Jurnal Informatika, Komputer dan Bisnis
Vol-5, Issue-1, 2025 (JIKOBIS)
E-ISSN: 2807-5587
https://jurnal.itbaas.ac.id/index.php/jikobis

Penggunaan Algoritma Naïve Bayes Untuk Klasifikasi Penyakit Kardiovaskular

Hermawan Nur Wahiid*[1], Rina Firliana[2], Arie Nugroho[3]
[1]Program Studi Sistem Informasi, Universitas Nusantara PGRI Kediri
E-mail: *[1]mawan6989@gmail.com, [2]rina@unpkediri.ac.id, [3]arienugroho@unpkediri.ac.id

Abstrak

Penyakit kardiovaskular merupakan salah satu faktor utama penyebab kematian global, sehingga diperlukan upaya deteksi dini untuk mencegah komplikasi serius. Penelitian ini bertujuan mengimplementasikan algoritma Naïve Bayes untuk klasifikasi penyakit kardiovaskular dengan bantuan tahapan preprocessing data seperti seleksi fitur berbasis korelasi, serta teknik penyeimbangan data menggunakan SMOTE. Data yang digunakan diambil dari UC Irvine Machine Learning Repository dan berisi data rekam medis pasien gagal jantung. Proses penelitian meliputi pemisahan data, pengolahan, pelatihan model, dan evaluasi performa menggunakan indikator seperti akurasi, presisi, recall, dan F1-score. Model Naïve Bayes memberikan hasil terbaik dengan akurasi sebesar 81,66% pada skenario pelatihan 60% dan pengujian 40%. Selain itu, penggunaan seleksi fitur dan metode SMOTE terbukti mampu meningkatkan kinerja klasifikasi secara signifikan. Temuan ini menunjukkan bahwa pendekatan tersebut efektif untuk membantu proses klasifikasi awal penyakit kardiovaskular secara cepat dan tepat.

Kata Kunci*—Penyakit Kardiovaskular, Klasifikasi, Naïve Bayes, Preprocessing, SMOTE*

Abstract

Cardiovascular disease is one of the leading causes of death worldwide, making early detection essential to prevent severe complications. This study aims to implement the Naïve Bayes algorithm for the classification of cardiovascular disease, supported by data preprocessing steps such as correlation-based feature selection and data balancing using the SMOTE technique. The dataset used was obtained from the UC Irvine Machine Learning Repository and contains clinical records of heart failure patients. The research process includes data separation, preprocessing, model training, and performance evaluation using metrics such as accuracy, precision, recall, and F1-score. The Naïve Bayes model achieved its best result with an accuracy of 81.66% using a 60% training and 40% testing split. Furthermore, the application of feature selection and SMOTE significantly improved the model's classification performance. These findings indicate that the proposed approach is effective in supporting fast and accurate early classification of cardiovascular disease.

Keywords*—Cardiovascular Disease, Classification, Naïve Bayes, Preprocessing, SMOTE*

1. PENDAHULUAN

Penyakit kardiovaskular merupakan salah satu masalah kesehatan paling serius di dunia, dengan tingkat kematian yang terus meningkat dari tahun ke tahun. Gangguan ini mencakup berbagai kondisi yang melibatkan jantung dan pembuluh darah, seperti hipertensi, penyakit jantung

koroner, stroke, dan gagal jantung [1]. Berdasarkan laporan dari Organisasi Kesehatan Dunia (WHO), lebih dari 17 juta orang meninggal setiap tahunnya akibat penyakit kardiovaskular. Berbagai faktor risiko seperti usia, riwayat keluarga, hipertensi, diabetes, obesitas, serta gaya hidup tidak sehat, menjadi penyebab utama meningkatnya prevalensi penyakit ini. Oleh karena itu, deteksi dini menjadi aspek yang sangat penting dalam upaya pencegahan dan penanganan yang lebih efektif.

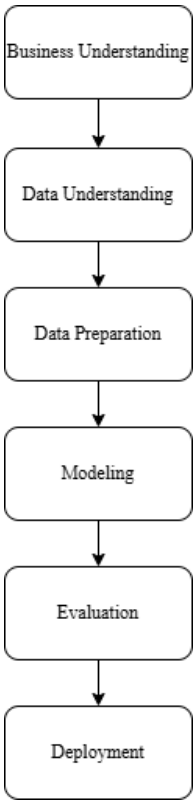
Seiring berkembangnya teknologi informasi, pemanfaatan data mining dalam bidang medis telah menjadi pendekatan penting untuk mendukung proses pengambilan keputusan klinis. Data mining merupakan proses yang diterapkan pada basis data berskala besar dengan memanfaatkan kecerdasan buatan, metode matematika, statistika, dan pembelajaran mesin guna menemukan pola, menyimpulkan informasi, serta mengungkap pengetahuan yang bernilai dan relevan [2]. Salah satu metode yang banyak digunakan dalam klasifikasi data medis adalah algoritma Naïve Bayes. Naïve Bayes adalah teknik statistik yang bersifat sederhana, namun efektif dalam menghasilkan akurasi tinggi dan tingkat kesalahan yang rendah saat melakukan klasifikasi [3]. Klasifikasi dalam data mining merupakan tahapan untuk mengelompokkan objek ke dalam kategori tertentu yang telah ditetapkan sebelumnya [4]. Metode ini digunakan untuk memperkirakan nilai suatu variabel yang belum diketahui dengan memanfaatkan informasi dari variabel lain yang telah diketahui sebelumnya [5]. Proses klasifikasi berbasis machine learning dapat membantu tenaga medis untuk mengidentifikasi pasien dengan risiko tinggi penyakit kardiovaskular berdasarkan data rekam medis yang tersedia.

