Journal of Applied Informatics and Computing (JAIC)
Vol.9, No.2, April 2025, pp. 318~327
e-ISSN: 2548-6861

Comparison of Support Vector Machine and Decision Tree Algorithm Performance with Undersampling Approach in Predicting Heart Disease Based on Lifestyle

Gusti Ayu Putu Febriyanti [1]*, Anna Baita [2]**
* Informatika, Universitas Amikom Yogyakarta, Indonesia
gustiayu@students.amikom.ac.id [1], anna@amikom.ac.id [2]

Article Info

Article history:

Received 2024-11-21
Revised 2024-12-10
Accepted 2025-01-21

Keyword:

Cardiac
Decision Tree,
K-Fold,
Support Vector Machine,
Prevention.

ABSTRACT

Heart disease is one of the leading causes of death in the world with risk factors such as atherosclerosis, high blood pressure, and smoking. Early diagnosis is essential to reduce mortality and improve patients' quality of life. This study evaluates the performance of two machine learning algorithms, namely Support Vector Machine (SVM) and Decision Tree (DT), in predicting heart disease risk by applying undersampling techniques to handle data imbalance. The K-fold cross-validation method with K=10 and hyperparameter tuning were applied to obtain the optimal performance of both models. The results showed that SVM without undersampling achieved 92% accuracy, while with undersampling the accuracy decreased to 76%. DT without undersampling has 91% accuracy, while with undersampling the accuracy reaches 75%. The undersampling technique successfully improved the balance in recognizing minority classes, although it reduced the overall accuracy. This finding confirms that SVM is more reliable in predicting heart disease in datasets with unbalanced class distribution.

I. PENDAHULUAN

Penyakit jantung merupakan penyakit kronis penyebab kematian nomor 1 di seluruh dunia [1]. Data World Health Organization (WHO) memperkirakan 17,9 juta orang di dunia meninggal akibat penyakit jantung [2]. Pravalensi penyakit jantung menunjukkan peningkatan seiring dengan pertambahan usia, kurangnya aktivitas fisik, penggunaan tembakau serta penggunaan alkohol yang berbahaya. Penyakit jantung coroner memiliki masa perkembangan yang panjang, sehingga dapat dicegah dengan memodifikasi gaya hidup sejak dini [2], [3]. Faktor risiko penyakit jantung dapat dibedakan menjadi 2, yaitu faktor mayor dan faktor minor, faktor risiko mayor yaitu umur, jenis kelamin, ras, merokok, hipertensi, serta diabetes melitus sedangkan faktor risiko minor yaitu stress, diet, nutrisi dan alkohol [3]. Deteksi tahap awal pada penyakit jantung merupakan hal yang penting untuk mengurangi jumlah korban. Dari sekian banyak teknik untuk meningkatkan deteksi dan diagnosis penyakit ini adalah, data mining [4].

Beberapa penelitian menunjukkan bahwa penggunaan machine learning memiliki potensi besar dalam mengatasi topik klasifikasi dan meningkatkan optimasi dalam pengembangan system layanan kesehatan [5]. Pada penelitian ini, dilakukan evaluasi terhadap model pembelajaran mesin dengan menggunakan algoritma Support Vector Machine (SVM) dan Decision Tree (DT), serta menggunakan teknik undersampling untuk menyeimbangkan data. SVM merupakan algoritma *supervised learning*, yang berfungsi untuk menemukan *hyperplane* optimal yang memisahkan data ke dalam kelas-kelas yang berbeda dengan margin maksimum [6]. Sedangkan *Decision Tree* adalah algoritma yang berbentuk struktur pohon yang digunakan untuk pengambilan keputusan dalam konteks klasifikasi dan regresi, model tersebut menyusun data dalam bentuk cabang-cabang yang menggambarkan berbagai kemungkinan keputusan berdasarkan kondisi tertentu [6]. Dalam penelitian ini juga menggunakan undersampling, untuk menyeimbangkan kumpulan data yang tidak merata dengan mempertahankan semua data dikelas minoritas dan memperkecil ukuran kelas mayoritas dalam dataset yang kemudian akan menghasilkan sample sintetis [7].

Terdapat beberapa penelitian yang mengkaji topik serupa dengan penelitian ini. Seperti pada penelitian di tahun 2021 yang ditulis oleh Permana dan Silvanie, penelitian tersebut bertujuan untuk memprediksi penyakit jantung bersarkan 13 kondisi klinis pasien, dataset yang digunakan bersumber dari V. A Medical Center Long Beach and Cleveland Clinic Foundation dengan 303 pasien dan 14 atribut. Kelas target nya terbagi dua yaitu kelas target 1 untuk pasien memiliki penyakit jantung dan 0 untuk pasien tidak memiliki penyakit jantung, kemudian penelitian tersebut menguji metrik kinerja model SVM dengan empat kernel dan menghasilkan nilai akurasi kinerja kernel terbaik dari kernel linear yaitu 90.11% dan disimpulkan bahwa data yang digunakan merupakan bentuk linear [8]. Pada penelitian ditahun yang sama oleh Ali, dkk dalam penelitian tersebut menggunakan kumpulan data dari kaggle yang berisi 1025 catatan pasien, yang terdiri dari 713 pria dan 312 wanita dari berbagai usia, dimana pada data tersebut ada 499 (48,68%) pasien normal dan 526 (51,32%) pasien memiliki penyakit jantung. Ali, dkk menggunakan 6 algoritma yaitu logistic regression (LR), MLPClassifier, Adaboost (ABM1), KNN, decision tree (DT) dan random forest (RF). Dari 6 algoritma yang digunakan, hanya tiga algoritma yang mendapatkan akurasi yang sempurna yaitu KNN, DT dan RF dengan akurasi 100%, serta sensitivitas dan spesifisitas juga mencapai 100%, sehingga pada penelitian tersebut penggunaan algoritma KNN, DT dan RF lebih efektif digunakan untuk memprediksi penyakit jantung [4]. Kemudian pada penelitian oleh putra dkk, melakukan klasifikasi serangan jantung kedalam 8 kategori menggunakan algoritma Naïve Bayes, Decision Tree, dan Support Vector Machine. Pada penelitian tersebut menggunakan data yang imbalance sehingga diperlukan teknik untuk menyeimbangkan data yaitu teknik oversampling. Kemudian pada penelitian tersebut mendapatkan hasil akurasi yang baik dari 2 model yaitu decision tree 98% dan SVM 91% sedangkan naïve bayes hanya menghasilkan akurasi model sebesar 49%, sehingga dapat disimpulkan bahwa model dengan algoritma DT dan SVM lebih baik dalam mengklasifikasikan penyakit jantung dibandingkan NB [9].

