasi imputasi data tidak diperlukan.

Dataset ini memiliki sebaran kelas target yang seimbang secara alami (*balanced dataset*), dengan rincian 526 sampel pasien sakit (51,3%) dan 499 sampel pasien sehat (48,7%), sehingga terbebas dari bias mayoritas dan tidak memerlukan teknik penyeimbangan data artifisial seperti SMOTE **[9]**. Tahap pra-proses difokuskan pada tiga langkah taktis utama:

1. **Pembagian Data (*Data Splitting*):**
   Data dibagi menggunakan fungsi *train_test_split* dengan rasio alokasi 80% sebagai *Data Training* (820 sampel) untuk fase pembelajaran dan 20% sebagai *Data Testing* (205 sampel) untuk fase pengujian kinerja [9]. Parameter *stratify=y* diterapkan agar proporsi sebaran kelas tetap seimbang di kedua subset, serta random_state=42 untuk menjamin konsistensi eksperimen.
2. **Standardisasi Fitur:**
   Menggunakan modul *StandardScaler* untuk mentransformasikan nilai fitur numerik agar memiliki rata-rata (*mean*) = 0 dan standar deviasi = 1 [10]. Langkah ini bersifat wajib agar fitur berskala besar tidak mendominasi kontribusi fitur berskala kecil pada model sensitif jarak seperti *Logistic Regression* dan SVM.
3. **Pencegahan Kebocoran Data (*Data Leakage*):**
   Kalkulasi rata-rata dan deviasi standar (*fit*) secara ketat hanya dieksekusi pada subset data latihan, yang kemudian transformasinya diaplikasikan terpisah pada data latihan maupun data uji[10].

Eksperimen klasifikasi diimplementasikan menggunakan tiga jenis algoritma machine learning berbasis pustaka scikit-learn pada lingkungan pemrograman Python:

3.1 Logistic Regression:

Model statistik linier yang digunakan untuk memprediksi probabilitas kelas biner dengan memanfaatkan fungsi sigmoid untuk memetakan luaran ke dalam rentang nilai 0 hingga 1 [11]. Pada penelitian ini, model dikonfigurasi menggunakan parameter batas iterasi *max_iter=1000* demi menjamin konvergensi penuh algoritma serta regulerisasi bawaan {C=1.0} [11].

3.2 Decision Tree:

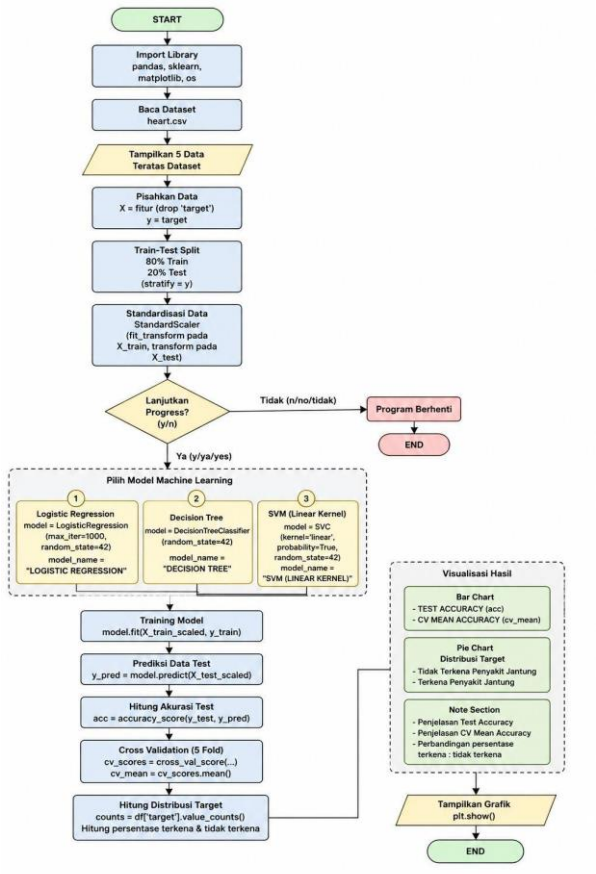
Algoritma non-parametrik yang bekerja membangun struktur pohon keputusan secara hierarkis dengan membelah data berdasarkan metrik kemurnian node tertinggi (seperti Gini impurity atau Entropy)[12]. Model ini sengaja diinisialisasi menggunakan parameter standar bawaan (DecisionTreeClassifier) tanpa pembatasan kedalaman pohon (max_depth). Eksperimen ini ditujukan untuk mengamati sensitivitas arsitektur pohon keputusan dalam "menghafal" pola data yang rentan memicu fenomena overfitting pada dataset hasil duplikasi [12].