Namun, tantangan umum dalam pengolahan data medis adalah ketidakseimbangan distribusi kelas dan keberadaan fitur yang tidak relevan, yang dapat mengurangi performa model klasifikasi. Oleh karena itu, dalam penelitian ini digunakan pendekatan *preprocessing* yang mencakup seleksi fitur berbasis korelasi dan penyeimbangan data dengan metode SMOTE (*Synthetic Minority Oversampling Technique*), guna meningkatkan akurasi model. Dataset yang digunakan berasal dari *UC Irvine Machine Learning Repository*, yang berisi data klinis pasien gagal jantung. Evaluasi kinerja model dilakukan menggunakan metrik akurasi, presisi, *recall*, dan *F1-score*.

Beberapa studi terdahulu telah menunjukkan bahwa algoritma Naïve Bayes memiliki potensi yang baik dalam klasifikasi data medis. Sebagai contoh, penelitian oleh Rizqullah et al. (2024) memperoleh akurasi sebesar 76,59% dalam klasifikasi penyakit kardiovaskular menggunakan dataset publik. Fakta ini menunjukkan bahwa pengaruh metode praproses seperti seleksi fitur dan penyeimbangan data dapat memberikan kontribusi besar terhadap peningkatan performa model klasifikasi.

2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan CRISP-DM (CRoss-Industry Standard Process for Data Mining) sebagai metode penelitian, yaitu model yang bersifat iteratif dan interaktif, terdiri dari enam tahapan yang dirancang untuk menggali pengetahuan dari data [6]. Tahapan-tahapan tersebut divisualisasikan pada Gambar 1.



Gambar 1. Tahapan CRISP-DM

2.1. Sumber Data

Penelitian ini menggunakan dataset publik yang bersumber dari *UCI Machine Learning Repository*, yang berisi data terkait penyakit kardiovaskular. Dataset ini berasal dari penelitian terdahulu yang dilakukan oleh Rizqullah [7], dan telah dipublikasikan pada tahun 2024 untuk mendukung penelitian lanjutan di bidang yang sama. Data mencakup berbagai atribut penting yang berpotensi memengaruhi kondisi kardiovaskular, seperti usia, riwayat anemia, diabetes, tekanan darah, jumlah trombosit, jenis kelamin, kebiasaan merokok, dan atribut kesehatan lainnya. Dataset ini telah melalui proses pembersihan oleh penyusun sebelumnya, sehingga tidak mengandung missing value sejak awal diperoleh dari *UCI Machine Learning Repository*. Hal ini memudahkan proses analisis karena tidak diperlukan penanganan data hilang pada tahap *preprocessing*.

2.2. Teknik Pengumpulan Data

Teknik pengumpulan data dalam penelitian ini dilakukan melalui metode dokumentasi terhadap data sekunder yang tersedia secara daring. Dataset yang digunakan diperoleh dari platform *UCI Machine Learning Repository*, yang merupakan sumber data terbuka dan tepercaya di bidang pembelajaran mesin. Dataset yang digunakan berjudul *Heart Failure Clinical Records*, yang memuat 299 data pasien dan 13 atribut yang menggambarkan kondisi klinis. Data dikumpulkan dengan cara mengunduh file dalam format comma-separated values (CSV), yang berisi 13 atribut klinis, termasuk usia, tekanan darah, fraksi ejeksi, kadar kreatinin serum, serta status anemia dan merokok. Seluruh data digunakan sebagaimana adanya tanpa modifikasi, sehingga mencerminkan kondisi medis yang sebenarnya berdasarkan rekaman klinis yang tersedia dalam dataset tersebut.

