Indonesian Journal of Computer Science
ISSN 2302-4364 (*print*) dan 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam, No. 1, Padang, Indonesia, Telp. (0751) 7056199, 7058325
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

Klasifikasi Penyakit Jantung Menggunakan Metode Synthetic Minority Over-Sampling Technique Dan Random Forest Clasifier

Abd Mizwar A. Rahim[1], Inggrid Yanuar Risca Pratiwi[2], Muhammad Ainul Fikri[3]
abdulmizwar@amikom.ac.id, inggridyrp@amikom.ac.id, fikri.ma@amikom.ac.id
Universitas Amikom Yogyakarta

Informasi Artikel

Diterima : 29 Sep 2023
Direview : 6 Okt 2023
Disetujui : 30 Okt 2023

Kata Kunci

Klasifikasi, Penyakit Jantung, Synthetic Minority Over-sampling Technique (SMOTE), Random Forest Classifier

Abstrak

Penelitian ini bertujuan untuk meningkatkan akurasi dalam klasifikasi penyakit jantung dengan menggabungkan Synthetic Minority Over-sampling Technique (SMOTE) dan algoritma Random Forest Classifier, dengan menggunakan data pasien yang mengalami diagnosis penyakit jantung dan tidak mengalami penyakit jantung. Tahapan awal yang dilakukan yaitu mengatasi masalah ketidakseimbangan kelas dengan mengaplikasikan SMOTE untuk menghasilkan sampel sintetis dari kelas minoritas, lalu dilanjutkan pada proses normalisasi data menggunakan metode min-max normalisasi, setelah itu masuk pada proses klasifikasi menggunakan Random Forest Classifier untuk melatih model dalam melakukan klasifikasi. Hasilnya menunjukkan bahwa pendekatan ini mampu meningkatkan kemampuan model dalam mengidentifikasi kasus penyakit jantung. Evaluasi model menghasilkan akurasi yang lebih baik dibandingkan hasil yang didapatkan pada penelitian yang dilakukan sebelumnya yaitu mencapai akurasi 92%, hasil terbaik ini terjadi peningkatan 2% dari hasil akurasi yang dihasilkan penelitian sebelumnya yaitu 90%.

Keywords

Classification, Heart disease, Synthetic Minority Over-sampling Technique (SMOTE), Random Forest Classifier

Abstract

This research aims to increase accuracy in the classification of heart disease by combining the Synthetic Minority Over-sampling Technique (SMOTE) and the Random Forest Classifier algorithm, using data from patients who have a diagnosis of heart disease and who do not have heart disease. The initial stage carried out was to overcome the class imbalance problem by applying SMOTE to produce synthetic samples from the minority class, then continued with the data normalization process using the min-max normalization method, after entering the classification process using the Random Forest Classifier to train the model to carry out classification. The results show that this approach can improve the model's ability to identify cases of heart disease. The model evaluation produced better accuracy compared to the results obtained in previous research, namely reaching 92% accuracy. This best result was an increase of 2% from the accuracy results produced by previous research, namely 90%.

A. Pendahuluan

Penyakit jantung adalah kondisi ketika kinerja jantung tidak optimal, sehingga kemampuannya untuk memompa darah dan mengalirkan oksigen ke seluruh tubuh dapat terhambat[1]. Terdapat beberapa faktor-faktor yang dapat meningkatkan risiko terkena penyakit jantung, termasuk gaya hidup yang tidak sehat seperti konsumsi makanan tinggi karbohidrat atau lemak, kelebihan berat badan, kurangnya aktivitas fisik, serta kebiasaan merokok, selain itu, faktor riwayat keluarga juga memegang peranan penting dalam peningkatan risiko terkena penyakit jantung[2].

Jumlah kasus penyakit serius di indonesia yang terus meningkat dengan signifikan pada tahun 2022, BPJS Kesehatan harus menangani sekitar 23,3 juta kasus penyakit yang mengalami peningkatan sebesar 18,6% dibandingkan dengan tahun 2021, biaya pengobatan untuk penyakit yang ditanggung oleh BPJS Kesehatan juga mencapai hampir Rp24,1 triliun pada tahun 2022, naik sebesar 34,3% dibandingkan dengan tahun sebelumnya, data sepanjang tahun 2022 menunjukkan bahwa penyakit jantung menjadi kasus paling umum, dengan 15,5 juta kasus, diikuti oleh kanker (3,2 juta kasus), stroke (2,5 juta kasus), dan gagal ginjal (1,3 juta kasus)[3].

Beberapa penelitian yang telah dilakukan terkait penyakit jantung menyoroti urgensi penanganan serius terhadap kondisi ini. Penelitian-penelitian ini menegaskan bahwa penyakit jantung harus diperlakukan dengan serius, karena serangan jantung yang parah atau terlambat ditangani dapat mengakibatkan berbagai komplikasi yang sangat berbahaya, komplikasi tersebut meliputi gangguan irama jantung atau aritmia, gagal jantung, syok kardiogenik, dan bahkan henti jantung[4].

Ada beberapa metode yang dapat membantu petugas medis dalam mendeteksi indikasi penyakit jantung pada pasien dengan lebih efisien. Salah satunya adalah pemanfaatan teknologi Machine Learning. Penelitian telah menunjukkan bahwa penggunaan Machine Learning memiliki potensi besar dalam mengatasi topik klasifikasi dan meningkatkan optimasi dalam pengembangan sistem layanan kesehatan. Sebagai contoh, sebuah penelitian yang dilakukan oleh (Widiastiwi & Ernawati, 2021) mengungkapkan bahwa penerapan Machine Learning telah terbukti efektif dalam membantu dalam penilaian dini terhadap indikasi penyakit jantung, sehingga memungkinkan deteksi yang lebih cepat dan akurat untuk pasien yang memerlukan perhatian medis[5].

Terdapat beberapa penelitian sebelumnya yang mengkaji topik yang serupa dengan penelitian ini. Pertama dilakukan oleh (Mohan et al., 2019) mengenai prediksi penyakit jantung menggunakan metode machine learning, hasil dari penelitian ini menunjukkan bahwa metode Hybrid Random Forest with a Linear Model (HRFLM) mampu memberikan prediksi penyakit jantung dengan tingkat akurasi yang cukup tinggi, mencapai 88,4%, namun, terdapat beberapa kekurangan dalam penelitian ini yang perlu diperhatikan, salah satunya adalah penghapusan data yang hilang pada fitur tertentu, yang mengakibatkan sebagian besar kelas dataset menjadi hilang, hal ini dapat menyebabkan model hanya memiliki pengetahuan tentang data yang memiliki kelas mayoritas, tanpa memperhatikan apakah terdapat ketidakseimbangan dalam data atau tidak[6]. Selain itu, penelitian ini juga tidak memperhatikan perbedaan skala dalam data, jika terdapat perbedaan