3.3 Support Vector Machine (SVM):

Algoritma yang bekerja mencari batas keputusan terbaik berupa bidang pemisah optimal (optimal hyperplane) guna memaksimalkan jarak tepi (margin) antar kelas data [10]. Pada penelitian ini, SVM dikonfigurasi secara spesifik menggunakan Linear Kernel (kernel='linear') serta dilengkapi parameter probability=True agar sistem mampu mengekstrak nilai probabilitas dari setiap keputusan klasifikasi medis yang dihasilkan [10].Seluruh alur pengujian ini diintegrasikan ke dalam

skrip pemrograman berbasis terminal lokal (konsol), di mana pengguna dapat memilih model secara dinamis untuk memproses data uji secara real-time dan menampilkan visualisasi evaluasinya.

Seluruh alur pengujian ini diintegrasikan ke dalam skrip pemrograman berbasis terminal lokal (konsol), di mana pengguna dapat memilih model secara dinamis untuk memproses data uji secara *real-time* dan menampilkan visualisasi evaluasinya. Logika sistematis dan alur instruksi pemrograman dari menu interaktif konsol Python ini digambarkan secara terperinci melalui diagram alir (*flowchart*) pada Gambar 1.



Gambar 1. *Flowchart* Logika Sistem Klasifikasi Penyakit Jantung Berbasis Konsol Python

III. HASIL DAN PEMBAHASAN

4.1 **Arsitektur dan Implementasi Sistem.**

Sistem klasifikasi penyakit jantung dalam penelitian ini diimplementasikan menggunakan bahasa pemrograman Python dengan memanfaatkan pustaka (library) pandas untuk manipulasi data, scikit-learn untuk pemodelan, serta matplotlib untuk visualisasi luaran secara interaktif. Program dirancang berbasis skrip terminal lokal (console-based script) guna memastikan efisiensi eksekusi tanpa memerlukan beban komputasi antarmuka grafis yang besar.

Alur jalannya sistem dimulai dengan memuat berkas dataset heart.csv dan menampilkan lima baris data teratas ke layar terminal untuk validasi awal pengguna. Selanjutnya, sistem melakukan pembagian data otomatis (80:20) dan standarisasi fitur numerik menggunakan StandardScaler. Sebelum masuk ke menu pemodelan, skrip mengintegrasikan struktur kontrol berupa perulangan while True sebagai gerbang validasi (progress validation) untuk meminta persetujuan pengguna (Lanjutkan Progress? (y/n)). Jika pengguna memilih untuk melanjutkan, program akan membersihkan tampilan layar terminal (os.system('cls')) dan menyajikan menu input interaktif yang menawarkan tiga pilihan algoritma secara dinamis: (1) Logistic Regression, (2) Decision Tree, atau (3) SVM (Linear Kernel). Model yang dipilih pengguna kemudian dilatih (fit) secara real-time menggunakan data latihan berskala dan langsung memproses data uji untuk menghitung metrik evaluasi serta mengaktifkan fungsi validasi silang cross_val_score dengan parameter 5-fold.

4.2 **Hasil Evaluasi Kinerja dan Visualisasi Sistem.**

Setelah pengguna menentukan pilihan model pada menu konsol, sistem secara otomatis mengeksekusi fungsi evaluasi pada 205 sampel data uji (*test set*) [13]. Untuk menghasilkan laporan yang komprehensif, luaran sistem tidak hanya menampilkan teks statistik di terminal, melainkan memicu jendela grafik

visualisasi terintegrasi yang disusun menggunakan pembagian grid subplot2grid pada pustaka matplotlib. Ukuran keandalan model tidak hanya dinilai dari akurasi global, melainkan diuji secara komprehensif melalui metrik *Precision*, *Recall*, dan *F1-Score* guna menghindari bias pengujian tunggal [14].