2.3. Langkah Analisis Data

Langkah-langkah analisis yang dilakukan dalam penelitian ini untuk melakukan klasifikasi penyakit kardiovaskular adalah sebagai berikut:

1. Diawali dengan proses pengumpulan dan pemuatan data *dari UCI Machine Learning Repository* dalam format CSV.
2. Dilanjutkan dengan tahap preprocessing data, seperti seleksi fitur menggunakan korelasi dan penyeimbangan data menggunakan teknik SMOTE.
3. Setelah data siap, dilakukan proses pelatihan model menggunakan algoritma Naïve Bayes dengan pendekatan GaussianNB dari pustaka Scikit-learn.
4. Proses implementasi dilakukan menggunakan platform Jupyter Notebook berbasis Python.
5. Model kemudian diuji dan dianalisis menggunakan metrik evaluasi seperti akurasi, presisi, *recall*, *F1-score*, serta confusion matrix untuk melihat kinerja model dalam mengklasifikasikan data medis.
6. Evaluasi ini dilakukan untuk menilai seberapa efektif model dalam mengenali pasien berisiko penyakit kardiovaskular berdasarkan pendekatan klasifikasi Naïve Bayes.

3. HASIL DAN PEMBAHASAN

Penelitian ini menerapkan pendekatan CRISP-DM (*Cross-Industry Standard Process for Data Mining*) dalam membangun model klasifikasi penyakit kardiovaskular menggunakan algoritma Naïve Bayes. Pendekatan ini memberikan kerangka kerja yang sistematis dan terstruktur, dimulai dari pemahaman konteks permasalahan medis, eksplorasi dan persiapan data, pembangunan model klasifikasi, hingga evaluasi performa model secara kuantitatif menggunakan metrik seperti akurasi, presisi, *recall*, dan *F1-score*.

3.1. Business Understanding

Penyakit kardiovaskular menjadi salah satu faktor utama penyebab kematian secara global, sehingga deteksi dini menjadi sangat krusial untuk mencegah komplikasi serius dan meningkatkan harapan hidup pasien. Namun demikian, proses diagnosis yang tepat seringkali memerlukan waktu yang cukup lama, biaya tinggi, dan keterlibatan tenaga medis dengan keahlian tertentu. Kondisi ini mendorong perlunya solusi alternatif yang dapat mendukung proses diagnosis awal secara lebih efisien dan akurat. Dalam konteks klasifikasi data medis, algoritma Naïve Bayes dikenal luas karena kemampuannya dalam menangani data dengan cepat dan memberikan hasil prediksi yang cukup andal. Oleh karena itu, pemanfaatan algoritma ini diharapkan dapat menjadi pendekatan yang lebih praktis dan efektif dalam mengidentifikasi penyakit kardiovaskular sejak dini, sekaligus meringankan beban kerja tenaga medis dalam pengambilan keputusan awal.

3.2. Data Understanding

Tahapan ini menganalisis hubungan antar fitur dalam dataset serta keterkaitannya dengan variabel target. Untuk keperluan ini, digunakan metode korelasi Pearson guna menghitung tingkat keterhubungan antara masing-masing fitur. Tujuan dari analisis ini adalah untuk mengidentifikasi kekuatan serta arah hubungan antar variabel, sehingga fitur yang memiliki korelasi tinggi satu sama lain dapat dihindari karena berpotensi menimbulkan redundansi dan menyebabkan *overfitting* pada model. Di sisi lain, fitur yang menunjukkan korelasi signifikan terhadap variabel target *DEATH_EVENT* dianggap penting dan relevan untuk dipertahankan dalam proses pelatihan model.

Untuk analisis korelasi saya menggunakan Jupyter Notebook. Berikut potongan kode programnya seperti di bawah ini.

```
1. #Features and target
```

```
2. X = df.drop(columns='DEATH_EVENT')  # Memisahkan fitur
3. y = df['DEATH_EVENT']  # Kolom target
4. #Korelasi dengan target
5. correlation = df.corr()
6. print("\nKorelasi Fitur dengan Target (DEATH_EVENT):")
7.print(correlation['DEATH_EVENT'].sort_values(ascending=False))
```

Korelasi yang dihitung menggambarkan hubungan antara setiap atribut dengan *DEATH_EVENT*, yaitu label yang menunjukkan apakah pasien meninggal (1) atau tetap hidup (0) selama masa observasi. Nilai korelasi berada pada rentang antara -1 hingga 1, dimana nilai 1 menunjukkan korelasi positif sempurna (kenaikan satu fitur diikuti oleh fitur lain), nilai -1 menunjukkan korelasi negatif sempurna (kenaikan satu fitur diiringi penurunan fitur lain) dan nilai 0 mengindikasikan tidak adanya hubungan linier antar variabel.

```
Korelasi Fitur dengan Target (DEATH_EVENT):
DEATH_EVENT                1.000000
serum_creatinine           0.294278
age                        0.253729
high_blood_pressure        0.079351
anaemia                    0.066270
creatinine_phosphokinase   0.062728
diabetes                  -0.001943
sex                       -0.004316
smoking                   -0.012623
platelets                 -0.049139
serum_sodium              -0.195204
ejection_fraction         -0.268603
time                      -0.526964
```

Gambar 2. Tampilan hasil korelasi fitur terhadap target *DEATH_EVENT*

Tabel 1. Hasil fitur prioritas untuk korelasi terhadap target

Fitur	Korelasi ke DEATH_EVENT	Artinya
Time	-0.53	Waktu pengamatan berbanding terbalik dengan kematian. Semakin lama pasien bertahan, semakin kecil peluang meninggal.
ejection_fraction	-0.27	Semakin kecil ejection fraction, semakin besar peluang pasien meninggal.
serum_creatinine	0.29	Semakin tinggi serum kreatinin, semakin tinggi risiko meninggal.
Age	0.25	Pasien yang lebih tua cenderung lebih berisiko meninggal.