skala yang signifikan antara fitur-fitur, maka model mungkin tidak akan optimal dalam melakukan klasifikasi. Berikutnya penelitian mengenai perbandingan metode machine learning saat memprediksi penyakit jantung yang dilakukan oleh [7] Hasil penelitian menunjukkan bahwa algoritma K-Nearest Neighbors (KNN) terbukti sangat efektif dalam memprediksi penyakit jantung, dengan tingkat akurasi mencapai 87%. Namun, seperti penelitian sebelumnya, penelitian ini juga memiliki beberapa kelemahan yang perlu diperhatikan. Pertama yaitu penghapusan data yang hilang pada fitur tertentu, sehingga yang dapat mengakibatkan sebagian besar kelas dataset menjadi hilang, dan mempertimbangkan perbedaan skala data pada dataset pengolahan[7]. Selanjutnya penelitian mengenai Klasifikasi Penyakit Jantung Pada Dataset Imbalanced Class menggunakan Algoritme Stacking yang dilakukan oleh (Nurmasani & Pristyanto, 2021). Hasil penelitian ini menunjukkan bahwa algoritma stacking memiliki kinerja yang lebih baik dari segi akurasi, TPR (True Positive Rate), TNR (True Negative Rate), GMean, dan AUC (Area Under the Curve) dibandingkan dengan menggunakan single classifier lainnya. Pengujian dalam penelitian ini mencapai tingkat akurasi tertinggi sebesar 90%. Penting untuk dicatat bahwa penelitian ini tidak memperhatikan perbedaan skala antara nilai-nilai fitur dalam dataset, sehingga diperlukan normalisasi data untuk mengatasi hal ini[8]. Yang terakhir penelitian yang dilakukan oleh (Shiva Shanta Mani & Manikandan, 2020), dalam penelitian ini, sejumlah metode klasifikasi telah diuji, dan hasilnya menunjukkan bahwa metode terbaik untuk melakukan klasifikasi adalah Support Vector Machine (SVM) dengan tingkat akurasi sebesar 72,6%[9].

Penelitian ini mengangkat topik penyakit jantung, data yang dipakai dalam pengolahan penelitian ini adalah heart disease dataset yang diambil dari kaggel, dataset tersebut mengalami imbalance class dimana jumlah kategori terindikasi penyakit (1) sebanyak 526 dan tidak terindikasi penyakit (0) sebanyak 499, imbalance kelas dapat mempengaruhi model saat klasifikasi, model hanya dapat menentukan kelas mayoritas dan kemungkinan kelas minoritas yang diprediksi, akan diprediksi sebagai kelas mayoritas[10]. Dengan terjadinya imbalance pada dataset, maka penelitian ini menerapkan metode Synthetic Minority Over-Sampling Technique untuk dapat mengatasi dataset yang mempunyai masalah imbalance class. Selain itu penelitian ini, saat melakukan klasifikasi penyakit jantung metode yang digunakan yaitu algoritma random forest. algoritma tersebut dapat mengurangi masalah overfitting yang sering terjadi pada model yang kompleks, ini dilakukan dengan menggabungkan prediksi dari banyak pohon keputusan yang terkondisi secara independen, yang mencegah model terlalu "menghafal" data pelatihan[11].

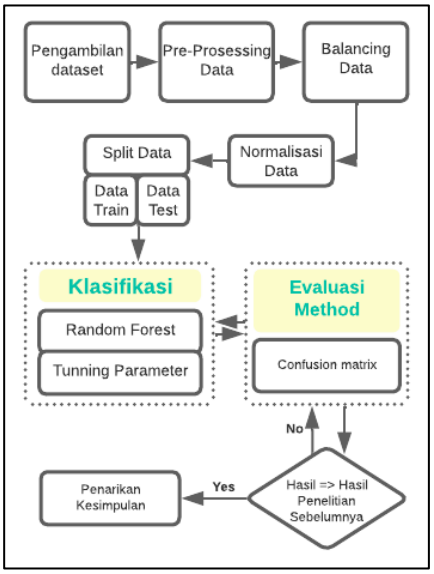
Penelitian sebelumnya dengan study kasus yang sama memiliki hasil akurasi yang paling baik yaitu 90%. Secara umum penelitian-penelitian tersebut tidak mengatasi data yang tidak seimbang, skala data yang berbeda, dan pengubahan missing value pada fitur dataset, maka dengan itu tujuan dari penelitian ini ialah meningkatkatkan hasil akurasi pada klasifikasi penyakit jantung dengan menutupi kekurangan yang dilakukan oleh penelitian sebelumnya. Penelitian ini mengusulkan metode SMOTE dan machine learning dengan algoritma random forest untuk mendapatkan hasil akurasi yang lebih baik.

Penelitian ini diharapkan dapat membantu dalam mengidentifikasi lebih dini indikasi penyakit jantung pada pasien, sehingga penanganan medis dapat dimulai

lebih cepat dan efisien. Selain itu diharapkan metode yang digunakan dapat memberikan hasil diagnosa yang lebih akurat, sehingga mengurangi kesalahan diagnosa yang dapat berdampak negatif pada pasien dan dengan penerapan metode SMOTE, penelitian ini diharapkan dapat mengatasi masalah ketidakseimbangan data yang sering ditemukan dalam dataset penyakit jantung, sehingga model dapat belajar dengan lebih baik dari kedua kelas data pada dataset heart disease.

B. Metode Penelitian

Penelitian ini melibatkan beberapa tahapan, seperti yang tergambar dalam Gambar 1. Tahapan dimulai dengan pengambilan dataset yang sesuai dengan topik penelitian ini yaitu penyakit jantung, lalu dilakukan pre-prosessing data untuk memeriksa dan memperbaiki kesalahan yang ditemukan pada data yang di gunakan dalam penelitian ini, berikutnya dilakukan balancing data dikarenakan data yang digunakan pada penelitian ini mengalami ketidakseimbangan kelas, maka pada tahap ini diterapkannya metode SMOTE untuk mengatasi ketidakseimbangan kelas. Selanjutnya masuk pada proses normalisasi data menggunakan metode min-max normalisasi, proses ini dilakukan untuk mengubah atribut numerik dalam dataset sehingga memiliki skala yang seragam atau sebanding, lalu masuk pada proses split data, pada bagian ini dilakukan untuk membagi dataset menjadi dua bagian yakni bagian yang digunakan untuk training data, dan untuk testing data. berikutnya masuk pada proses klasifikasi dan pengaturan parameter metode random forest. Setelah itu dilakukannya evaluasi terhadap proses klasifikasi yang dilakukan sebelumnya, jika hasil akurasi dari klasifikasi melebihi hasil akurasi yang dihasilkan pada penelitian sebelumnya maka dilanjutkan pada proses berikutnya, jika tidak maka kembali pada proses sebelumnya yaitu proses klasifikasi dan mengatur parameter hingga mendapatkan hasil yang optimal. Dan yang terakhir yaitu penarikan kesimpulan, hasil pengujian dan evaluasi dijadikan kesimpulan akhir mengenai metode balancing data, normalisasi data, proses klasifikasi, dan parameter yang diujikan.