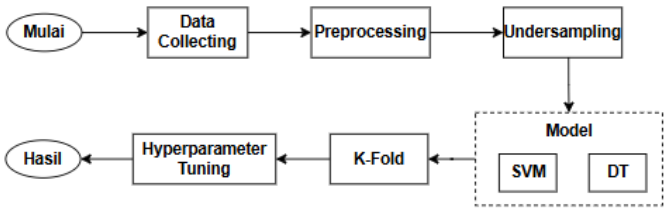
Kemudian penelitian yang dilakukan oleh Putranto, dkk, pada penelitian tersebut melakukan prediksi penyakit jantung menggunakan metode SVM, data yang digunakan terdiri dari 918 pasien dengan 12 indikator penyebab penyakit jantung. Penelitian tersebut mendapatkan akurasi sebesar 85% dengan skor latih sebesar 85% dan skor ujinya 87% sehingga tidak memiliki gap antara skor latih dan skor uji, maka dapat disimpulkan model dalam penelitian tersebut tidak mengalami overfitting maupun underfitting [10]. Selanjutnya pada penelitian oleh Gunawan, dkk, melakukan penelitian dengan dataset berjumlah 918 data dengan 12 fitur. Penelitian tersebut menggunakan decision tree series C4.5 sebagai perbaikan dari ID3 untuk memperoleh perhitungan yang baik dalam tahap sebenarnya pada tools rapidminer version 9.10, pada pengujian matrix mendapatkan nilai akurasi 80.43% dan didapatkan juga error classification sebesar 19,57% [11]. Penelitian oleh Arifuddin, dkk melakukan perbandingan algoritma DT dan SVM menggunakan 5 dataset dari sumber yang berbeda yang digabungkan dengan jumlah 918 data dan 11 fitur, penelitian tersebut membandingkan akurasi model SVM dan dt dengan dan tanpa preprocessing mendapatkan nilai akurasi terbaik pada model SVM tanpa preprocessing, yaitu dengan akurasi 89% dan dt dengan preprocessing sebesar 83% [6]. Penelitian oleh Yusufi, dkk melakukan penelitian dengan membandingkan 7 model akurasi, pada penelitian tersebut menggunakan 299 data dengan 13 atribut dimana kolom target yang dicari dalam pengklasifikasian tersebut berupa risiko kematian. Penelitian tersebut juga menggunakan teknik penyeimbangan data yaitu smote dengan hasil akurasi terbaik dari random forest yaitu 91%, Ensemble voting 89%, SVM 86%, KNN 84%, DT 82%, logistic regression 81%, dan naïve bayes 80% [12]. Sehingga dari penelitian tersebut diperlukan eksplorasi teknik pengolahan data untuk meningkatkan akurasi model algoritma SVM dan DT, dengan mengeksplorasi teknik pra-pemrosesan yang berbeda.

Penelitian-penelitian sebelumnya menunjukkan bahwa penggunaan algoritma support vector machine (SVM) dan decision tree telah memberikan hasil yang baik dalam memprediksi penyakit jantung, serta memberikan kontribusi penting dalam pemodelan prediksi penyakit ini. Namun, masih terdapat kebutuhan eksplorasi lebih lanjut mengenai variasi dataset, teknik pemrosesan data, dan perbandingan metode evaluasi. Oleh karena itu, penelitian ini bertujuan untuk mengisi gap tersebut dengan mengeksplorasi teknik pengolahan data yang berbeda, termasuk penerapan teknik undersampling untuk menyeimbangkan data target, serta menggunakan ukuran data yang lebih besar dibandingkan penelitian sebelumnya. Belum ada penelitian serupa yang menggunakan undersampling untuk menyeimbangkan data target, dan hasil penelitian sebelumnya menunjukkan perbedaan signifikan dalam akurasi antara model algoritma yang digunakan. Penelitian ini juga bertujuan untuk membandingkan performa akurasi SVM dan DT. SVM dievaluasi dengan menggunakan hyperparameter tunning dengan 3 parameter yaitu C, gamma, dan kernel, hal ini dilakukan untuk mengetahui parameter mana yang lebih optimal untuk digunakan pada model. Serta menggunakan teknik k-fold untuk mendapatkan estimasi kinerja model yang lebih akurat. Kemudian untuk model decision tree di lakukan tuning juga dengan GridSearch, parameter yang diuji meliputi criteria, splitter, max_depth, min_samples_split, min_samples_leaf dan max_features. Teknik tersebut digunakan untuk menemukan kombinasi hyperparameter yang optimal berdasarkan kinerja model pada data pelatihan yang divalidasi melalui cross-validation. Setelah proses tuning selesai maka akan didapatkan model dari parameter terbaik. Dengan demikian, diharapkan penelitian ini dapat memberikan pemahaman yang lebih baik kepada masyarakat dalam upaya pengendalian angka kematian akibat penyakit jantung.

II. METODE

A. Kerangka Kerja Penelitian

Penelitian ini dimulai dari pencarian dasar teoritis dari beberapa literatur yang relevan untuk mendukung penelitian dan akan digunakan dalam penelitian ini. Prosedur dalam penelitian ini akan dilakukan seperti pada Gambar 1.



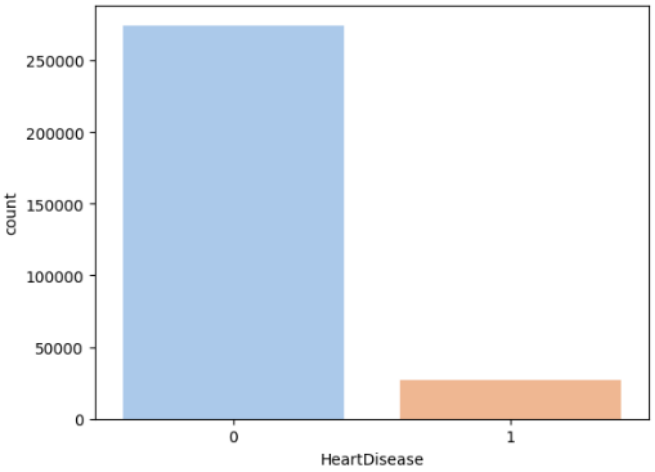
Gambar 1. Tahapan penelitian

Kemudian data set tersebut dilakukan proses preprocessing untuk menghindari adanya data duplikat, missing value, *outlier* dan melakukan standarisasi data. Kemudian setelah didapatkan data yang berkualitas, masuk ke tahap undersampling untuk menyeimbangkan jumlah data, hal ini dilakukan untuk meningkatkan kinerja model dan menghindari adanya bias pada model. Setelah data sudah seimbang dan metode seleksi fitur sudah diterapkan, selanjutnya data tersebut dibagi dengan rasio 80:20 untuk data latih dan uji. Langkah selanjutnya adalah pembuatan model dengan algoritma SVM dan DT untuk mengklasifikasi penyakit jantung berdasarkan pola hidup. Setelah mendapatkan hasil model dari kedua algoritma, tahap selanjutnya adalah melakukan *Hyperparamenter Tuning* untuk mendapatkan parameter terbaik dari masing-masing model. Pada model *support vector machine* (SVM) dan *decision tree* (DT), *tunning* dilakukan dengan teknik GridSearch pada parameter C dan jenis kernel yang digunakan untuk menemukan kombinasi parameter terbaik yang menghasilkan performa optimal. Sementara itu, pada model *decision tree* (DT), *tunning* mencakup beberapa parameter, yaitu *criterion*, *max_depth*, *min_samples_split*, *min_samples_leaf*, *max_feature*, dan *splitter*, untuk menentukan konfigurasi pohon keputusan yang paling efektif.

Evaluasi model dilakukan dengan memanfaatkan teknik *cross-validation* sebagai metode pengukuran performa yang andal, diikuti oleh pengoptimalan model melalui *Hyperparameter Tuning*. Proses ini bertujuan untuk mendapatkan estimasi akurasi yang lebih kuat serta mengevaluasi performa model secara keseluruhan.

B. Data Collecting

Dataset yang digunakan dalam penelitian ini bersumber dari kaggle dengan judul "Heart_2020_cleaned" [13]. Data set ini terdiri dari 17 variabel independent dan 1 variable dependent yaitu *HeartDisease* yang berisikan 2 kelas label yaitu pasien sehat (0) dan pasien memiliki penyakit jantung (1). Dataset ini memiliki total data dengan jumlah 319.795 pasien, jumlah data tersebut memiliki rasio masing-masing 91:9 pada 2 kelas. Dari rasio tersebut menunjukkan jumlah data yang tidak seimbang, dimana 27.373 data termasuk dalam kelas minoritas yaitu pasien mengidap penyakit jantung (1) dan 292.422 data termasuk kedalam kelas mayoritas yaitu pasien sehat (0), dapat dilihat pada gambar 2.