3.3. Data Preparation

Pada tahap ini dilakukan beberapa proses *preprocessing* untuk memastikan data agar optimal digunakan dalam model klasifikasi. Proses dimulai dengan melakukan seleksi fitur, yakni dengan mengidentifikasi atribut-atribut yang paling relevan untuk mengurangi kompleksitas data dan mengeliminasi fitur yang kurang berpengaruh. Tujuan dari tahapan ini adalah memastikan bahwa hanya variabel yang memiliki kontribusi terhadap hasil prediksi yang digunakan dalam pemodelan. Setelah fitur relevan ditentukan, data dipisahkan menjadi dua bagian, yaitu fitur sebagai variabel independen dan target sebagai variabel dependen. Selanjutnya, dataset dibagi menjadi dua subset, yakni data pelatihan (*training set*) dan data pengujian (*testing set*).

Untuk mengatasi permasalahan distribusi kelas yang tidak seimbang, diterapkan metode SMOTE (*Synthetic Minority Oversampling Technique*), yang berfungsi untuk meningkatkan

data pada kelas minoritas sehingga distribusi kelas menjadi lebih proporsional. Seluruh tahapan *preprocessing* ini bertujuan untuk meningkatkan kinerja model dalam melakukan klasifikasi terhadap penyakit kardiovaskular secara lebih akurat.

3.3.1. Seleksi Fitur

Seleksi fitur merupakan tahap penting dalam proses pembangunan model, di mana sejumlah fitur yang paling relevan dipilih dari keseluruhan atribut dalam dataset [8]. Dalam konteks algoritma Naïve Bayes, langkah ini menjadi krusial karena memungkinkan model untuk memfokuskan proses pembelajaran hanya pada atribut-atribut yang memiliki pengaruh signifikan terhadap hasil prediksi. Dengan mengurangi fitur yang tidak diperlukan, kompleksitas model dapat diminimalkan, sehingga akurasi prediksi dapat ditingkatkan dan potensi gangguan dari informasi yang tidak relevan dapat ditekan. Selain itu, seleksi fitur juga memberikan gambaran yang lebih jelas terhadap karakteristik data yang digunakan.

Pada penelitian ini digunakan *Filter Method* dengan pendekatan berbasis korelasi. Metode ini menggunakan teknik statistik untuk menilai kekuatan hubungan antara masing-masing fitur terhadap variabel target [9]. Fitur-fitur yang memiliki korelasi tinggi dengan label target akan dipertahankan, sedangkan yang tidak relevan akan dieliminasi. Pemilihan metode ini didasarkan pada kelebihannya yang efisien, tidak bergantung pada algoritma klasifikasi tertentu, serta mampu mengurangi risiko *overfitting* dan mempercepat proses pelatihan model.

Berdasarkan hasil analisis korelasi yang telah dilakukan, tahap selanjutnya adalah melakukan seleksi fitur untuk menyaring atribut-atribut yang paling berpengaruh terhadap *output* yang akan diprediksi. Proses seleksi fitur memiliki peranan penting dalam pembelajaran mesin karena dapat menyederhanakan struktur model, mempercepat waktu pemrosesan, serta meningkatkan performa prediksi dengan mengurangi efek dari atribut yang kurang relevan. Atribut yang menunjukkan pengaruh rendah terhadap variabel target akan dikeluarkan dari proses pelatihan model agar tidak mengganggu kualitas prediksi. Untuk potongan kode program proses seleksi fitur di bawah ini.

```
1. #Features and target
2. X = df.drop(columns='DEATH_EVENT')  #Memisahkan fitur
3. y = df['DEATH_EVENT']  # Kolom target
4. #Korelasi dengan target
5. correlation = df.corr()
6. print("\nKorelasi Fitur dengan Target (DEATH_EVENT):")
7.print(correlation['DEATH_EVENT'].sort_values(ascending=False))
8. #Pilih fitur berdasarkan korelasi signifikan
9. selected_features = ['age', 'ejection_fraction', 'serum_creatinine', 'time']
10. X = df[selected_features]  #Menggunakan fitur yang terpilih
```

Kode program tersebut digunakan untuk melakukan pemilihan fitur dengan mempertimbangkan tingkat korelasinya terhadap variabel target *DEATH_EVENT*. Proses diawali dengan memisahkan data menjadi fitur independen (X) dan target dependen (y). Kemudian, korelasi antar fitur dihitung menggunakan metode Pearson untuk menilai kekuatan hubungan antara masing-masing atribut dengan target. Berdasarkan nilai korelasi tersebut, dipilih beberapa fitur yang paling berpengaruh, yaitu *age*, *ejection_fraction*, *serum_creatinine*, dan *time*. Fitur-fitur ini dianggap relevan dan digunakan dalam pelatihan model guna menyederhanakan struktur data serta meningkatkan efektivitas proses klasifikasi.