Gambar 1. Tahap Penelitian

1. Pengambilan Dataset.
   Dataset yang digunakan dalam penelitian ini diperoleh dari platform kaggle yang dikenal sebagai Heart Disease Dataset. Dataset ini merupakan kumpulan data yang tersedia secara public milik David lapp yang dapat diakses melalui tautan Kaggle berikut ini https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset?resource=download dataset ini digunakan untuk melakukan prediksi mengenai kemungkinan seseorang menderita penyakit jantung berdasarkan sejumlah parameter input[12]. Parameter-parameter ini meliputi informasi seperti usia, jenis kelamin, jenis nyeri dada yang dirasakan, tekanan darah, tingkat kolesterol, kadar gula darah, hasil elektrokardiografi, detak jantung maksimum, adanya angina yang diinduksi oleh latihan, nilai depresi segmen ST yang diinduksi oleh latihan relatif terhadap kondisi istirahat (oldpeak), kemiringan segmen ST pada puncak latihan, dan jumlah kapal utama yang diwarnai oleh fluoroskopi (dalam skala 0-3). Setiap baris dalam dataset ini berisi informasi yang relevan mengenai seorang pasien yang dapat digunakan untuk analisis dan prediksi terkait dengan penyakit jantung. kategori pasien pada variabel dependent berjumlah 2 kategori yaitu pasien terindikasi penyakit jantung dan tidak terindikasi penyakit. Berikut ini tabel 1 yang menggambarkan informasi fitur pada dataset ini.

Tabel 1. Dataset

No	Fitur	Keterangan
1	Age	Menyimpan usia pasien
2	Sex	(1 = laki-laki; 0 = perempuan)
3	Chest Pain Type	jenis nyeri dada
4	Resting Blood Pressure	tekanan darah
5	serum cholestoral in mg/dl	Kolesterol serum dalam mg/dl
6	fasting blood sugar	(gula darah puasa > 120 mg/dl) (1 = benar; 0 = salah)
7	resting electrocardiographic results	hasil elektrokardiografi istirahat
8	maximum heart rate achieved	detak jantung maksimum tercapai
9	exercise induced angina	latihan diinduksi angina (1 = ya; 0 = tidak ada)
10	oldpeak	Depresi ST
11	slope	segmen ST latihan puncak
12	number of major vessels (0-3) colored by flourosopy	jumlah kapal utama (0-3) diwarnai oleh flourosopy
Class	Target	bilangan bulat bernilai 0 = tidak ada penyakit dan 1 = penyakit.

2. Pre-prosessing data.
   Preprocessing adalah tahap yang diperlukan untuk mengevaluasi data dalam dataset yang telah dipilih, mengidentifikasi dan memperbaiki kesalahan yang mungkin ada dalam dataset tersebut, sehingga memungkinkan kelangsungan proses berikutnya[13]. Teknik yang digunakan dalam penelitian yaitu mengatasi missing value dan normalisasi data. Mengatasi missing value pada atribut dataset dilakukan karena akan terdapat informasi yang hilang, bias dalam analisis, dan dapat menurunkan hasil akurasi saat proses klasifikasi[14]. Masalah dalam analisis dan klasifikasi dapat timbul ketika atribut dalam dataset memiliki skala yang bervariasi, normalisasi digunakan untuk mengatasi perbedaan skala ini dengan menyesuaikan semua nilai variabel dalam rentang yang serupa, hal ini bertujuan untuk memastikan bahwa tidak ada variabel yang mendominasi proses klasifikasi hanya karena memiliki skala yang lebih besar, sehingga membantu mencegah bias dalam model[15].
3. Normalisasi data.
   Proses untuk mengubah atribut numerik dalam dataset agar memiliki skala yang seragam atau sebanding disebut normalisasi. Salah satu metode normalisasi yang umum digunakan adalah metode min-max normalization. Proses ini dapat dijelaskan menggunakan rumus berikut :

$$N = \frac{MinRange+(X-MinValue)(MaxRange-MinRange)}{MaxValue-MinValue} \quad (1)$$

   N =Normalisasi Min_Max
   MinRange =Nilai Konversi Kecil Yang ditentukan.
   MaxRange =Nilai Konversi Terbesar yang ditentukan.
   MaxValue =Nilai Terbesar pada atribut yang dibandingkan.
   MinValue =Nilai Terkecil pada atribut yang dibandingkan.
4. Balancing data Smote.
   SMOTE adalah teknik oversampling yang digunakan untuk mengatasi ketidakseimbangan kelas dalam dataset. Pada SMOTE, data pada kelas minoritas diperbanyak dengan menciptakan data sintetis yang didasarkan pada replikasi data dari kelas minoritas. Prosedur oversampling dalam SMOTE melibatkan pemilihan instance dari kelas minoritas dan mencari k-nearest neighbor untuk setiap instance tersebut. Kemudian, instance sintetis dihasilkan dengan cara menggabungkan atau interpolasi antara instance-instance yang telah dipilih, bukan hanya dengan menggandakan instance kelas minoritas yang ada. Dengan pendekatan ini, SMOTE dapat membantu mengatasi masalah overfitting yang berlebihan yang mungkin terjadi ketika melakukan oversampling dengan hanya mereplikasi instance kelas minoritas[16]. Berikut ini rumus dari smote :

$$X_{syn} = Xi + (X_{knn}-x_i) \times \delta \quad (2)$$

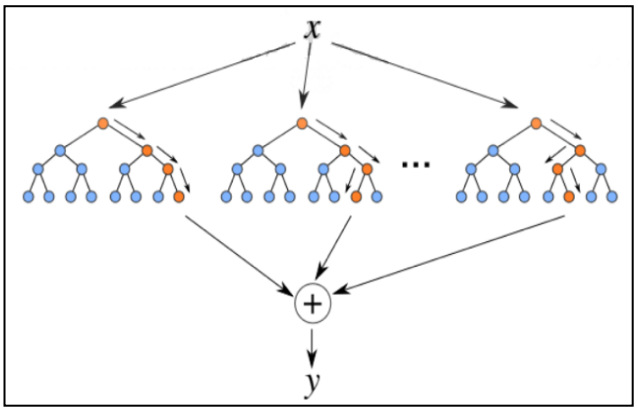
   Keterangan :
   X_{syn} = data synthesis yang akan diciptakan
   X_i = data yang akan di replikasi
   X_{knn} = data yang memiliki jarak dari data Xi

δ = angka acak antara 0 sampai 1

5. Model Random Forest.

Tahun 2001, Breiman memperkenalkan algoritma Random Forest, yang memiliki kemampuan untuk menangani dua jenis permasalahan, yaitu klasifikasi dan regresi[17]. Random Forest adalah hasil pengembangan dari metode Classification dan Regression Tree (CART) yang mengadopsi teknik bagging (bootstrap aggregation) dan pemilihan fitur acak, bagging adalah salah satu teknik yang digunakan untuk meningkatkan kinerja algoritma klasifikasi, konsep bagging ini terkait dengan pendekatan ensemble dalam statistik[18]. Penerapan metode Random Forest melibatkan serangkaian langkah yang meliputi [19] :

a. Langkah pertama dalam penerapan metode Random Forest adalah membuat sampel data dengan mengambil sejumlah data acak dari dataset, dan proses ini melibatkan pengambilan data dengan pengembalian.
b. Selanjutnya, sampel data tersebut digunakan untuk konstruksi pohon ke-i, di mana nilai i berkisar dari 1 hingga k sebagai iterasi.
c. Langkah 1 dan 2 diterapkan berulang-ulang sebanyak k kali, sesuai dengan jumlah pohon yang ingin dibangun dalam ensemble hutan acak.