Gambar 2. Bar chart distribusi class pasien

Dengan total jumlah data pada 2 kelas tersebut menunjukkan bahwa kelas minoritas memiliki jumlah yang jauh lebih sedikit dibandingkan dengan kelas mayoritas, sehingga diperlukan penggunaan teknik penyeimbangan data pada penelitian ini. Fitur-fitur pendukung lainnya mencakup 17 variabel yang terkait dengan pola hidup, kondisi pasien dan kondisi mental pasien. Variable-variabel ini meliputi informasi kebiasaan pasien seperti merokok, konsumsi alkohol, aktivitas fisik, bmi, diabetes, riwayat penyakit kronis, serta kondisi kesehatan mental. Rincian lengkap mengenai fitur dengan tipe datanya disajikan dalam tabel 1 untuk memberikan gambaran yang komprehensif mengenai data yang digunakan dalam analisis prediksi pada penelitian ini.

TABEL I
FITUR PADA DATASET YANG DIGUNAKAN

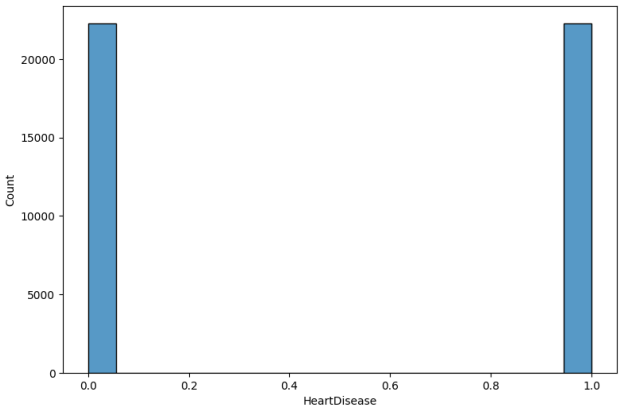
Nama Fitur	Tipe Data
HeartDisease	Object
BMI	Float
Smoking	Object
AlcoholDrinking	Object
Stroke	Object
PhysicalHealth	Int
MentalHealth	Int
DiffWalking	Object
AgeCategory	Object
Race	Object
Diabetic	Object
PhysicalActivity	Object
GenHealth	Object
SleepTime	Int
Asthma	Object
KidneyDisease	Object
Skin Cancer	Object

C. Preprocessing Data

Tahapan awal sebelum menggunakan data dalam pembuatan sebuah model adalah preprocessing data, hal ini dilakukan agar data yang diolah terjaga kualitas dan konsistensinya [14]. Preprocessing data bertujuan untuk meningkatkan kualitas data, menghilangkan kesalahan, dan mengubahnya menjadi format yang lebih cocok untuk pemodelan atau analisis [15]. Langkah-langkah dalam data preprocessing pada penelitian ini, terdiri dari handling outlier, missing value, cleaning data, scaling data, encoding categorial variables, dan splitting data. Dengan melakukan langkah ini, model dapat meningkatkan akurasi dan keterandalan hasil analisa serta meminimalkan risiko bias atau kesalahan yang mungkin terjadi, karena data yang digunakan dalam penelitian dapat dipastikan kebersihan, kualitas serta integritas nya [15].

D. Undersampling

Tidak seimbangnya data dalam penerapan machine learning diakibatkan oleh distribusi kelas yang tidak merata dalam sebuah kumpulan data [16]. Ketidakseimbangan data menjadi salah satu tantangan terbuka dari pembelajaran mesin kontemporer, yang mempengaruhi sebagian besar masalah klasifikasi di dunia nyata, hal ini terjadi setiap kali menemukan kelas data mayoritas karena sistem hanya mengamati data dengan jumlah pengamatan yang lebih tinggi daripada kelas minoritas lainnya [17]. Dataset pada data mining yang memiliki jumlah data yang tidak seimbang dapat menyebabkan misleading atau kesalahan dalam hasil pemodelan, dimana data kelas minoritas sering diklasifikasikan sebagai kelas mayoritas [18]. Untuk mencapai akurasi tinggi dalam pengklasifikasian data, terkadang model akan cenderung mengklasifikasikan seluruh sampel ke dalam kelas mayoritas, dan hal ini akan menyebabkan semua sampel pada kelas minoritas diklasifisikasikan sebagai data yang salah, fenomena seperti ini disebut sebagai "paradoks akurasi" dimana akurasi yang tinggi tidak mencerminkan performa model yang sebenar-benarnya dalam menangani ketidakseimbangan kelas [19]. Salah satu teknik yang bisa digunakan untuk menyeimbangkan data adalah undersampling. Undersampling adalah metode yang digunakan untuk mengambil beberapa data mayoritas sehingga jumlah data mayoritas sama besar dengan jumlah data minoritas [18]. Teknik undersampling yang digunakan dalam penelitian ini adalah *random undersampling. Random undersampling* bekerja dengan cara mengurangi jumlah sampel pada kelas mayoritas secara acak hingga mencapai proporsi yang lebih seimbang dengan kelas minoritas [20]. Pada dataset yang digunakan dalam penelitian ini ditemukan rasio ketidakseimbangan antara class pasien sehat (0) 92% dan pasien menderita penyakit jantung (1) 8%. Maka untuk mendapatkan akurasi yang optimal, pada penelitian ini menggunakan teknik *random undersampling* untuk menyeimbangkan jumlah data target pada 2 kelas. Hasil dari data yang sudah seimbang, dapat dilihat pada gambar 3.



Gambar 3. Jumlah data setelah *undersampling*

E. Algoritma Model

1) Support Vector Machine (SVM): Algoritma support vector machine adalah metode klasifikasi untuk data linear dan non linear. SVM merupakan representasi dari kelas yang berbeda dalam ruang multidimensi [21]. Tujuannya adalah untuk membagi dataset ke dalam kelas-kelas untuk menemukan hyperplane yang optimal untuk memisahkan kelas-kelas dalam data dengan margin maksimum. Kelebihan dari algoritma ini adalah memiliki daya komputasi yang cepat [10]. Fungsi utama dari pengklasifikasi SVM adalah untuk menyelesaikan pemilihan subset fitur dengan penyesuaian parameter [14]. Parameter yang digunakan adalah C (regularization parameter), Gamma (kernel coefficient), dan probabilitas . Parameter tersebut sering juga disebut sebagai 'hyperparameter' [22]. Hyperparameter C pada model SVM berfungsi untuk mengatur bobot penalti terhadap kesalahan klasifikasi selama pelatihan, di mana nilai C yang rendah menghasilkan margin pemisah (hyperplane) yang lebih lebar dengan toleransi kesalahan yang lebih tinggi, sedangkan nilai C yang tinggi mempersempit margin untuk meningkatkan akurasi dan berisiko overfitting, sementara parameter gamma mengontrol pengaruh setiap sampel terhadap model dengan nilai gamma yang lebih tinggi menciptakan pengaruh local yang lebih kuat, dan pengaturan probabilitas memungkinkan perkiraan probabilitas per kelas dengan kernel default RBF (Radial Basis Function) dalam proses perhitungannya [23].