3.3.2. Penanganan Imbalance Data dengan SMOTE

SMOTE (*Synthetic Minority Oversampling Technique*) merupakan metode *resampling* yang dirancang untuk menangani ketidakseimbangan kelas dalam dataset dengan cara menambah jumlah data pada kelas minoritas melalui pembuatan data sintetis [10]. Berbeda dengan pendekatan duplikasi sederhana, SMOTE menghasilkan data baru dengan melakukan

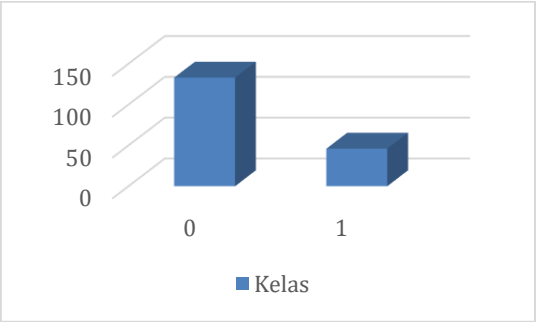
interpolasi antar data pada kelas minoritas yang sudah ada. Dengan demikian, data tambahan yang dihasilkan lebih bervariasi dan representative.

Penerapan SMOTE dalam proses pelatihan model bertujuan agar distribusi antar kelas menjadi lebih seimbang. Hal ini memungkinkan algoritma Naïve Bayes dalam mengenali pola-pola yang terdapat pada kelas minoritas secara lebih efektif. Dengan distribusi yang lebih merata, model dapat mengurangi kecenderungan bias terhadap kelas mayoritas, sehingga berdampak positif pada peningkatan performa model, khususnya pada metrik *recall* dan *F1-score* dua indikator yang sangat penting ketika kelas minoritas memiliki peran krusial dalam analisis klasifikasi.

Distribusi pada variabel target *DEATH_EVENT* dianalisis untuk mengidentifikasi adanya ketimpangan jumlah antar kelas. Hasil analisis menunjukkan bahwa jumlah pasien yang bertahan hidup (kelas 0) secara signifikan lebih banyak dibandingkan pasien yang meninggal (kelas 1). Ketidakseimbangan ini menjadi tantangan penting dalam proses klasifikasi, karena model cenderung lebih akurat dalam mengenali kelas mayoritas, namun kesulitan dalam mendeteksi kelas minoritas. Hal ini dapat berdampak negatif terhadap performa model, terutama pada metrik seperti *recall* dan *F1-score*, yang sensitif terhadap kesalahan dalam prediksi kelas minoritas.

Tabel 2. Distribusi kelas sebelum SMOTE

Kelas 0	Kelas 1
133	46



Gambar 3. Distribusi kelas (target) sebelum SMOTE

Berikut adalah potongan kode program untuk SMOTE.

```
#Mengecek distribusi data sebelum SMOTE (Imbalance check)
print("Distribusi kelas sebelum SMOTE pada training data:")
print(y_train.value_counts())  #Menampilkan jumlah sampel per kelas
#Plot distribusi kelas sebelum SMOTE
plt.figure(figsize=(6,4))
sns.countplot(x=y_train)
plt.title("Distribusi Kelas Sebelum SMOTE")
plt.xlabel("Kelas")
plt.ylabel("Jumlah Sampel")
plt.show()
#Mengatasi imbalance data dengan SMOTE (oversampling kelas minoritas)
smote = SMOTE(random_state=42)
X_train_res, y_train_res = smote.fit_resample(X_train, y_train)
#Mengecek distribusi data setelah SMOTE (Balance check)
print("\nDistribusi kelas setelah SMOTE pada training data:")
print(y_train_res.value_counts())  #Menampilkan jmlh sampel per kelas setelah SMOTE
#Plot distribusi kelas setelah SMOTE
plt.figure(figsize=(6,4))
sns.countplot(x=y_train_res)
```

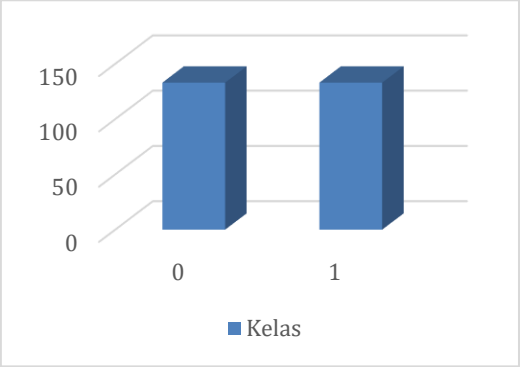
```
20. plt.title("Distribusi Kelas Setelah SMOTE")
21. plt.xlabel("Kelas")
22. plt.ylabel("Jumlah Sampel")
23. plt.show()
```

Untuk menangani permasalahan distribusi kelas yang tidak seimbang, menerapkan metode SMOTE (*Synthetic Minority Oversampling Technique*) yang tersedia dalam pustaka *imblearn*. Berbeda dengan metode penyalinan sederhana, SMOTE menghasilkan data sintetis baru untuk kelas minoritas melalui proses interpolasi antar data yang saling berdekatan. Teknik ini bertujuan untuk menciptakan distribusi kelas yang lebih proporsional, sehingga model dapat mempelajari pola dari kedua kelas secara adil dan tidak bias terhadap kelas mayoritas.

