

Penerapan Algoritma *Decision Tree* dalam Klasifikasi Data “Framingham” Untuk Menunjukkan Risiko Seseorang Terkena Penyakit Jantung dalam 10 Tahun Mendatang

Agung Rachmat Raharja¹, Jayadi², Angga Pramudianto³, Yoki Muchsam⁴

Teknik Informatika Universitas Bandung¹²³⁴

¹agungmat@bandunguniversity.ac.id

²jayadi@bandunguniversity.ac.id

³anggapramudianto@bandunguniversity.ac.id

⁴yokimuchsam@bandunguniversity.ac.id

ABSTRAK

Penyakit jantung adalah salah satu penyakit paling mematikan di Indonesia dan di dunia, banyak dari penderita yang tidak mengetahui risiko penyakit jantung. Pada penelitian ini akan menganalisis dan mengidentifikasi faktor-faktor yang akan mempengaruhi risiko penyakit jantung. Dengan pendekatan algoritma decision tree yang merupakan salah satu metode machine learning yang dapat menghasilkan model prediktif berdasarkan serangkaian keputusan logis. Penerapan yang dilakukan adalah dengan klasifikasi data Framingham untuk menilai risiko penyakit jantung dalam 10 tahun mendatang. Hasilnya model Decision Tree yang digunakan untuk memprediksi risiko penyakit jantung berdasarkan dataset Framingham. Model ini mencapai akurasi 74% pada data pengujian.

Kata kunci : Penyakit Jantung, Data Mining, Decision Tree

ABSTRACT

Heart disease is one of the deadliest diseases in Indonesia and in the world, many of the sufferers do not know the risk of heart disease. This research will analyze and identify factors that will affect the risk of heart disease. With the decision tree algorithm approach which is one of the methods of machine learning that can produce predictive models based on a series of logical decisions. The application is done by classifying Framingham data to assess the risk of heart disease in the next 10 years. The result is a Decision Tree model used to predict the risk of heart disease based on the Framingham dataset. The model achieved 74% accuracy on the test data

Keywords: Heart Disease, Data Mining, Decision Tree

1. Pendahuluan

Penyakit jantung adalah salah satu penyebab utama kematian di seluruh dunia. Penyakit jantung adalah istilah umum untuk semua kelainan yang mempengaruhi jantung. Penyakit jantung artinya sama dengan penyakit jantung, namun bukan penyakit kardiovaskular. Penyakit jantung merupakan salah satu penyakit yang menyebabkan angka kematian yang cukup tinggi dalam dunia kesehatan. Penyebab penyakit ini jarang teridentifikasi dan serangan jantung merupakan salah satu penyakit paling mematikan di dunia, menyumbang 43,32% serangan jantung baru dan 12,91% kematian [1][2][3].

Penelitian ini akan menganalisis dan mengidentifikasi faktor-faktor risiko yang dapat mempengaruhi kemungkinan seseorang terkena penyakit jantung menjadi krusial dalam upaya pencegahan dan pengelolaan kesehatan masyarakat. Dalam era digital ini, penggunaan teknologi dan algoritma kecerdasan buatan telah menjadi pilihan yang efektif untuk menganalisis data kesehatan dan mengidentifikasi pola yang sulit dikenali secara manual.

Salah satu pendekatan yang telah terbukti efektif adalah penggunaan algoritma Decision Tree dalam klasifikasi data. Decision Tree adalah sebuah diagram alir yang berbentuk seperti struktur pohon yang mana setiap internal node menyatakan pengujian terhadap suatu atribut, setiap cabang menyatakan output dari pengujian tersebut dan leaf node menyatakan kelas-kelas atau distribusi kelas. Node yang paling atas disebut sebagai root node atau node akar, classifier inilah yang akan digunakan untuk menyusun kelas-kelas yang terkandung dari data, misalnya

untuk Decision Tree maka kelas-kelas tersebut digambarkan dalam bentuk pohon. [4] [5] Decision Tree merupakan metode machine learning yang dapat menghasilkan model prediktif berdasarkan serangkaian keputusan logis. Dalam konteks kesehatan, implementasi algoritma Decision Tree dapat membantu mengklasifikasikan risiko seseorang terkena penyakit jantung berdasarkan data yang terkait.