2) Decision Tree (DT): Decision Tree adalah model pembelajaran mesin yang digunakan untuk klasifikasi dan regresi. DT akan menghasilkan struktur pohon yang terdiri dari simpul (nodes) dan cabang (branches), struktur pohon ini disebut pohon keputusan. Pohon keputusan merupakan struktur seperti diagram alur yang digunakan sebagai alat pendukung keputusan dalam masalah klasifikasi dan regresi uuntuk membantu membangun model yang prediktif otomatis, model pohon keputusan ini dihasilkan melalui penempatan atribut terbaik dari himpunan data di akar pohon [24]. Decision tree yang digunakan pada penelitian ini adalah classification and regression trees (CART). Pada penelitian

yang dilakukan oleh Ozcan dan Peker merekomendasikan penggunaan decision tree CART ini untuk digunakan, karena algoritma CART bekerja dengan mengekstraksi aturan keputusan dari fitur dan membangun model untuk memprediksi nilai target [25]. Nilai ini merupakan metode model prediktif yang sering digunakan pada bidang ilmu statistic, data mining, dan machine learning. Pada bentuk pohon, label kelas terwakili oleh daun serta label kelas terwakili oleh cabang yang terkoneksi fitur [26]. Pada penggunaan algoritma decision tree ini juga menggunakan hyperparameter tuning untuk mengoptimalkan hasil model. Sebagian besar penelitian menyelidiki berbagai cara untuk mengoptimalkan hyperparameter dalam model machine learning untuk mengatasi masalah klasifikasi tertentu [27]. Pada algoritma pohon keputusan, penyetelan hiperparameter seperti tree_depth berperan penting dalam menjaga keseimbangan antara performa model dan risiko overfitting, di mana nilai sedang sering memberikan hasil optimal sementara nilai yang terlalu tinggi dapat menurunkan kinerja, sehingga penyetelan dapat dilakukan secara manual atau dengan strategi berbasis data untuk meminimalkan kesalahan prediksi melalui pencarian parameter [28].

Pada algoritma decision tree ini, menyetel beberapa hyperparameter utama untuk mengatur performa dan kompleksitas model, antara lain:

- Hyperparameter criterion menentukan fungsi pemisah yang digunakan di setiap node, di mana opsi gini lebih cepat karena menghitung impuritas Gini sebagai default, sementara entropy mengandalkan informasi gain, yang lebih lambat namun kadang menghasilkan pemisahan lebih baik.
- Hyperparameter splitter menetapkan strategi pemisahan node, dengan opsi best untuk memilih pemisahan optimal atau random untuk mempercepat pelatihan dan mengurangi risiko overfitting.
- Batas kedalaman pohon diatur oleh max_depth, dengan pilihan None untuk kedalaman tak terbatas atau nilai tertentu yang mencegah pohon menjadi terlalu kompleks.
- Parameter min_samples_split menetapkan jumlah minimum sampel di setiap node untuk memungkinkan pembagian lebih lanjut,
- Min_samples_leaf menentukan jumlah minimum sampel yang harus dimiliki sebuah daun, membantu mengontrol ukuran daun dan kesederhanaan model.
- Max_features menentukan jumlah maksimum fitur yang dipertimbangkan dalam pemisahan terbaik; pilihan seperti sqrt yang membantu mempercepat proses pelatihan dan mengurangi risiko overfitting.

F. Evaluasi

Tahapan evaluasi digunakan untuk mengukur performa model yang dihasilkan. Peneliti menggunakan confusion matrix untuk melakukan analisa performa dari model [10]. Dalam pengembangan machine learning, terdapat evaluasi yang bisa digunakan untuk menilai kinerja model. Salah satu teknik yang populer adalah k-fold cross validation. Dengan teknik ini data akan di evaluasi dengan k-fold cross validation sebanyak k nya adalah 10 kali. Dengan demikian data latih akan dibagi menjadi 10 bagian yang mana 10 bagian ini berarti terdapat 1 bagian yang digunakan sebagai percobaan dan 9 bagian lainnya sebagai data latih [9]. Cara kerja dari teknik ini adalah untuk memisahkan data K menjadi data uji dan data latih. Pengujian menggunakan dua teknik ini untuk mengurangi bias yang melekat pada data acak [29]. Dalam penelitian ini, membagi kumpulan data menjadi 10 bagian (lipatan) yang berbeda dengan ukuran yang sama dan melatih serta menguji model sebanyak 10 kali. Selain itu *hyperparameter tuning* juga proses yang penting dalam tahap evaluasi model, dengan tujuan menemukan kombinasi yang terbaik yang dapat meningkatkan kinerja model. Dalam *hyperparameter tuning* menggunakan beberapa parameter untuk dibandingkan dan nantinya akan menemukan hasil akurasi terbaik, parameter model ini menunjukkan bagaimana data input diubah menjadi output yang diinginkan, sedangkan hyperparameter menunjukkan bagaimana struktur dari model, performa model machine learning ini dapat berubah drastis tergantung pada pilihan hyperparameter nya [28]. Pada penelitian ini menerapkan metode *tuning grid search*, pada algoritma SVM parameter yang ditetapkan adalah C dan menggunakan kernel linear dan rbf. Kemudian untuk algoritma decision tree menggunakan beberapa parameter grid, yaitu criterion, splitter, max_depth, min_samples_split, min_samples_leaf, dan max_features, untuk nilai yang diterapkan pada tiap parameter dapat dilihat pada tabel 2.

TABEL II
PARAMETER YANG DIGUNAKAN PADA TUNING

Algoritma	Parameter	Nilai
SVM	Kernel	Linear ; RBF
	C	0.01, 0.1, 1
	gamma	0.01, 0.1, 1
DT	Criterion	Gini, entropy
	Max_depth	None, 5, 10, 20, 30
	Min_samples_split	2, 5,10
	Min_samples_leaf	1,2,4
	Splitter	'best', 'random'
	Max_features	None, 'sqrt', 'log2'

III. HASIL DAN PEMBAHASAN

A. Preprocessing Data

1. Data Cleaning: Proses ini merupakan proses pembersihan data terhadap data duplikat, missing value dan outlier. Dari dataset yang digunakan memiliki jumlah data sebanyak 319.795, didalam dataset tersebut terdapat data duplikat sebesar 18.078 data, kemudian data duplikat tersebut dihapus dan jumlah datanya menjadi 301.717. Pada pengecekan missing value tidak ditemukan adanya data yang hilang.

2. *Penanganan Outlier:* Penanganan outlier dilakukan menggunakan dua metode statistik, yaitu Z-Score dan Interquartile Range (IQR), untuk mengidentifikasi data ekstrem yang berpotensi mendistorsi analisis. Z-Score digunakan untuk mendeteksi nilai yang berada lebih dari 3 standar deviasi dari rata-rata, sedangkan IQR mengidentifikasi data di luar rentang. Hasilnya sebanyak 11.741 data outlier dihapus untuk menjaga kualitas analisis tanpa mengurangi representative dataset. Setelah penghapusan, jumlah data yang tersisa adalah 289.976, yang tetap memadai untuk analisis lanjutan dengan hasil yang andal dan valid.

3. *Transformation Data:* Transformasi data dilakukan untuk mengubah tipe data dari object menjadi numerik untuk mendukung kualitas hasil analisis, karena Algoritma SVM dan decision tree CART memerlukan data dalam bentuk numerik untuk bekerja optimal dan membedakan pola atau batasan kelas pada dataset dengan lebih baik. Pada transformasi data ini terdapat 14 tipe data object, yang kemudian ditransformasikan kedalam tipe data numerik.