Setelah proses SMOTE selesai dijalankan, langkah selanjutnya adalah memverifikasi kembali distribusi pada variabel target guna memastikan bahwa jumlah data antara kelas mayoritas dan minoritas telah berimbang. Verifikasi ini dilakukan dengan menghitung jumlah *instance* pada tiap kelas dan memvisualisasikannya untuk memudahkan analisis. Apabila jumlah kedua kelas sudah seimbang, maka data dinyatakan siap untuk digunakan dalam proses pelatihan model klasifikasi.

Tabel 3. Distribusi kelas setelah SMOTE

Kelas 0	Kelas 1
133	133



Gambar 4. Distribusi kelas (target) sebelum SMOTE

3.3.3. Data Splitting

Dataset dibagi menjadi dua bagian, yakni data latih (*training set*) dan data uji (*testing set*). Komposisi pembagian yang diterapkan adalah 60% untuk pelatihan model dan 40% untuk pengujian. Agar proses pembagian bersifat konsisten dan dapat direplikasi, digunakan parameter *random_state* yang diatur secara eksplisit. Pembagian ini memiliki peran penting dalam menilai kinerja model terhadap data yang belum pernah digunakan dalam proses pelatihan, sehingga dapat memberikan evaluasi yang lebih objektif terhadap kemampuan model dalam melakukan generalisasi terhadap data baru.

3.4. Modeling (Naïve bayes)

Naïve Bayes merupakan algoritma klasifikasi yang menggunakan prinsip probabilistik dan pendekatan statistik untuk mengelompokkan data ke dalam kelas tertentu, dengan mendasarkan prosesnya pada identifikasi kesamaan dan perbedaan antar atribut, serta menerapkan konsep Teorema Bayes dalam perhitungannya [11]. Algoritma ini dikenal luas karena proses

pelatihannya yang sederhana dan efisien, namun tetap mampu memberikan tingkat akurasi yang kompetitif dalam tugas klasifikasi [12].

Dalam penerapannya pada klasifikasi penyakit kardiovaskular, Naïve Bayes dimanfaatkan untuk memprediksi potensi risiko seorang pasien terhadap penyakit jantung berdasarkan data medis yang tersedia. Keunggulan utama dari metode ini terletak pada kemampuannya memproses data dalam jumlah besar dengan cepat, serta kesederhanaan model yang tetap dapat memberikan hasil prediksi yang memadai, meskipun didasarkan pada asumsi independensi antar fitur yang tidak selalu sepenuhnya terpenuhi dalam praktik.

Model klasifikasi selanjutnya dikembangkan menggunakan algoritma Gaussian Naïve Bayes yang disediakan oleh pustaka *scikit-learn*. Algoritma ini menerapkan prinsip dasar dari Teorema Bayes, yaitu dengan menggabungkan nilai probabilitas awal (*prior*) dan kemungkinan kemunculan data (*likelihood*) untuk menghitung probabilitas akhir (*posterior*) bahwa suatu data termasuk ke dalam kelas tertentu. Proses pelatihan dilakukan menggunakan data yang telah melalui tahapan seleksi fitur dan penyeimbangan kelas, dengan harapan model dapat mengenali pola secara lebih efektif dan akurat.

Rumus:

$$P(C \mid X) = \frac{P(X|C).P(C)}{P(X)}$$

Dimana X adalah sampel data dengan *class* (*label*) yang belum diketahui, C yaitu hipotesis bahwa X termasuk ke dalam *class (label*) tertentu, P(C) adalah probabilitas dari hipotesis C, P(X) adalah peluang terjadinya data sampel yang diamati (berdasarkan probabilitas C) dan P(X|C) adalah probabilitas data sampel X dengan asumsi bahwa hipotesis C benar.