Gambar 2. Random Forest

Ketika membangun pohon keputusan dengan metode CART, proses komputasi melibatkan evaluasi informasi yang menjelaskan tingkat kepentingan atribut dalam mengklasifikasikan setiap simpul dalam pohon. Dengan lebih spesifik, jika kita mempertimbangkan simpul N yang digunakan untuk memisahkan data kelas D berdasarkan atribut-atributnya, perhitungan ini membantu mengukur sejauh mana atribut-atribut tersebut memiliki relevansi atau memberikan informasi dalam proses pemisahan kelas data. Pembagian simpul dalam pohon keputusan dilakukan dengan memilih atribut yang memiliki tingkat informasi validasi tertinggi. Perhitungan tingkat informasi validasi ini dapat dijelaskan melalui rumus berikut :

$$Gain\ (A) = Info\ (D) - Info(D) \quad (3)$$

Untuk mendapatkan nilai info(D), kita dapat menghitungnya dengan menggunakan rumus 2 dan 3, yang akan menghasilkan nilai info A(D).

$$Info(D) = -\sum_{i=1}^{n} p_i log2(p_i) \quad (4)$$

Keterangan :

N = jumlah kelas target

Pi = proporsi kelas i terhadap partisi D

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j) \quad (5)$$

Keterangan :

V = jumlah partisi.

Dj = total partisi ke j.

D = total baris pada semua partisi.

Untuk atribut yang mengandung nilai kontinu atau numerik, diperlukan penentuan titik pemisahan optimal untuk mengelompokkan nilai tersebut. Proses pencarian nilai pemisahan terbaik dimulai dengan mengurutkan data. Median atau rata-rata dari setiap pasangan nilai yang berdekatan dapat digunakan sebagai titik pemisahan yang potensial. Sebagai contoh, jika kita memiliki atribut A yang berisi nilai kontinu, maka langkahnya adalah mengurutkan semua nilai A, dan nilai tengahnya dapat menjadi salah satu dari titik pemisahan. Hasilnya bisa menghasilkan dua atau lebih partisi, seperti dalam contoh ini di mana v = 2 (dengan j=1 dan 2) adalah jumlah partisi yang mungkin[20].

d. Penyetelan Hyperparameter.

Penyetelan parameter dalam algoritma Random Forest adalah langkah penting untuk menyesuaikan dan mengoptimalkan nilai-nilai parameter yang dapat memiliki dampak pada performa keseluruhan model Random Forest[21]. Parameter-parameter ini memberikan kontrol terhadap cara pembangunan pohon-pohon keputusan dalam ensemble dan pengaruhnya terhadap hasil akhir model. beberapa parameter umum yang dapat disesuaikan dalam proses penyetelan pada algoritma Random Forest[22].

Tabel 2. Parameter Random Forest.

No	Parameter	Keterangan
1	*n_estimators*	jumlah pohon keputusan yang akan dibangun
2	*max_depth*	mengatur kedalaman maksimum setiap pohon
3	*max_features*	mengontrol jumlah fitur yang akan dipertimbangkan
4	*min_samples_leaf*	jumlah minimum sampel yang diperlukan
5	*min_samples_split*	jumlah minimum sampel yang diperlukan untuk mempertimbangkan pemisahan simpul
6	*bootstrap*	pengambilan sampel acak dengan pengembalian dari dataset pelatihan

6. Evaluasi Method.

Setelah proses analisis selesai dan kita memiliki hasil prediksi, langkah berikutnya adalah mengevaluasi prediksi. Secara umum, evaluasi kinerja dalam metode klasifikasi dilakukan dengan membandingkan hasil prediksi dengan nilai aktual pada data uji sebagai data sebenarnya. Evaluasi kinerja model ini menggunakan confusion matrix. Confusion matrix adalah tabel yang memberikan rincian kesalahan terkait dengan hasil klasifikasi. Ini adalah alat yang digunakan untuk mengevaluasi kinerja model klasifikasi dengan memperhitungkan jumlah kasus yang diklasifikasikan dengan benar dan yang salah berdasarkan hasil prediksi[23].

Tabel 3. Confusion Matrix

Classification	*Predicted Positive*	*Predicted Negative*
Actual Positive	*True Positive (TP)*	*False Negative (FN)*
Actual Negative	*False Positive (FP)*	*True Negative (TN)*

Dalam pengukuran kinerja menggunakan confusion matrix, terdapat empat komponen atau bagian yang digunakan untuk mengidentifikasi hasil prediksi, dan di antaranya adalah sebagai berikut[24].

a. TP (True Positive) menghitung jumlah data yang memiliki nilai aktual positif dan juga diprediksi sebagai positif.
b. TN (True Negative) menghitung jumlah data yang memiliki nilai aktual negatif dan juga diprediksi sebagai negatif.
c. FP (False Positive) menghitung jumlah data yang memiliki nilai aktual negatif tetapi salah diprediksi sebagai positif.
d. FN (False Negative) menghitung jumlah data yang memiliki nilai aktual positif tetapi salah diprediksi sebagai negatif.

Terdapat nilai evaluasi yang sering di pakai pada klasifikasi biner, nilai tersebut adalah akurasi, yang dapat dilihat berdasarkan nilai confusion matrix[25]. Accuracy (ACC) adalah efektivitas dari hasil yang didapatkan dalam proses klasifikasi rumus dari akurasi dapat dilihat pada persamaan ke 6 :

$$Accuracy\ (\%) = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (6)$$

C. Hasil dan Pembahasan

1. Dataset.

Penelitian ini, menggunakan dataset mengenai penyakit jantung yang diperoleh dari platform Kaggle milik David Lapp, kumpulan data ini berasal dari tahun 1988 dan terdiri dari empat database yaitu Cleveland, Hongaria, Swiss, dan Long Beach V. Kumpulan data ini berisi 76 atribut, termasuk atribut yang diprediksi, tetapi semua eksperimen yang dipublikasikan mengacu pada penggunaan subset dari 14 atribut tersebut. atribut "target" mengacu pada adanya penyakit jantung pada pasien. Bernilai bilangan bulat 0 = tidak ada penyakit dan 1 = sakit. berikut ini merupakan tampilan dataset yang dapat dilihat pada tabel 4.