B. *Undersampling*

Pada penelitian ini ditemukan distribusi data target yang tidak seimbang, dengan jumlah data pasien yang tidak mengidap penyakit jantung (0) adalah 250.132, sedangkan data pasien yang mengidap penyakit jantung (1) sebanyak hanya 22.293. Untuk mengatasi ketidakseimbangan ini, dilakukan penyeimbangan data terhadap kelas mayoritas (0), jenis undersampling yang digunakan pada penelitian ini adalah *random undersampling*, yang bekerja dengan mengurangi jumlah sampel mayoritas secara acak, sehingga jumlah sampel di kedua kelas menjadi seimbang, yaitu masing-masing kelas memiliki jumlah data 22.293. Hasil akhir dari proses ini adalah jumlah data dengan total 44.586 data, dapat dilihat pada tabel 3.

TABEL III
JUMLAH DATA UNDERSAMPLING

Data	Jumlah Data		Total Data
	Kelas 0	Kelas 1	
Sebelum Undersampling	250.132	22.293	272.425
Setelah Undersampling	22.293	22.293	44.586

C. *Modeling*

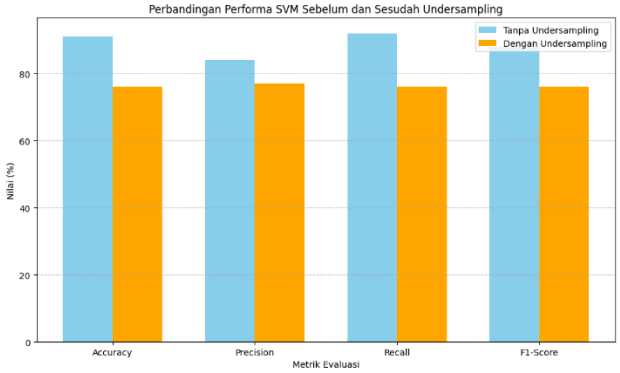
Dalam tahap modeling, penelitian ini menggunakan dua algoritma machine learning, yaitu support vector machine (SVM) dan decision tree classification and regression trees (DTCART) yang bertujuan untuk membandingkan algoritma mana yang lebih baik dalam sistem prediksi penyakit jantung yang akurat. Split data pada penelitian ini dilakukan dengan perbanding 80:20. Model ini melakukan perbandingan nilai akurasi dari model yang menggunakan teknik undersampling dan tanpa undersampling, kemudian model dengan akurasi terbaik akan diuji menggunakan hyperparameter tuning gridsearch untuk mencari kombinasi nilai parameter terbaik serta mengoptimalkan performa model. Parameter yang digunakan pada tuning model svm adalah C, gamma, dan kernel. Sedangkan pada decision tree menggunakan beberapa parameter yaitu criterion, splitter, max_depth, min_samples_split, min_samples_leaf, dan max_features. Kemudian pada penelitian ini juga menerapkan cross validation dengan nilai 10 guna mengevaluasi performa model dengan membagi data menjadi k bagian (folds) dengan nilai k adalah 10, hal ini akan membantu untuk mendapatkan estimasi performa model yang lebih stabil dan mengurangi bias karena variasi data.

Pada modeling SVM dan DT dilakukan perbandingan terhadap nilai akurasi, precision, recall dan f1-score pada model SVM dan DT, pada modelling ini menggunakan pembagian data 80:20. Persentase dari masing-masing metrik nilai disajikan dalam tabel 4, berikut.

TABEL IV
PERBANDINGAN NILAI AKURASI SEBELUM DAN SETELAH DI TERAPKAN UNDERSAMPLING PADA SVM DAN DT

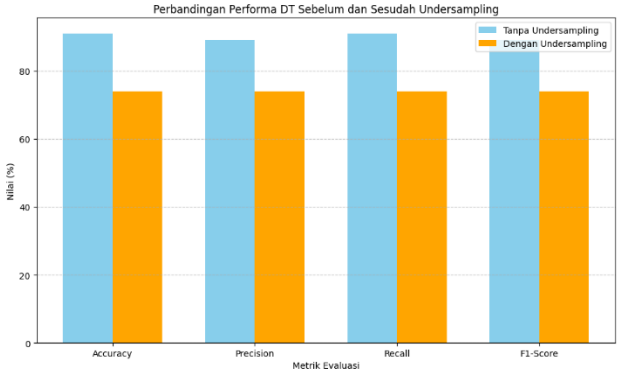
Metode	Nilai			
	Accuracy	Precision	Recall	F1-Score
SVM Tanpa Undersampling	91%	84%	92%	88%
SVM Dengan Undersampling	76%	77%	76%	76%
DT Tanpa Undersampling	91%	89%	91%	89%
DT Dengan Undersampling	74%	74%	74%	74%

Hasil model menunjukkan bahwa penerapan undersampling memberikan pengaruh yang signifikan terhadap performa model dalam mengenali class minoritas. Sebelum diterapkan undersampling, model tanpa undersampling memiliki akurasi sebesar 91% dengan precision, recall, dan f1-score masing-masing 84%, 92%, dan 88%. Namun, model ini gagal mengenali kelas minoritas (kelas 1), sebagaimana terlihat dari nilai recall masing masing class yang didapatkan, yaitu sebesar 0% untuk kelas tersebut, yang mengindikasikan bias model terhadap kelas mayoritas (kelas 0). Setelah diterapkan undersampling, performa model dalam mengenali kedua kelas menjadi lebih seimbang, meskipun akurasi secara keseluruhan menurun menjadi 76%. Nilai precision, recall dan f1-score masing-masing berubah menjadi 77%, 76%, dan 76%. Hal ini menunjukkan bahwa model dengan undersampling mampu mempelajari pola dari kedua kelas secara lebih baik dibandingkan model tanpa undersampling, yang hanya mempelajari kelas mayoritas.



Gambar 4. Grafik persentase precision, recall, dan F1-score SVM

Hasil model menunjukkan pengaruh signifikan dari penerapan teknik undersampling terhadap performa algoritma SVM. Berdasarkan gambar 4 dan tabel 4, terlihat bahwa SVM dengan undersampling menunjukkan kinerja yang paling stabil dengan nilai *precision*, *recall*, dan F1-score yang saling mendekati, menandakan keseimbangan yang baik antara kemampuan mengenali kelas positif dan akurasi prediksi. Sebaliknya, sebelum diterapkan undersampling, semua metrik pada SVM mengalami penurunan yang konsisten.



Gambar 5. Grafik persentase precision, recall, dan F1-score DT

Dari gamber 5, didapatkan bahwa performa model *decision tree* CART (DT) bervariasi pada penggunaan teknik undersampling. Tanpa undersampling, model menghasilkan akurasi 91% dengan precision, recall, dan F1-score masing-masing sebesar 89%, 91%, dan 89%, menunjukkan performa tinggi namun terdapat bias pada kelas mayoritas. Setelah menerapkan undersampling, akurasinya menurun menjadi 74%, namun performa pada kelas minoritas meningkat dengan precision dan recall masing-masing mencapai 73% dan 77%, mencerminkan keseimbangan yang lebih baik.