Berikut kode program untuk implementasi model naïve bayes beserta evaluasi dan confusion matrix.

```
#Naive Bayes classifier
nb = GaussianNB()
nb.fit(X_train_res, y_train_res)
y_pred = nb.predict(X_test)
#Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)
f1 = f1_score(y_test, y_pred)
#Confusion matrix
cm = confusion_matrix(y_test, y_pred)
```

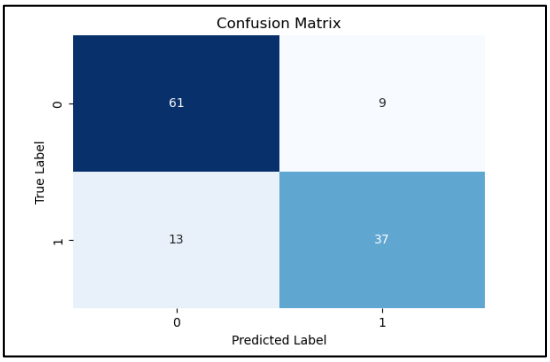
3.5. Evaluation

Model klasifikasi dievaluasi menggunakan sejumlah indikator performa, antara lain akurasi, presisi, *recall*, dan *F1-score*. Akurasi menggambarkan proporsi total prediksi yang tepat dibandingkan dengan keseluruhan data yang diuji. Presisi digunakan untuk menilai sejauh mana model mampu mengidentifikasi prediksi positif yang benar. Sementara itu, *recall* menunjukkan tingkat keberhasilan model dalam menemukan semua kasus positif yang sebenarnya ada dalam data. Adapun *F1-score* merupakan rata-rata harmonis dari presisi dan *recall*, yang memberikan gambaran menyeluruh mengenai keseimbangan performa model dalam mengenali kelas positif secara akurat dan menyeluruh.

```
Hasil Evaluasi Model:
      Metric     Score
0   Accuracy  0.816667
1  Precision  0.804348
2     Recall  0.740000
3   F1-Score  0.770833
```

Gambar 5. Hasil evaluasi model

Selain menggunakan metrik evaluasi numerik, analisis juga dilengkapi dengan confusion matrix untuk mengamati distribusi hasil klasifikasi yang benar maupun salah. Matriks ini menyajikan jumlah prediksi berdasarkan empat kategori, yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Dengan menelaah *confusion matrix*, dapat diperoleh hasil mengenai kecenderungan model dalam melakukan kesalahan, apakah lebih sering terjadi pada prediksi kelas positif atau negatif, serta mengidentifikasi jenis kesalahan yang paling dominan selama proses klasifikasi.



Gambar 6. Hasil *confusion matrix*

3.6. Deployment

Dalam penelitian ini, tahap *deployment* difokuskan pada penyajian model klasifikasi penyakit kardiovaskular berbasis algoritma Naïve Bayes yang telah dibangun dan dievaluasi. Hasil model dapat digunakan sebagai dasar untuk mengembangkan sistem bantu keputusan medis (*decision support system*) yang berfungsi dalam proses *skrining* awal terhadap pasien berisiko.

Model yang dihasilkan mampu memproses data klinis seperti usia, tekanan darah, kadar kreatinin, serta parameter lainnya untuk memberikan prediksi cepat mengenai kemungkinan risiko penyakit kardiovaskular. Meskipun penerapan langsung ke lingkungan klinis belum dilakukan, model ini berpotensi untuk diintegrasikan ke dalam sistem digital, seperti aplikasi berbasis web atau perangkat lunak desktop, guna mendukung proses diagnosis awal secara efisien.

4. KESIMPULAN

Berdasarkan penelitian dan uraian diatas, maka didapatkan kesimpulannya sebagai berikut:

1. Model Naïve Bayes terbukti efektif dalam mengklasifikasikan penyakit kardiovaskular, dengan akurasi mencapai 81,66% pada skenario pembagian data 60% untuk pelatihan dan 40% untuk pengujian. Hasil ini menunjukkan peningkatan performa dibandingkan

penelitian terdahulu yang memperoleh akurasi 76,59% pada dataset serupa, sehingga pendekatan yang digunakan dalam penelitian ini dinilai lebih optimal.
2. Tahap *preprocessing* data, khususnya melalui seleksi fitur berbasis korelasi, berhasil menyederhanakan kompleksitas data dan meningkatkan akurasi model dengan hanya mempertahankan fitur-fitur yang relevan terhadap target klasifikasi.
3. Penerapan metode SMOTE (*Synthetic Minority Oversampling Technique*) secara efektif mengatasi masalah ketidakseimbangan kelas dalam dataset. Teknik ini berkontribusi pada peningkatan kemampuan model dalam mengenali kelas minoritas, yang tercermin dari performa evaluasi yang lebih baik, terutama dalam hal sensitivitas dan keseimbangan prediksi antara kedua kelas.
4. Kombinasi antara seleksi fitur dan teknik SMOTE terbukti sebagai pendekatan yang signifikan dalam meningkatkan kualitas klasifikasi pada data medis yang tidak seimbang, dan dapat dijadikan acuan dalam pengembangan sistem klasifikasi serupa di bidang kesehatan.