Tabel 4. Dataset

age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
52	1	0	125	212	0	1	168	0	1	2	2	3	0
53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
70	1	0	145	174	0	1	125	1	2.6	0	0	3	0
61	1	0	148	203	0	1	161	0	0	2	1	3	0
62	0	0	138	294	1	1	106	0	1.9	1	3	2	0
58	0	0	100	248	0	0	122	0	1	1	0	2	1
58	1	0	114	318	0	2	140	0	4.4	0	3	1	0

2. Pre-prosessing data.

Terdapat dua Teknik yang diimplementasikan pada proses ini, pertama mengatasi missing value dikarenakan atribut thalach terdapat 36 data yang kosong, bisa dilihat pada gambar 3 berikut ini.

```
data.isna().sum()

age          0
sex          0
cp           0
trestbps     0
chol         0
fbs          0
restecg      0
thalach     36
exang        0
oldpeak      0
slope        0
ca           0
thal         0
target       0
dtype: int64
```

Gambar 3. Missing value

Atribut yang memiliki nilai yang hilang akan diisi dengan rata-rata dari atribut thalach untuk pasien yang mengalami penyakit jantung dan pasien yang tidak mengalami penyakit jantung. Pendekatan ini dipilih daripada menghapus baris data yang memiliki nilai kosong, karena dengan cara ini tidak ada data yang terbuang, dapat mengurangi bias, dan perbaikan akurasi model[26]. Hasil dari proses ini dapat ditemukan pada Gambar 4.

```
mean_thalach_has_jantung = data[data['target']==1]['thalach'].mean()
mean_thalach_has_jantung
```

```
158.61023622047244
```

```
mean_thalach_has_No_jantung = data[data['target']==0]['thalach'].mean()
mean_thalach_has_No_jantung
```

```
138.985446985447
```

```
data.loc[data['target']==1,'thalach'] = data.loc[data['target']==1,
'thalach'].fillna(mean_thalach_has_jantung)
data.loc[data['target']==0,'thalach'] = data.loc[data['target']==0,
'thalach'].fillna(mean_thalach_has_No_jantung)
```

Gambar 4. Replace missing value

Setelah proses penggantian nilai yang hilang dilakukan, langkah selanjutnya adalah melakukan pengecekan untuk memastikan apakah nilai yang hilang telah berhasil diganti atau tidak. Hasil dari pengecekan ini dapat ditemukan pada Gambar 5.

```
1  data.isna().sum()

age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

Gambar 5. Hasil Pengubahan Missing Value dengan rata-rata nilai atribut Thalach terindikasi penyakit dan Thalach yang tidak terindikasi penyakit

Teknik kedua pada proses pre-prosessing ini adalah melakukan normalisasi data, normalisasi data dilakukan untuk menghindari atribut dengan skala besar mendominasi atribut dengan skala kecil[27]. Skala atribut yang besar dapat menyebabkan masalah numerik dalam perhitungan model, seperti overflow atau underflow, normalisasi mengatasi masalah ini dengan memperkecil rentang nilai atribut[28]. Teknik ini dapat membuat model lebih stabil dan kurang sensitif terhadap perubahan dalam skala data. Ini dapat membantu mencegah overfitting[29]. Berikut ini hasil standarisasi menggunakan teknik normalisasi min_max pada dataset penyakit jantung yang dapat dilihat pada gambar 6.

```
1  dataNormalisasi = min_max_scaler.fit_transform(X) #transformasi MinMax untuk fitur