D. Hyperparameter Tuning

Dari nilai model yang sudah didapatkan, penerapan undersampling berhasil mengatasi bias terhadap class mayoritas, meskipun dengan kompromi pada akurasi keseluruhan. Hal ini menegaskan pentingnya pemilihan strategi penyeimbangan data untuk memastikan model dapat memberikan hasil yang dapat diandalkan dalam scenario dengan data tidak seimbang, seperti pada kasus ini. Maka selanjutnya dilakukan uji tuning pada model dengan *undersampling* dan pada model tanpa *undersampling*, proses *tuning* dilakukan dengan menggunakan metode *Grid Search* untuk mengoptimalkan parameter kunci, termasuk nilai C, gamma dan kernel, yang kemungkinan berpengaruh langsung pada kompleksitas model dan kemampuannya dalam mengklasifikasikan data. Pada model SVM dan DT, *best parameter tuning* mendapatkan hasil disajikan pada tabel 5 sebagai berikut:

TABEL V
BEST PARAMETER TUNING DENGAN *GRID SEARCH*

Metode	Parameter	Hasil Grid Search
Support Vector Machine tanpa undersampling	Kernel: Linear , C: 0.01	92%
Support Vector Machine dengan undersampling	Kernel: Linear, C: 1	76%
Decision Tree CART tanpa undersampling	criterion: gini, splitter: random, max_depth: 5, min_samples_split: 10, min_samples_leaf: 2, max_features: None	91%
Decision Tree CART dengan undersampling	criterion: entropy, splitter: random, max_depth: 10, min_samples_split: 5, min_samples_leaf: 4, max_features: None	75%

Pada Support Vector Machine (SVM) tanpa undersampling, parameter terbaik diperoleh dengan kernel *linear* dan nilai *C=0.01*. Hasil pengujian menggunakan *k-fold cross-validation* menunjukkan akurasi sebesar 92%, dengan *precision* 84%, *recall* 92%, dan *F1-score* 84%. Model ini menunjukkan kinerja yang sangat baik dalam mengenali kelas mayoritas (class 0), namun *precision* yang sedikit lebih rendah menunjukkan bahwa model memiliki kesalahan dalam memprediksi kelas minoritas (class 1), yang secara proporsi jauh lebih sedikit. Hal ini mengindikasikan bahwa meskipun akurasi tinggi, model SVM tanpa *undersampling* kurang optimal dalam menangani ketidakseimbangan kelas. Pada SVM dengan teknik undersampling, parameter terbaik diperoleh dengan kernel *linear* dan nilai *C*=1. Hasil pengujian menghasilkan akurasi sebesar 76%, dengan *precision* 77%, *recall* 76%, dan *F1-score* 76%. Teknik undersampling berhasil menyeimbangkan distribusi kelas, terlihat dari *precision* dan *recall* yang lebih konsisten, namun informasi yang lebih terbatas akibat pengurangan data memengaruhi kemampuan model untuk membuat prediksi yang lebih akurat, terutama pada kelas mayoritas. Penurunan performa

ini menunjukkan bahwa SVM lebih efektif pada data yang tidak diubah distribusinya.

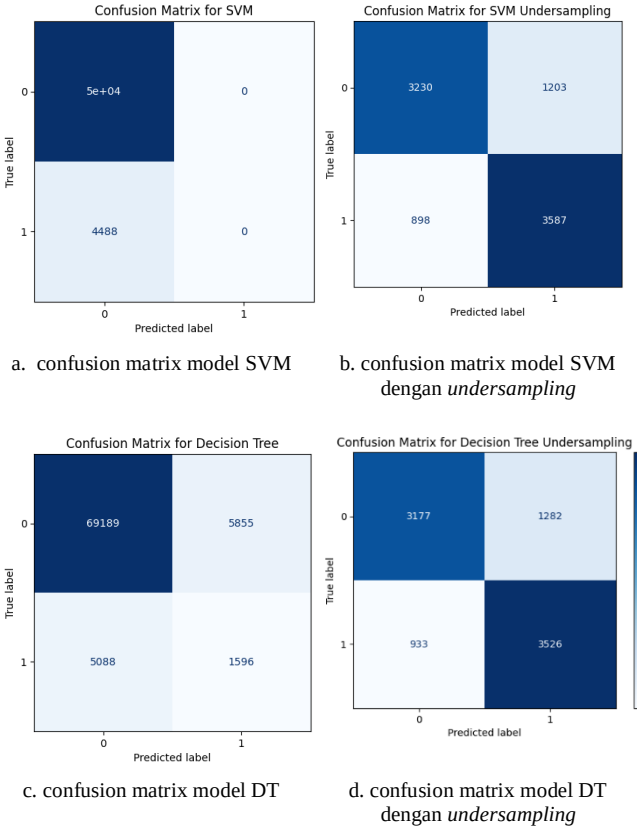
Pada Decision Tree (DT) tanpa undersampling, parameter terbaik diperoleh dengan *criterion=gini*, *splitter=random*, *max_depth=5*, *min_samples_split=10*, *min_samples_leaf=2*, dan *max_features=None*. Model ini menghasilkan akurasi sebesar 91%, yang menunjukkan bahwa model mampu mempelajari pola penting pada data tanpa mempelajari terlalu banyak detail, sehingga dapat menjaga keseimbangan antara kompleksitas model dan performa. Sebaliknya, pada DT dengan teknik undersampling, parameter terbaik adalah *criterion=entropy*, *splitter=random*, *max_depth=10*, *min_samples_split=5*, *min_samples_leaf=4*, dan *max_features=None*. Hasil pengujian menunjukkan akurasi sebesar 75%. *Max_depth* yang lebih besar pada model DT dengan *undersampling* dibandingkan model tanpa undersampling mengindikasikan bahwa model mencoba mempelajari detail lebih dalam dari data pelatihan yang telah diundersampling. Hal ini menyebabkan model *overfitting* terhadap data pelatihan, yang berdampak pada penurunan performa ketika diuji pada metode yang berbeda. kemudian dilakukan juga pengujian terhadap model SVM dan DT dengan masing-masing empat perlakuan berbeda, sebagai berikut pada tabel 6.

TABEL VI
PENGUJIAN CROSS VALIDATION BERBAGAI METODE PENGUJIAN

Model	Perlakuan	Nilai Akurasi [10-fold]
SVM	Tanpa Undersampling	91%
SVM	Tanpa Undersampling + Tuning	92%
SVM	Dengan Undersampling	75%
SVM	Dengan Undersampling + Tuning	76%
DT	Tanpa Undersampling	92%
DT	Tanpa Undersampling + Tuning	92%
DT	Dengan Undersampling	74%
DT	Dengan Undersampling + Tuning	75%

Pada model Support Vector Machine (SVM), hasil pengujian tanpa teknik undersampling menunjukkan akurasi sebesar 91% dengan metode 10-fold cross-validation. Setelah dilakukan tuning hyperparameter, akurasi meningkat menjadi 92%, yang menunjukkan bahwa pengaturan parameter seperti nilai C dan kernel yang optimal dapat meningkatkan performa model. Sebaliknya, pada model SVM dengan teknik undersampling, akurasi menurun menjadi 75%, yang mengindikasikan bahwa undersampling mengurangi informasi dalam data sehingga memengaruhi kemampuan model dalam mempelajari pola. Setelah tuning, akurasi model SVM dengan undersampling meningkat sedikit menjadi 76%, tetapi hasil ini tetap lebih rendah dibandingkan model tanpa undersampling. Hal ini menunjukkan bahwa SVM lebih cocok untuk data dengan distribusi kelas asli tanpa pengurangan data.