5. SARAN

Untuk penelitian selanjutnya disarankan agar digunakan dataset yang lebih besar dan beragam guna meningkatkan kemampuan generalisasi model terhadap populasi yang lebih luas. Selain itu, pengujian terhadap algoritma klasifikasi lain seperti Random Forest, SVM, atau XGBoost dapat dilakukan untuk membandingkan kinerja dengan Naïve Bayes dan mengidentifikasi metode terbaik dalam konteks klasifikasi penyakit kardiovaskular. Penambahan atribut klinis yang lebih kompleks dan integrasi dengan data *real-time* dari sistem rekam medis elektronik juga dapat meningkatkan akurasi prediksi. Di sisi lain, pendekatan evaluasi berbasis validasi silang (*cross-validation*) perlu dipertimbangkan untuk memperoleh hasil evaluasi yang lebih stabil dan representatif.

DAFTAR PUSTAKA

[1] A. Nugroho, A. Bimo Gumelar, A. G. Sooai, D. Sarvasti, and P. L. Tahalele, “Perbandingan Performansi Algoritma Pengklasifikasian Terpandu Untuk Kasus Penyakit Kardiovaskular,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 1, no. 3, pp. 998–1006, 2020, doi: https://doi.org/10.29207/resti.v4i5.2316.

[2] E. Priyanto, E. Daniati, and A. Ristyawan, “Implementasi Metode K-Nearest Neighbor Untuk Memprediksi Kondisi Cuaca Penulis Korespondensi,” *SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi)*, vol. 8, pp. 2549–7952, 2024, doi: https://doi.org/10.29407/inotek.v8i1.4954.

[3] A. Dewi, A. Safira Surya, and Y. Yamasari, “Penerapan Algoritma Naïve Bayes (NB) untuk Klasifikasi Penyakit Jantung,” *Journal of Informatics and Computer Science*, vol. 05, 2024, doi: https://doi.org/10.26740/jinacs.v5n03.p447-455.

[4] A. Desiani, M. Akbar, I. Irmeilyana, and A. Amran, “Implementasi Algoritma Naïve Bayes dan Support Vector Machine (SVM) Pada Klasifikasi Penyakit Kardiovaskular,” *Jurnal Teknik Elektro dan Komputasi (ELKOM)*, vol. 4, no. 2, pp. 207–214, Aug. 2022, doi: 10.32528/elkom.v4i2.7691.

[5] A. F. Riany and G. Testiana, “PENERAPAN DATA MINING UNTUK KLASIFIKASI PENYAKIT JANTUNG KORONER MENGGUNAKAN ALGORITMA NAÏVE

BAYES," *MDP Student Conference (MSC)*, 2023, doi: https://doi.org/10.35957/mdp-sc.v2i1.4388.

[6] A. Ristyawan, A. Nugroho, and T. K. Amarya, "Optimasi Preprocessing Model Random Forest Untuk Prediksi Stroke," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 12, no. 1, pp. 29–44, 2025, doi: https://doi.org/10.35957/jatisi.v12i1.9587.

[7] M. F. Rizqullah, N. T. Raihana, and M. I. Jambak, "Komparasi Penerapan Algoritma C4.5, K-Nearest Neighbor, dan Naïve Bayes untuk Keberlangsungan Pasien Gagal Jantung," *KLIK: Kajian Informatika dan Komputer*, vol. 4, no. 5, pp. 2580–2587, 2024, doi: 10.30865/klik.v4i5.1788.

[8] A. Devia and B. Soewito, "Analisis Perbandingan Metode Seleksi Fitur untuk Mendeteksi Anomali pada Dataset CIC-IDS-2018," *Jurnal Teknologi Dan Sistem Informasi Bisnis-JTEKSIS*, vol. 5, no. 4, p. 572, 2023, doi: 10.47233/jteksis.v5i4.1069.

[9] Y. Setiawan, "Data Mining berbasis Nearest Neighbor dan Seleksi Fitur untuk Deteksi Kanker Payudara," *Jurnal Informatika: jurnal Pengembangan IT (JPIT)*, vol. 8, no. 2, 2023, doi: https://doi.org/10.30591/jpit.v8i2.4994.

[10] K. Pramayasa, I. Md, D. Maysanjaya, G. Ayu, and A. Diatri Indradewi, "Analisis Sentimen Program Mbkm Pada Media Sosial Twitter Menggunakan KNN Dan SMOTE," *SINTECH Journal*, vol. 6, 2023, doi: https://doi.org/10.31598/sintechjournal.v6i2.1372.

[11] I. Nurjanah, J. Karaman, I. Widaningrum, D. Mustikasari, and Sucipto, "Penggunaan Algoritma Naive Bayes Untuk Menentukan Pemberian Kredit Pada Koperasi Desa," 2023. doi: https://doi.org/10.47065/explorer.v3i2.766.

[12] H. A. N. Syifa, A. Nugroho, and R. Firliana, "Perbandingan Algoritma Naïve Bayes Classifier Dan K-Nearest Neighbors Untuk Analisis Sentimen Covid-19 Di Twitter," *Jurnal Ilmiah informatika*, vol. 11, 2023, doi: https://doi.org/10.33884/jif.v11i01.7069.

.