1  dataNormalisasi

array([[0.47916667, 1.        , 0.        , ..., 1.        , 0.5       ,
        1.        ],
       [0.5       , 1.        , 0.        , ..., 0.        , 0.        ,
        1.        ],
       [0.85416667, 1.        , 0.        , ..., 0.        , 0.        ,
        1.        ],
       ...,
       [0.375     , 1.        , 0.        , ..., 0.5       , 0.25      ,
        0.66666667],
       [0.4375    , 0.        , 0.        , ..., 1.        , 0.        ,
        0.66666667],
       [0.52083333, 1.        , 0.        , ..., 0.5       , 0.25      ,
        1.        ]])
```

Gambar 6. Normalisasi data.

3. Balancing data.

Label pada dataset penyakit jantung ini terjadi imbalance kelas atau terdapat kelas mayoritas dan kelas minoritas, kondisi dari dataset ini menyebabkan hasil klasifikasi tidak optimal. dengan mengatasi imbalance yang terjadi penelitian ini menggunakan metode Synthetic Minority Over-Sampling Technique (SMOTE) dengan cara menambah kelas minoritas agar sama dengan kelas mayoritas dengan

menambahkan data buatan, data buatan atau sintesis tersebut di buat berdasarkan k-tetangga terdekat. berikut hasil balancing data dapat dilihat pada gambar 7.

1027	41.794001	1	0	115.98	172.98	0	0	117.588	1	2.2990001	1	0	3	0
1028	52	1	0	128	255	0	1	161	1	0	2	1	3	0
1029	46	1	2	150	231	0	1	147	0	3.6	1	0	2	0
1030	60	1	0	125	258	0	0	141	1	2.8	1	1	3	0
1031	48.290754	1	0.0726885	130	256	1	0	149.70925	1	0.0218065	1.9636558	1.9636558	2.9273115	0
1032	46.506552	1	1.0131045	113.92137	245.96069	0	0.5065523	148.05242	0	0.3947582	2	0	2.4934477	0
1033	57	1	1	154	232	0	0	164	0	0	2	1	2	0
1034	47	1	2	108	243	0	1	152	0	0	2	0	2	0
1035	59	1	3	170	288	0	0	159	0	0.2	1	0	3	0
1036	49.913919	0.1810135	0	132.17216	305.72405	0	0.8189865	142.90507	1	0.9827838	1	0.5430404	3	0
1037	55.476105	1	0	110	239	0	0.704779	130.72354	1	2.3276463	1	1	3	0
1038	57	1	0	130	131	0	1	115	1	1.2	1	1	3	0
1039	50.920771	0.8158458	0	141.05353	200.92077	0	0.3683085	126.73662	1	1.1025697	1	0.1841542	3	0
1040	69	1	2	140	254	0	0	146	0	2	1	3	3	0
1041	59	1	3	134	204	0	1	162	0	0.8	2	2	2	0
1042	62.557323	1	0	142.59554	209.59554	0	0	134.88535	0.4808924	1.9519108	1.4808924	1.5191076	1.9617848	0
1043	44	1	0	120	169	0	1	144	1	2.8	0	0	1	0
1044	57.837758	1	0	150.32448	270.64897	0	0.1622415	107.26845	1	0.8648966	1.8377585	0.1622415	3	0
1045	57.54899	1	2.2744948	135.45101	203.75816	0.2418351	0.7581649	160.30715	0.2418351	1.3562206	1.5163299	1.5163299	2.2418351	0
1046	58	0	0	170	225	1	0	146	1	2.8	1	2	1	0
1047	63.044064	1	0	139.58983	167.81695	0	0.2271182	125	1	3.3728818	0.7728818	0.7728818	2.2271182	0
1048	47.694426	1	0	116.92705	203.92705	0	0.4618032	172.84377	0	0.4618032	2	1.4618032	2.4618032	0
1049	43	0	0	132	341	1	0	136	1	3	1	0	3	0
1050	64.229879	1	0	120	245.31036	0	0.0766263	94.084342	0.9233737	2.1080484	0.0766263	0.9233737	2	0
1051	57	1	2	128	229	0	0	150	0	0.4	1	1	3	0
1052	47.135174	1	1.1729652	110.86483	229.17297	0	1	169.21076	0.0864826	1.2421512	0.0864826	0	3	0
1053	54.576809	1	0.6576809	121.36928	280.57681	0	0	162.05391	0	1.3549852	1	0	2.3423191	0

Gambar 7. Hasil Balancing data dengan SMOTE.

Pada gambar 7 merupakan hasil dari proses balancing kelas pada dataset menggunakan smote, hasil dari smote ini menambahkan kelas minoritas pada kelas 0 (tidak ada penyakit) agar seimbang dengan kelas mayoritas yaitu kelas 1 (penyakit). jumlah data yang ditambahkan berjumlah 27 data.

4. **Split data.**

   Pembagian yang dilakukan penelitian ini yaitu data training dan data testing menjadi 80/20. Dengan jumlah pembagian tersebut mempunyai tujuan melihat model dalam memprediksi ketika mempunyai data test dengan jumlah 211, secara umum model machine learning mendapatkan hasil akurasi yang baik jika memiliki jumlah data testing yang sedikit dan data training yang banyak. Maka dalam penelitian ini meningkatkan data test dan menguji apakah model mendapatkan hasil yang baik atau tidak. Pada Tabel 5 menggambarkan pembagian data yang di lakukan.

Tabel 5. Train/Test Split

Keterangan	Data Training	Data Testing	Total
Proporsi	80%	20%	100%
Jumlah	841	211	1052

   Pada Tabel 4 di jelaskan bahwa pembagian data training dan data testing yang dilakukan menjadi 80/20, 80% untuk data training yang berjumlah 841 data dan 20% untuk data testing yang berjumlah 211 data, dengan itu jumlah keseluruhan data dari dataset berjumlah 1052 total data.

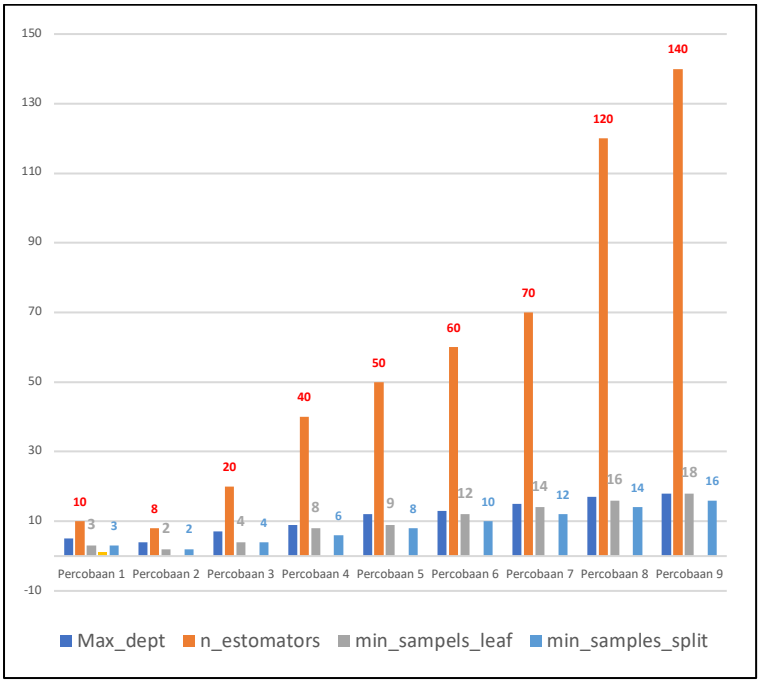
5. **Klasifikasi & Tunning Parameter.**

   Proses klasifikasi penelitian ini dimulai dengan langkah utama, yaitu penyesuaian parameter (tuning parameter), dan selanjutnya, dilakukan klasifikasi berdasarkan parameter-parameter yang telah diatur. Hasil dari tuning parameter dapat dilihat pada tabel 6 di bawah ini.

Tabel 6. Hasil Tuning Parameter Metode Random Forest.

No	max _dept	n_ estimators	min_ samples_ leaf	min_ samples_ split	acc
1	5	10	3	3	86%
2	4	8	2	2	83%
3	7	20	4	4	88%
4	9	40	8	6	85%
5	12	50	9	8	87%
6	13	60	12	10	89%
7	15	70	14	12	90%
8	17	120	16	14	92%