Kemudian pada model Decision Tree (DT), hasil tanpa teknik undersampling menunjukkan performa yang serupa dengan SVM, dengan akurasi mencapai 92% baik sebelum maupun sesudah tuning. Hal ini mengindikasikan bahwa model Decision Tree sudah cukup optimal dalam mempelajari pola data tanpa membutuhkan banyak penyesuaian parameter. Namun, pada model DT dengan teknik undersampling, akurasi turun menjadi 74%, serupa dengan yang terjadi pada SVM. Setelah tuning, akurasi meningkat sedikit menjadi 75%, tetapi tetap lebih rendah dibandingkan model tanpa undersampling. Penurunan performa ini menunjukkan bahwa Decision Tree juga kurang efektif dalam menangani data yang informasinya yang dibatasi oleh teknik undersampling. Dengan hasil ini, dapat disimpulkan bahwa hyperparameter tuning pada model SVM berkontribusi secara signifikan terhadap peningkatan performa, membuktikan pentingnya proses tuning dalam pengembangan model machine learning yang efektif. Tuning parameter seperti nilai C dan kernel mampu meningkatkan akurasi model, terutama pada data dengan distribusi kelas asli (tanpa undersampling).



a. confusion matrix model SVM

b. confusion matrix model SVM dengan *undersampling*

c. confusion matrix model DT

d. confusion matrix model DT dengan *undersampling*

Gambar 6. Confusion matrix dari masing-masing model

Pada gambar 6a, yaitu model SVM tanpa teknik undersampling, terlihat bahwa model memprediksi class mayoritas (0) dengan sangat baik, menghasilkan *true negatives* (TN) sebanyak 50.000. Namun model tidak dapat mengenali satu pun data class minoritas (1) dengan benar, yang terlihat dari nilai *true positives* (TP) sebanyak 0. Kemudian sebanyak 4.488 data pada class minoritas salah

diprediksi sebagai class mayoritas *false negative* (FN). Hal ini menunjukkan bahwa model SVM tanpa *undersampling* sepenuhnya bias terhadap class mayoritas dan sama sekali tidak mampu membaca pola dari data class minoritas akibat ketidakseimbangan data. Pada gambar 6b, model SVM dengan *undersampling* distribusi class menjadi lebih seimbang, sehingga model SVM menunjukkan peningkatan kemampuan dalam mengenali class minoritas. Hal ini terlihat dari nilai *true positive* (TP) sebesar 3.587, yang jauh lebih baik dibandingkan tanpa *undersampling*. Selain itu, jumlah *false negative* (FN) juga berkurang menjadi 898. Namun, terjadi peningkatan *false positive* (FP) sebesar 1.203 menunjukkan bahwa beberapa data class mayoritas salah diprediksi sebagai class minoritas. Pada gambar 6c, model Decision Tree tanpa teknik undersampling, performa terhadap kelas mayoritas cukup baik, menghasilkan True Negatives (TN) sebesar 69,189. Selain itu, model juga mampu mengenali sebagian data kelas minoritas, yang ditunjukkan oleh nilai True Positives (TP) sebesar 1,596. Namun, jumlah False Negatives (FN) masih cukup tinggi, yaitu 5,088, menunjukkan bahwa model masih mengalami kesulitan dalam membaca pola data kelas minoritas. Sedangkan pada model DT dengan undesampling, pada gambar 6d menunjukkan peningkatan yang signifikan dalam mengenali kelas minoritas. Hal ini terlihat dari nilai True Positives (TP) sebesar 3,526, yang hampir setara dengan hasil SVM dengan undersampling. Selain itu, jumlah False Negatives (FN) menurun menjadi 933, menandakan bahwa model lebih baik dalam mengidentifikasi data kelas minoritas dibandingkan versi tanpa undersampling. Namun, sama seperti pada SVM, teknik undersampling juga menyebabkan peningkatan jumlah False Positives (FP) menjadi 1,282, sehingga memengaruhi kemampuan model dalam mengenali kelas mayoritas secara akurat. Akibatnya, True Negatives (TN) juga menurun menjadi 3,177, mengindikasikan adanya penurunan performa keseluruhan akibat hilangnya data mayoritas.

IV. KESIMPULAN

Penelitian ini mengevaluasi performa SVM dan Decision Tree pada prediksi penyakit jantung, dengan fokus pada efek handling imbalance melalui *random undersampling* dan *hyperparameter tuning*. Hasil menunjukkan bahwa teknik *random undersampling* berhasil meningkatkan keterbacaan kelas minoritas pada kedua model, dengan peningkatan nilai *true positive* yang signifikan. Namun, teknik ini mengurangi akurasi keseluruhan akibat hilangnya informasi dari kelas mayoritas. Hal ini menunjukkan bahwa teknik *handling imbalance* seperti *random undersampling* penting diterapkan untuk meningkatkan keseimbangan prediksi pada dataset yang tidak seimbang, tetapi kurang optimal jika digunakan tanpa pendekatan tambahan. *Hyperparameter tuning* memberikan dampak yang signifikan pada performa model, terutama pada SVM. Dengan *tuning*, akurasi SVM meningkat menjadi 92% pada data tanpa *undersampling*, hal ini menunjukkan bahwa pengaturan parameter seperti nilai C dan kernel yang optimal dapat membantu model menangkap pola yang lebih kompleks. Sedangkan pada model *decision tree* tuning tidak memberikan perubahan signifikan karena model sudah cukup optimal di kondisi sebelum tuning. SVM menunjukkan akurasi tertingginya pada data tanpa undersampling setelah tuning, membuktikan keunggulannya dalam menangkap pola yang lebih kompleks. Namun, model ini memiliki keterbatasan dalam mengenali kelas minoritas tanpa teknik *random undersampling*. Sebaliknya, Decision Tree lebih fleksibel dalam mengenali pola pada kelas minoritas, meskipun akurasinya menurun saat menggunakan teknik undersampling. Penelitian ini diharapkan dapat diterapkan sebagai system prediksi dini penyakit jantung, namun hasil prediksi model harus dikombinasikan dengan pemeriksaan lebih lanjut oleh tenaga medis untuk mendapatkan prediksi penyakit yang akurat. karena mengingat dataset yang digunakan dalam penelitian ini adalah dataset sekunder yang diambil dari satu sumber dataset, sehingga kemungkinan dataset pada penelitian ini hanya mewakili sebagian populasi dari pasien. Implementasi system berbasis *machine learning* ini dapat menjadi langkah awal dalam mendukung peningkatan layanan kesehatan, terutama pada deteksi dini penyakit jantung.

UCAPAN TERIMA KASIH

Penulis mengucapkan terimakasih kepada dosen pembimbing saya yang mendukung dan selalu memberikan arahan dalam penulisan jurnal ini, serta Luye Zhang pemilik dataset Heart_2020_cleaned atas kesediaannya untuk menyediakan data penelitian yang sangat membantu dalam penelitian ini, karena data ini memungkinkan penulis menyelesaikan penelitian ini dan dapat mengembangkan model prediksi jantung berdasarkan pola hidup yang akurat dan dapat diandalkan.

DAFTAR PUSTAKA

[1] "10 penyebab kematian teratas," World Health Organization. Accessed: Oct. 19, 2024. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death

[2] "Penyakit Kardiovaskular (PKV)," World Health Organization.

[3] W. Hanifah, W. Septi Oktavia, and dan Hoirun Nisa, "Faktor gaya Hidup dan Penyakit Jantung Koroner: Review Sistematik Pada Orang Dewasa di Indonesia (Lifestyle Factors and Coronary Heart Disease: A Systematic Review Among Indonesian Adults))," vol. 44, no. 1, pp. 45–58, 2021.

[4] M. M. Ali, B. K. Paul, K. Ahmed, F. M. Bui, J. M. W. Quinn, and M. A. Moni, "Heart disease prediction using supervised machine learning algorithms: Performance analysis and comparison," *Comput Biol Med*, vol. 136, Sep. 2021, doi: 10.1016/j.compbiomed.2021.104672.