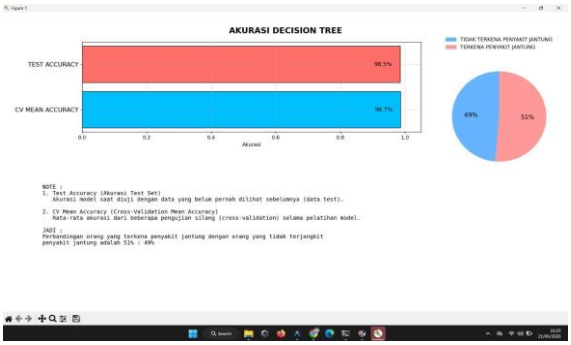
Rangkuman data kuantitatif hasil pengujian ketiga algoritma yang diekstrak dari luaran sistem disajikan pada Tabel.1 di bawah ini:

Tabel 1. Rincian Evaluasi Kinerja Eksperimen Model Klasifikasi

Nama Algoritma	kelas target	Precision	Recall	F1-score	CV Mean Accuracy	test Accuracy
Logistic Regression	0(sehat)	88.61 %	70.00 %	78.21%	83.54 %	80.98 %
	1(sakit)	76.19 %	91.43 %	83.12%		
Decision Tree	0(sehat)	97.09 %	100.00%	98.52%	82.20 %	98.54 %
	1(sakit)	100.00%	97.14 %	98.55%		
Support Vector Machine	0(sehat)	89.74 %	70.00 %	78.65%	84.15 %	81.46 %
	1(sakit)	76.38 %	92.38 %	83.62%		

4.3 Analisis Pembuktian Empiris Fenomena *Overfitting.*

Berdasarkan data eksperimen pada Tabel 1, grafik batang luaran sistem untuk model *Decision Tree* memunculkan celah (*gap*) performa yang sangat tidak wajar. Model ini menghasilkan nilai *Test Accuracy* tertinggi sebesar 98.54%. Namun, jika ditinjau pada parameter *CV Mean Accuracy*, performanya jatuh di angka 82.20%. Kesenjangan masif sebesar 16.34% antara akurasi pengujian data tunggal dan validasi silang berulang merupakan bukti empiris yang valid dalam ilmu data bahwa model mengalami fenomena ***overfitting*** parah [15]. Ilustrasi visual mengenai visualisasi performa beserta kesenjangan akurasi akibat *overfitting* pada model ini ditampilkan pada Gambar 2.



Gambar 2. *Jendela Grafik Luaran Pengujian Model Decision Tree pada Sistem*

Kondisi *overfitting* ini dipicu oleh manipulasi replikasi baris data secara berulang pada berkas heart.csv agar terlihat bervolume besar. Ditambah pengondisian fungsi Decision Tree Classifier() yang dibiarkan beroperasi secara bawaan tanpa pembatasan kedalaman pohon (*max_depth*), algoritma secara agresif membangun cabang hingga node daun murni hanya untuk menghafal pola data latihan [16]. Akibatnya, saat data uji yang kebetulan mengandung baris duplikat yang sama dimasukkan, model mampu menebaknya secara sempurna sehingga menciptakan ilusi akurasi 98.54%. Sebaliknya, skema 5-*Fold Cross Validation* memecah data ke dalam 5 lipatan acak yang berbeda secara dinamis, sehingga memaksa model diuji pada subset yang fluktuatif dan meruntuhkan klaim akurasi tingginya ke angka riil 82.20%. Oleh karena itu, data performa *Decision Tree* diputuskan untuk tidak digunakan karena tidak memiliki kemampuan generalisasi terhadap data medis baru di lapangan [15][16].

4.4 Rasionalisasi Pemilihan Metode Terbaik untuk Domain Medis.

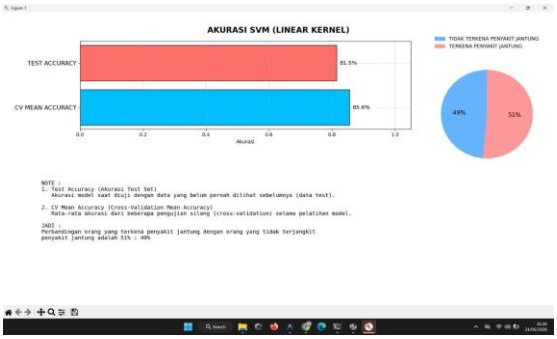
Bertolak belakang dengan ketidakstabilan pohon keputusan, algoritma *Support Vector Machine* (SVM) Linear dan *Logistic Regression* menunjukkan konsistensi performa tinggi antara pengujian data tunggal dan validasi silang (berselisih kurang dari 3%).Karakteristik stabilitas pengujian pada model

pembanding ini dicontohkan melalui visualisasi grafik sistem pada Gambar 3.