9	18	140	18	16	91%

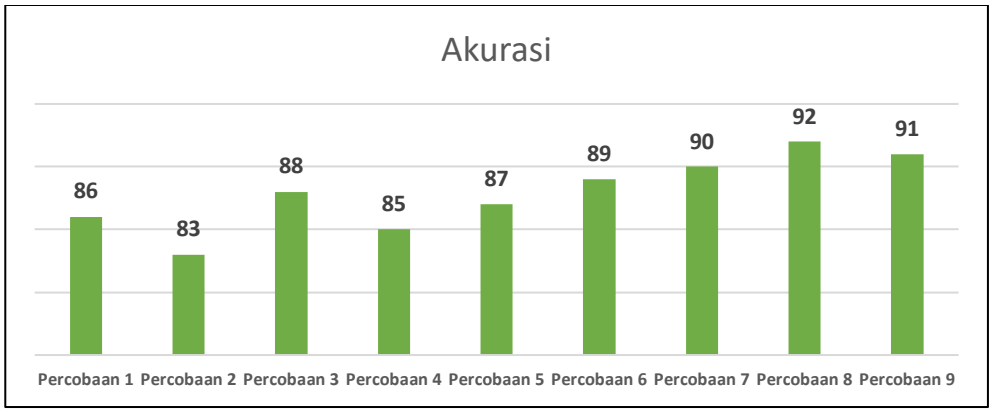
Tabel 6, dijelaskan bahwa parameter -parameter yang diuji pada proses klasifikasi meliputi max-depth, n-estimator, random-state, min-samples-leaf, dan min-samples-split. Pada percobaan yang dilakukan, parameter-parameter tersebut diatur dengan nilai yang rendah dalam iterasi awal, dan pada iterasi berikutnya, parameter tersebut diatur dengan nilai yang meningkat. Nilai-nilai parameter yang diatur dapat dilihat pada gambar 8.



Gambar 8. Value Tunning Parameter.

Gambar 8. Menunjukan bahwa nilai parameter yang terjadi peningkatan adalah parameter max-depth, n-estimator, min-samples-leaf, dan min-samples-split, keempat parameter tersebut juga terdapat sekali penurunan nilai yaitu pada percobaan ke 2 . Percobaan yang dilakukan oleh penelitian ini berjumlah 9 percobaan, dan hasil terbaik-nya terdapat pada percobaan ke-8 dengan hasil akurasi 92%, yang mempunyai kedalaman maksimum pohon dibatasi hingga 17 level, ada 120 pohon ensemble, 16 sampel dalam setiap daun, dan 14 sampel yang memenuhi syarat untuk pemisahan simpul.

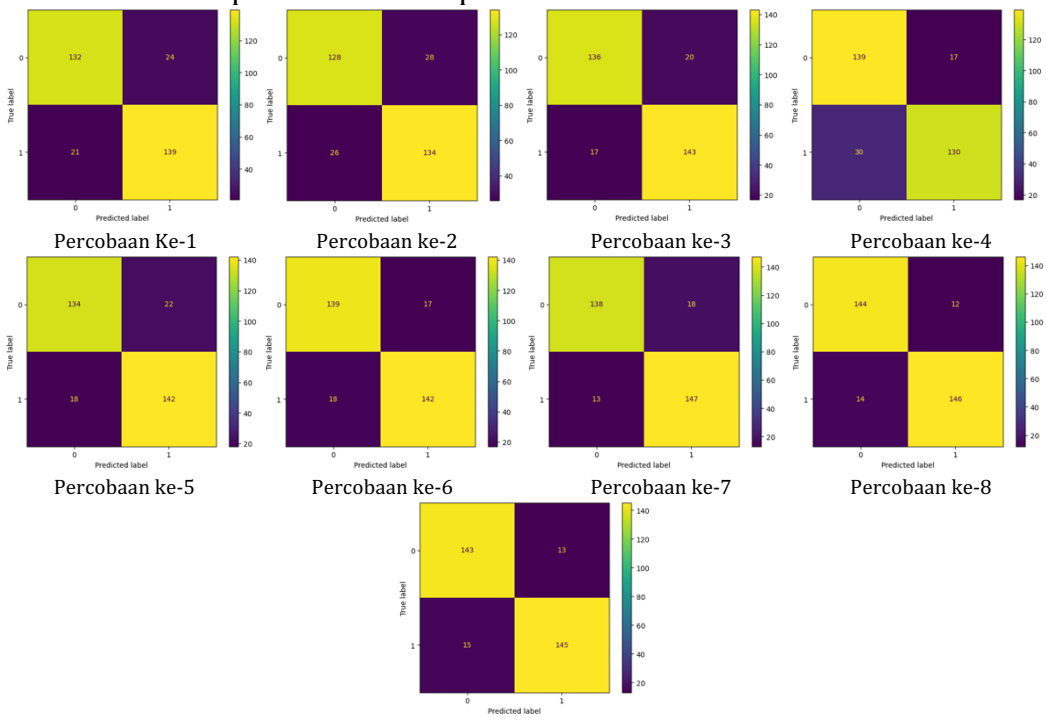
hasil akurasi dari keseluruhan percobaan menggunakan split data 80/20. Dapat dilihat pada gambar 9.



Gambar 9. Hasil akurasi split data 80/20

6. Evaluasi

Dalam penelitian ini, dilakukan 9 percobaan menggunakan metode klasifikasi Random Forest, dan hasil akurasi tertinggi ditemukan pada percobaan ke-delapan dengan pembagian data 80/20. Confusion matrix dari keseluruhan percobaan ini dapat ditemukan dalam Gambar 10.



Gambar 10. Hasil confusion matrix

Gambar 10 menunjukan hasil evaluasi keseluruhan percobaan yang dilakukan menggunakan confusion matrix. Nilai evaluasi terbaik terdapat pada percobaan ke-8, hasil tersebut menyampaikan bahwa sebanyak dua belas data yang salah diprediksi oleh model yang seharusnya class 0 (tidak

terindikasi penyakit) tetapi di prediksi sebagai class 1 (terindikasi penyakit), kemudian terdapat empat belas data yang seharusnya class 1 (terindikasi penyakit) tetapi diprediksi sebagai class 0 (tidak terindikasi penyakit). Eksperimen ke delapan mendapatkan hasil terbaik dengan nilai akurasi sebesar 92% yang dapat dilihat pada Gambar 9. Akurasi ini merupakan rasio prediksi yang tepat dalam mengidentifikasi seseorang terindikasi penyakit dan tidak terindikasi penyakit secara keseluruhan dalam dataset. Jumlah data yang diprediksi salah pada pengujian ke delapan (pengujian terbaik) adalah 26 data, dan total data yang benar di klasifikasikan adalah 290 data dari total seluruh data uji yang berjumlah 316 data. Terjadi penurunan akurasi 2% saat nilai parameter max-depth, n-estimator, min-samples-leaf, dan min-samples-split di kurangi, terjadi juga peningkatan akurasi 2% saat keempat nilai parameter yaitu max-depth, n-estimator, min-samples-leaf, dan min-samples-split ditingkatkan. Keempat nilai parameter tersebut jika melebihi value yang diatur pada pengujian ke 8 maka akurasi terjadi penurunan sebanyak 1% yaitu menjadi 91% akurasi.

D. Simpulan

Berdasarkan analisis yang dilakukan menggunakan dataset penyakit jantung yang mempunyai dua kelas (binary classification), dapat diambil kesimpulan sebagai berikut :

- Penelitian ini terdapat 9 kali percobaan, hasil terbaiknya terdapat pada percobaan ke 8 dengan akurasi 92%, hasil terbaik ini terjadi peningkatan 2% dari hasil akurasi yang dihasilkan penelitian sebelumnya yaitu 90%.
- Beberapa teknik penelitian yang berkontribusi dalam mencapai kondisi model yang optimal, sehingga menghasilkan akurasi yang tinggi dalam proses klasifikasi, mencakup proses pre-processing data, normalisasi data, balancing dataset, klasifikasi, dan pengaturan parameter model. Dengan menggabungkan teknik-teknik ini secara tepat, penelitian ini dapat menghasilkan model klasifikasi yang optimal dengan akurasi yang tinggi dalam mengidentifikasi pola dan prediksi dalam dataset penyakit jantung.
- Empat parameter yang dapat meningkatkan tingkat akurasi penelitian klasifikasi ini yaitu max-depth, n-estimator, min-samples-leaf, dan min-samples-split. Parameter ini dapat menghasilkan akurasi jika diatur value nya naik, jika value nya diatur menurun maka tingkat akurasi juga akan menurun. Value dari parameter terbaik yaitu tidak kurang atau lebih dari value yang diatur pada percobaan ke-8.
-

E. Ucapan Terima Kasih

Saya ingin mengucapkan terima kasih kepada semua yang telah berkontribusi dalam mendukung penelitian ini sepanjang perjalanan dari awal hingga akhir. Serta, saya ingin menyampaikan penghargaan yang besar kepada David Lapp, seorang pengguna Kaggle, yang telah berbagi data yang sangat berharga untuk penelitian ini.

F. Referensi

[1] D. Yana *et al.*, "Penerapan Metode Teorema Bayes Pada Sistem Pakar Untuk Mendiagnosa Gangguan Sistem Kardioveskular Pada Rumah Sakit Umum Pusat Haji Adam Malik," 2020.

[2] Rizal Fadli, "Penyakit Jantung." Accessed: Oct. 06, 2023. [Online]. Available: https://www.halodoc.com/kesehatan/nyeri-dada