[5] J. Khatib Sulaiman, A. A. Mizwar Rahim, I. Yanuar Risca Pratiwi, M. Ainul Fikri, and U. Amikom Yogyakarta, "Klasifikasi Penyakit Jantung Menggunakan Metode Synthetic Minority Over-Sampling Technique Dan Random Forest Clasifier," *Indonesian Journal of Computer Science Attribution*, vol. 12, no. 5, pp. 2023–2995.

[6] A. Arifuddin, G. S. Buana, R. A. Vinarti, and A. Djunaidy, "Performance Comparison of Decision Tree and Support Vector Machine Algorithms for Heart Failure Prediction," *Procedia*

Comput Sci, vol. 234, pp. 628–636, 2024, doi: 10.1016/j.procs.2024.03.048.

[7] A. Indrawati, "Penerapan Teknik Kombinasi Oversampling dan Undersampling Untuk Mengatasi Permasalahan Imbalanced Dataset," *Jurnal Informatika dan Komputer) Akreditasi KEMENRISTEKDIKTI*, vol. 4, no. 1, 2021, doi: 10.33387/jiko.

[8] A. Silvanie and D. S. Permana, "Prediksi Penyakit Jantung Menggunakan Support Vector Machine dan Python Pada Basis Data Pasien di Cleveland," DKI Jakarta, Apr. 2021.

[9] N. Yudistira and A. F. Putra, "Algoritma Decision Tree Dan Smote Untuk Klasifikasi Serangan Jantung Miokarditis Yang Imbalance," *Jurnal Litbang Edusaintech*, vol. 2, no. 2, pp. 112–122, Dec. 2021, doi: 10.51402/jle.v2i2.48.

[10] A. Putranto, N. L. Azizah, I. Ratna, I. Astutik, F. Sains, and D. Teknologi, "Sistem Prediksi Penyakit Jantung Berbasis Web Menggunakan Metode SVM dan Framework Streamlit," Sidoarjo Indonesia, Apr. 2023. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/heart+disease

[11] I. M. Agus Oka Gunawan, I. D. A. Indah Saraswati, I. D. G. Riswana Agung, and I. P. Eka Putra, "Klasifikasi Penyakit Jantung Menggunakan Algoritma Decision Tree Series C4.5 Dengan Rapidminer," *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 5, no. 2, pp. 73–83, Apr. 2023, doi: 10.47233/jteksis.v5i2.775.

[12] A. H. Yusufi, A. Kharisma, A. D. Adinata, D. F. Ramzy, and M. M. Santoni, "Prediksi Resiko Kematian Pada Penderita Penyakit Kadiovaskular Menggunakan Metode Ensemble Learning," 2022.

[13] Luye Zhang, "Heart_2020_cleaned," kaggle. Accessed: Oct. 26, 2024. [Online]. Available: https://www.kaggle.com/datasets/luyezhang/heart-2020-cleaned/data

[14] S. N. N. Arif, A. M. Siregar, S. Faisal, and A. R. Juwita, "Klasifikasi Penyakit Serangan Jantung Menggunakan Metode Machine Learning K-Nearest Neighbors (KNN) dan Support Vector Machine (SVM)," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 8, no. 3, p. 1617, Jul. 2024, doi: 10.30865/mib.v8i3.7844.

[15] S. P. R. Yulianto, A. Z. Fanani, A. Affandy, and M. I. Aziz, "Analisis Metode Smoote pada Klasifikasi Penyakit Jantung Berbasis Random Forest Tree," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 8, no. 3, p. 1460, Jul. 2024, doi: 10.30865/mib.v8i3.7712.

[16] R. Mohammed, J. Rawashdeh, and M. Abdullah, "Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results," in *2020 11th International Conference on Information and Communication Systems, ICICS 2020*, Institute of Electrical and Electronics Engineers Inc., Apr. 2020, pp. 243–248. doi: 10.1109/ICICS49469.2020.239556.

[17] M. Koziarski, "CSMOUTE: Combined Synthetic Oversampling and Undersampling Technique for Imbalanced Data Classification," Apr. 2020, [Online]. Available: http://arxiv.org/abs/2004.03409

[18] L. Qadrini, H. Hikmah, and M. Megasari, "Oversampling, Undersampling, Smote SVM dan Random Forest pada Klasifikasi Penerima Bidikmisi Sejawa Timur Tahun 2017," *Journal of Computer System and Informatics (JoSYC)*, vol. 3, no. 4, pp. 386–391, Sep. 2022, doi: 10.47065/josyc.v3i4.2154.

[19] A. Bansal and A. Jain, "Analysis of focussed under-sampling techniques with machine learning classifiers," in *2021 IEEE/ACIS 19th International Conference on Software Engineering Research, Management and Applications, SERA 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 91–96. doi: 10.1109/SERA51205.2021.9509270.

[20] Y. A. Sir and A. H. H. Soepranoto, "Pendekatan Resampling Data Untuk Menangani Masalah Ketidakseimbangan Kelas," *Jurnal Komputer dan Informatika*, vol. 10, no. 1, pp. 31–38, Mar. 2022, doi: 10.35508/jicon.v10i1.6554.

[21] Z. D. E. Maisat and A. Fauzan Dianta, "Implementasi Optimasi Hyperparameter GridSearchCV Pada Sistem Prediksi Serangan Jantung Menggunakan SVM," *Teknologi: Jurnal Ilmiah Sistem Informasi*, vol. 13, no. 1, pp. 8–15, 2023, doi: 10.26594/teknologi.v13i1.3098.

[22] J. Guo, K. Wang, and S. Jin, "Mapping of Soil pH Based on SVM-RFE Feature Selection Algorithm," *Agronomy*, vol. 12, no. 11, Nov. 2022, doi: 10.3390/agronomy12112742.

[23] F. S. Gomiasti, W. Warto, E. Kartikadarma, J. Gondohanindijo, and D. R. I. M. Setiadi, "Enhancing Lung Cancer Classification Effectiveness Through Hyperparameter-Tuned Support Vector Machine," *Journal of Computing Theories and Applications*, vol. 1, no. 4, pp. 396–406, Mar. 2024, doi: 10.62411/jcta.10106.

[24] R. Hasan, "Comparative Analysis of Machine Learning Algorithms for Heart Disease Prediction," *ITM Web of Conferences*, vol. 40, p. 03007, 2021, doi: 10.1051/itmconf/20214003007.

[25] M. Ozcan and S. Peker, "A classification and regression tree algorithm for heart disease modeling and prediction," *Healthcare Analytics*, vol. 3, Nov. 2023, doi: 10.1016/j.health.2022.100130.

[26] M. I. Aziz, A. Z. Fanani, and A. Affandy, "Analisis Metode Ensemble Pada Klasifikasi Penyakit Jantung Berbasis Decision Tree," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 7, no. 1, p. 1, Jan. 2023, doi: 10.30865/mib.v7i1.5169.

[27] Y. A. Ali, E. M. Awwad, M. Al-Razgan, and A. Maarouf, "Hyperparameter Search for Machine Learning Algorithms for Optimizing the Computational Complexity," *Processes*, vol. 11, no. 2, Feb. 2023, doi: 10.3390/pr11020349.

[28] E. Elgeldawi, A. Sayed, A. R. Galal, and A. M. Zaki, "Hyperparameter tuning for machine learning algorithms used for arabic sentiment analysis," *Informatics*, vol. 8, no. 4, Dec. 2021, doi: 10.3390/informatics8040079.

[29] D. Ismafillah, T. Rohana, and Y. Cahyana, "Implementasi Model Support Vector Machine dan Logistic Regression Untuk Memprediksi Penyakit Stroke," *Jurnal Riset Komputer)*, vol. 10, no. 1, pp. 2407–389, 2023, doi: 10.30865/jurikom.v10i1.5478.