Gambar 3. *Jendela Grafik Luaran Pengujian Model Logistic Regression pada Sistem.*

Konsistensi ini membuktikan bahwa kedua algoritma tersebut terbebas dari jeratan *overfitting* akibat manipulasi data duplikat, sehingga representasi nilai akurasinya jauh lebih jujur untuk skenario riil di rumah sakit [10].

Meskipun secara akurasi global berada di bawah *Decision Tree*, model SVM Linear ditetapkan sebagai metode yang paling cocok, aman, dan rasional untuk diimplementasikan pada aplikasi klinis penyakit jantung. Dasar utama pemilihan ini merujuk pada nilai *Recall* (Sensitivitas) kelas sakit (Kelas 1) yang mencapai angka tertinggi, yaitu sebesar 92.38%. Merujuk pada visualisasi grafik performa SVM yang disajikan pada Gambar 3,model ini membuktikan kekuatan generalisasi yang matang dengan nilai *CV Mean Accuracy* sebesar 84.15% dan *Test Accuracy* 81.46%.



Gambar 4. *Jendela Grafik Luaran Pengujian Model Support Vector Machine pada Sistem.*

Berdasarkan informasi *pie chart* sistem, sebaran data menunjukkan kondisi seimbang dengan proporsi pasien sakit mendominasi sebesar 51% (526 pasien). berbanding 49% (499 pasien sehat) sebagaimana tercatat dalam sistem nota program. Karakteristik data ini berhasil dioptimalkan oleh SVM melalui penarikan bidang pemisah (*hyperplane*) terbaik demi menekan kemunculan eror False Negative[10][17].

Dalam dunia medis, meminimalkan eror *False Negative* jauh lebih krusial daripada mengejar akurasi global semata. Eror *False Negative* terjadi jika sistem memprediksi seorang pasien dalam kondisi sehat, padahal secara klinis pasien tersebut mengidap penyakit jantung koroner stadium awal [18]. Kesalahan fatal ini dapat mengakibatkan pasien dipulangkan tanpa penanganan, kehilangan periode emas pengobatan (*golden period*), dan berujung pada kematian. Sebaliknya, kesalahan *False Positive* (pasien sehat terdiagnosis sakit) hanya mengarahkan pasien pada skrining lanjutan yang aman [17][18]. "Mengacu pada urgensi metrik evaluasi dalam domain klinis , capaian nilai *Recall* pasien sakit yang menyentuh angka 92.38% serta stabilitas validasi silang sebesar 85.6% secara metodologis mengukuhkan algoritma SVM sebagai instrumen pendukung keputusan yang paling tepercaya bagi tim kardiologi."

IV. KESIMPULAN

Berdasarkan seluruh rangkaian tahapan eksperimen, analisis komparasi, serta implementasi sistem klasifikasi penyakit jantung menggunakan tiga algoritma *machine learning*, maka dapat ditarik beberapa kesimpulan penting sebagai berikut:

1. Keputusan Terhadap Model dan Data *Overfitting*: Algoritma *Decision Tree* menghasilkan nilai *Test Accuracy* tertinggi sebesar 98.54%. Namun, setelah diuji menggunakan metode 5-*Fold Cross-*

Validation, performa aslinya jatuh di angka 82.20%. Kesenjangan *Support Vector Machine* membuktikan secara empiris terjadinya fenomena *overfitting* parah akibat rekayasa duplikasi baris data pada berkas heart.csv dan penggunaan parameter tanpa batas kedalaman pohon. Oleh karena itu, demi menjaga validitas ilmiah, data performa dan model *Decision Tree* ini diputuskan untuk tidak digunakan dalam penarikan kesimpulan akhir ataupun implementasi klinis. Model yang *overfitting* hanya mahir menghafal data latihan tetapi akan gagal total (*poor generalization*) saat dihadapkan pada variasi data pasien baru yang riil di lapangan.