[3] Adi Ahdiat, "Kasus Penyakit Katastropik di Indonesia Meningkat pada 2022."

[4] Pittara, "BAHAYA SERANGAN JANTUNG." [Online]. Available: https://puskesmas.kuburayakab.go.id/sungai-durian/read/173/bahaya-serangan-jantung#:~:text=Serangan jantung yang parah atau,syok kardiogenik%2C dan henti jantung.

[5] Y. Widiastiwi and I. Ernawati, "Klasifikasi Penyakit Batu Ginjal Menggunakan Algoritma Decision Tree C4 . 5 Dengan Membandingkan Hasil Uji Akurasi," *Jurnal IKRA-ITH INFORMATIKA*, vol. 5, no. 2, p. 128, 2021.

[6] S. Mohan, C. Thirumalai, and G. Srivastava, "Effective heart disease prediction using hybrid machine learning techniques," *IEEE Access*, vol. 7, pp. 81542–81554, 2019, doi: 10.1109/ACCESS.2019.2923707.

[7] A. Bhowmick, K. D. Mahato, C. Azad, and U. Kumar, "Heart Disease Prediction Using Different Machine Learning Algorithms," *Proceedings - 2022 IEEE World Conference on Applied Intelligence and Computing, AIC 2022*, pp. 60–65, 2022, doi: 10.1109/AIC55036.2022.9848885.

[8] A. Nurmasani and Y. Pristyanto, "Algoritme Stacking Untuk Klasifikasi Penyakit Jantung Pada Dataset Imbalanced Class," *Pseudocode*, vol. 8, no. 1, pp. 21–26, 2021, doi: 10.33369/pseudocode.8.1.21-26.

[9] B. Shiva Shanta Mani and V. M. Manikandan, "Heart disease prediction using machine learning," *Handbook of Research on Disease Prediction Through Data Analytics and Machine Learning*, pp. 373–381, 2020, doi: 10.4018/978-1-7998-2742-9.ch018.

[10] Reinert Yosua Rumagit, "Imbalanced Dataset." [Online]. Available: https://socs.binus.ac.id/2019/12/26/imbalanced-dataset/

[11] A. Liaw and M. Wiener, "Classification and Regression by randomForest," *R News*, vol. 2, no. 3, pp. 18–22, 2002.

[12] M. Harsha Vardhan, M. Rajesh Kumar, M. Vardhini, S. Leela Varalakshmi, and M. Kumar, "Heart Disease Prediction Using Machine Learning," 2023. [Online]. Available: https://jespublication.com/

[13] S. Khairunnisa, A. Adiwijaya, and S. Al Faraby, "Pengaruh Text Preprocessing terhadap Analisis Sentimen Komentar Masyarakat pada Media Sosial Twitter (Studi Kasus Pandemi COVID-19)," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 5, no. 2, p. 406, Apr. 2021, doi: 10.30865/mib.v5i2.2835.

[14] R. J. , & R. D. B. Little, *Statistical analysis with missing data*. 2019.

[15] J. Hastie, T., Tibshirani, R., & Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd ed.)*. Springer US, 2009.

[16] Abd Mizwar A. Rahim, "Prediksi Stroke Menggunakan Metode Synthetic Minority Over-Sampling Technique Dan Xtreme Gradient Boosting," 2022.

[17] L. Fadilah, *Klasifikasi Random Forest pada Data Imbalanced Program Studi Matematika Universitas Islam Negeri Syarif Hidayatullah 2018 / 1439 H Klasifikasi Random Forest*. 2018.

[18] N. K. Dewi, S. Y. Mulyadi, and U. D. Syafitri, "Penerapan Metode Random Forest Dalam Driver Analysis," *Forum Statistika Dan Komputasi*, vol. 16, no. 1, pp. 35–43, 2012, [Online]. Available: http://journal.ipb.ac.id/index.php/statistika/article/view/5443

[19] P. Choirunisa, "Implementasi Artificial Inteligence Untuk Memprediksi Harga Penjualan Rumah Menggunakan Metode Random Forest dan Flask (Tugas Akhir)," 2020.

[20] R. A. Haristu and P. H. P. Rosa, "Penerapan Metode Random Forest untuk Prediksi Win Ratio Pemain Player Unknown Battleground," *MEANS (Media Informasi Analisa dan Sistem)*, vol. 4, no. 2, pp. 120–128, 2019, doi: 10.54367/means.v4i2.545.

[21] R. Supriyadi, W. Gata, N. Maulidah, and A. Fauzi, "Penerapan Algoritma Random Forest Untuk Menentukan Kualitas Anggur Merah," *E-Bisnis : Jurnal Ilmiah Ekonomi dan Bisnis*, vol. 13, no. 2, pp. 67–75, 2020, doi: 10.51903/e-bisnis.v13i2.247.

[22] S. Kuter, "Completing the machine learning saga in fractional snow cover estimation from MODIS Terra reflectance data: Random forests versus support vector regression," *Remote Sens Environ*, vol. 255, Mar. 2021, doi: 10.1016/j.rse.2021.112294.

[23] A. Kulkarni, D. Chong, and F. A. Batarseh, "Foundations of data imbalance and solutions for a data democracy," in *Data Democracy: At the Nexus of Artificial Intelligence, Software Development, and Knowledge Engineering*, Elsevier, 2020, pp. 83–106. doi: 10.1016/B978-0-12-818366-3.00005-8.

[24] K. L. Kohsasih and Z. Situmorang, "Analisis Perbandingan Algoritma C4.5 dan Naïve Bayes Dalam Memprediksi Penyakit Cerebrovascular," *Jurnal Informatika*, vol. 9, no. 1, pp. 13–17, Apr. 2022, doi: 10.31294/inf.v9i1.11931.

[25] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf Process Manag*, vol. 45, no. 4, pp. 427–437, 2009, doi: 10.1016/j.ipm.2009.03.002.

[26] D. B. Little, R. J. A., & Rubin, *Statistical analysis with missing data (2nd ed.). Wiley.* 2002.

[27] G. , W. D. , H. T. , & T. R. ames, *An Introduction to Statistical Learning*. Springer, 2013.

[28] T. , T. R. , & F. J. Hastie, *The Elements of Statistical Learning.* Springer, 2009.

[29] J. D. , M. N. B. , & D. A. Kelleher, *Fundamentals of Machine Learning for Predictive Data Analytics*. MIT Press, 2015.