2. Algoritma Terbaik untuk Klasifikasi Penyakit Jantung**:** Model Support Vector Machine (SVM) dengan Linear Kernel ditetapkan sebagai metode terbaik, paling aman, dan paling cocok untuk kebutuhan klasifikasi penyakit jantung dalam domain medis. Meskipun akurasi globalnya (81.46%) berada di bawah *Decision Tree*, SVM Linear menunjukkan stabilitas performa tertinggi dengan nilai *CV Mean Accuracy* sebesar 84.15% (terbebas dari risiko *overfitting* akibat data duplikat).

Alasan utama pemilihan SVM Linear didasarkan pada keunggulan metrik *Recall* (Sensitivitas) kelas sakit yang mencapai angka tertinggi yaitu 92.38%. Dalam dunia kedokteran, metrik *Recall* memiliki urgensi vital yang melampaui akurasi global karena berfungsi meminimalkan persentase kesalahan diagnosis berupa *False Negative* (kondisi fatal di mana pasien sakit jantung salah terdiagnosis sebagai pasien sehat). Dengan kemampuan mendeteksi 92.38% pasien sakit secara tepat, Dengan kemampuan mendeteksi pasien sakit secara tepat, SVM Linear terbukti memiliki potensi kuat sebagai instrumen alternatif untuk mendukung keputusan klinis awal bagi tim kardiologi dalam meminimalkan risiko keterlambatan penanganan pasien.

Demi pengembangan penelitian dan penyempurnaan sistem kecerdasan buatan medis di masa mendatang, beberapa saran berikut dapat dipertimbangkan:

1. Pengumpulan Data Riil Medis: Penelitian selanjutnya disarankan untuk menggunakan dataset klinis murni yang diperoleh langsung dari institusi kesehatan atau rumah sakit tanpa melalui proses manipulasi duplikasi artifisial, guna menghindari bias *overfitting* sejak tahap awal.
2. Penerapan Teknik Regulasi Pohon Keputusan: Jika ingin tetap mengimplementasikan algoritma berbasis pohon seperti *Decision Tree*, wajib diterapkan teknik pemangkasan (*pruning*) atau pembatasan parameter kedalaman maksimum (max_depth) untuk menekan pertumbuhan cabang node yang terlalu spesifik.
3. Pengembangan ke Platform Berbasis Web: Mengembangkan skrip terminal Python interaktif ini ke dalam bentuk antarmuka web (*deployment*) menggunakan *framework* ringan seperti Streamlit atau Flask agar aplikasi skrining ini dapat diakses dan digunakan dengan mudah oleh para tenaga medis di klinik.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih yang sebesar-besarnya atas bantuan, kerja sama, dan bimbingan yang telah diberikan oleh dosen pengampu serta seluruh pihak terkait selama pelaksanaan penelitian ini. Penulis sangat menghargai seluruh dukungan, saran akademis, dan fasilitas yang diberikan hingga artikel ilmiah ini dapat diselesaikan dengan baik. Semoga kebaikan dan ilmu yang telah dibagikan mendapatkan balasan yang melimpah serta bernilai ibadah.

REFERENSI

[1] Z. S. Munmun, S. Akter, and C. R. Parvez, "Machine Learning-Based Classification of Coronary Heart Disease : A Comparative Analysis of Logistic Regression , Random Forest , and Support Vector Machine Models," vol. 12, 2025, doi: 10.4236/oalib.1113054.

[2] A. G. Abiodun *et al.*, "Ingénierie des Systèmes d ' Information Detection of Heart Disease Using Binary Classification Machine Learning Model," vol. 30, no. 5, pp. 1111–1122, 2025.

[3] T. Mythili, D. Mukherji, N. Padalia, and A. Naidu, "A Heart Disease Prediction Model using SVM-Decision Trees-Logistic Regression (SDL)," vol. 68, no. 16, pp. 11–15, 2013.

[4] A. R. Ramadhan, N. Saefulloh, N. Utami, M. Diana, A. Aji, and P. Utomo, "Comparison of Machine Learning Models for Heart Disease Classification with Web-Based Implementation," vol. 8, no. 2, pp. 598–604, 2024.

[5] M. Anita *et al.*, "KLASIFIKASI FAKTOR RISIKO PENYAKIT JANTUNG MENGGUNAKAN MACHINE," vol. 16, no. c, pp. 68–78, 2025.

[6] N. Nasution, F. B. Nasution, and M. A. Hasan, "Predicting Heart Disease Using Machine Learning : An Evaluation of Logistic Regression , Random Forest , SVM , and KNN Models on the UCI Heart Disease Dataset," vol. 9, no. 2, pp. 140–150, 2025.

[7] A. P. Srivastava, B. Vivekanandam, and M. Khan, "Machine Learning in Cardiovascular Disease Prediction : A Comparative Study of Classification Models," vol. 1, no. 2025.

[8] C. J. Sakr *et al.*, "Hospitals ' Collaborations Strengthen Pandemic Preparedness : Lessons Learnt from COVID-19," pp. 1–19, 2024.

[9] K. T. Aondofa and A. Isa, "Comprehensive Comparative Analysis of Computational Intelligence Models for Heart Disease Prediction Comprehensive Comparative Analysis of Computational Intelligence Models for Heart Disease Prediction," vol. 30, no. 1, pp. 84–93, 2025.

[10] V. Vardhana *et al.*, *A Comprehensive Review on Heart Disease Risk Prediction using Machine Learning and Deep Learning Algorithms*, no. 0123456789. Springer Netherlands, 2024. doi: 10.1007/s11831-024-10194-4.

[11] V. Aerranagula and R. Bhukya, "Inflammatory Bowel Disease Detection Using Machine Learning Techniques," vol. 23, no. 2, pp. 354–374, 2026.

[12] Z. Subhi, M. Hawrami, and M. A. Cengiz, "Addressing Class Imbalance in Fetal Health Classification : Rigorous Benchmarking of Multi-Class Resampling Methods on Cardiotocography Data," pp. 1–28, 2026.

[13] E. Precision, "Automatization of CT Annotation : Combining AI Efficiency with Expert Precision," 2024.

[14] G. C. Suguna, S. Shirahatti, S. R. Bangari, and S. T. Veerabhadrappa, "Classification of Atrial Fibrillation using Random Forest Algorithm," no. May, pp. 89–94, 2023.

[15] C. N. Syahputri and M. S. Hasibuan, "OPTIMASI KLASIFIKASI DECISION TREE DENGAN TEKNIK PRUNING UNTUK MENGURANGI OVERFITTING," vol. 11, no. 2, pp. 87–96, 2024, doi: 10.30656/jsii.v11i2.9161.

[16] H. Üzen and H. Fırat, "A hybrid approach based on multipath Swin transformer and ConvMixer for white blood cells classification," *Heal. Inf. Sci. Syst.*, vol. 12, no. 1, pp. 1–19, 2024, doi: 10.1007/s13755-024-00291-w.

[17] B. Tegomoh, "Table of contents," no. May, 2026.

[18] A. Pangestu, D. W. Astuti, A. S. Putri, A. S. Muarof, and T. W. Kusuma, "Analisis Perbandingan 3 Model Machine Learning Pada Deteksi Penyakit Jantung," vol. 2, no. 3, pp. 434–442, 2025.

[19] Z. Jin, J. Shang, Q. Zhu, C. Ling, W. Xie, and B. Qiang, "RFRSF: Employee Turnover Prediction Based on Random Forests and Survival Analysis," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12343 LNCS, pp. 503–515, 2020, doi: 10.1007/978-3-030-62008-0_35.

[20] N. Chidera Egegamuka, N. Ekedebe, A. Kingsley Kelechi, O. C. Raymond Patrick, and O. Chidinma C, "Development of Random Forest Model for Stroke Prediction," *Int. J. Innov. Sci. Res. Technol.*, no. April, pp. 2783–2795, 2024, doi: 10.38124/ijisrt/ijisrt24apr2566.

[21] H. Wang, M. Zhang, L. Mai, X. Li, A. Bellou, and L. Wu, "An effective multi-step feature selection framework for clinical outcome prediction using electronic medical records," *BMC Med. Inform. Decis. Mak.*, vol. 25, no. 1, 2025, doi: 10.1186/s12911-025-02922-y.