Penelitian ini akan fokus pada penerapan algoritma Decision Tree dalam klasifikasi data "Framingham" untuk menilai risiko seseorang terkena penyakit jantung dalam jangka waktu 10 tahun mendatang. Framingham Heart Study telah mengumpulkan data kesehatan selama beberapa dekade, menyediakan dataset kaya informasi yang melibatkan berbagai faktor seperti tekanan darah, kolesterol, riwayat merokok, dan faktor-faktor lain yang terkait dengan penyakit jantung.

Dengan menerapkan algoritma Decision Tree pada dataset Framingham, diharapkan dapat mengembangkan model klasifikasi yang akurat dan dapat diandalkan untuk menentukan risiko seseorang terkena penyakit jantung. Hasil dari studi ini diharapkan dapat memberikan kontribusi pada pengembangan strategi pencegahan yang lebih personal dan tepat sasaran, membantu masyarakat untuk mengambil langkah-langkah proaktif dalam menjaga kesehatan jantung mereka.

Untuk pemrograman menggunakan python. Python adalah bahasa pemrograman yang sangat powerful, serta sangat mudah dipelajari dan dibaca, karena mendekati bahasa manusia, Python adalah satu dari bahasa pemrograman tingkat tinggi yang bersifat interpreter, interaktif, object-oriented dan dapat beroperasi di hampir semua platform seperti keluarga Linux, Windows, Mac, dan platform lainnya. Python adalah bahasa pemrograman yang menggunakan interpreter untuk menjalankan kode programnya. Interpreter tersebut dapat menerjemahkan kode secara langsung, [6][7][8]

2. Landasan Teori

A. Penyakit Jantung

Salah satu penyakit jantung, disebabkan terutama karena penyempitan arteri koroner akibat aterosklerosis atau spasme atau kombinasi keduanya. Penyakit jantung koroner merupakan salah satu penyakit yang menakutkan dan masih menjadi masalah baik di negara maju maupun di negara berkembang. [9], Penyakit jantung merupakan penyakit yang sangat berbahaya bagi kelangsungan hidup manusia oleh sebab itu harus segera diatasi sejak dini munculnya gejala. [10].

B. Decision Tree

Algoritma Decision Tree C4.5 merupakan algoritma yang dapat digunakan untuk membentuk pohon keputusan. Pohon keputusan (Decision Tree) adalah salah satu metode yang cukup mudah untuk diinterpretasikan oleh manusia. [11]. Decision Tree adalah proses menemukan kumpulan pola atau fungsi-fungsi yang mendeskripsikan dan memisahkan kelas data satu dengan lainnya, untuk dapat digunakan untuk memprediksi data yang belum memiliki kelas data tertentu. [12]

C. Python

Python adalah salah satu bahasa pemrograman yang baru di masa sekarang, pada bahasa pemrograman ini kita lebih simpel dalam dan singkat dalam membuat sebuah program setiap program yang kita buat pasti dan pasti akan membutuhkan inputan dan hasil outputan. Dalam metode penginputannya pun bahasa ini sedikit berbeda, memang terlihat mudah tapi bukan berarti untuk dihiraukan karena pada kenyataan masih banyak yang kesulitan dalam membuat program pada python ini, dengan dibuatnya pembahasan ini semoga dapat membantu bagi pemula yang sedang belajar bahasa pemrograman. [13] dan menurut [7] Python adalah satu dari bahasa pemrograman tingkat tinggi yang bersifat interpreter, interaktif, object-oriented dan dapat beroperasi di hampir semua platform seperti keluarga Linux, Windows, Mac, dan platform lainnya. Python adalah salah satu bahasa pemrograman tingkat tinggi yang mudah dipelajari karena sintaks yang jelas dan elegan, yang dikombinasikan dengan penggunaan modul-modul yang mempunyai struktur data tingkat tinggi, efisien, dan siap langsung digunakan

3. Metode Penelitian

Penelitian ini menggunakan metode Decision Tree dalam menganalisis sebuah dataset yang berasal dari Kaggle dengan bertujuan untuk memprediksi resiko penyakit jantung pada 10 tahun yang akan datang. Dataset ini terdiri dari 4236 subjek penelitiandan 16 atribut yang meliputi umur, pendidikan, riwayat merokok, diabetes, dan variabel lainnya yang relevan.

Sebelum menjalankan Decision Tree, dataset melakukan tahap Data Processingterlebih dahulu, proses ini mencakup data understanding, data cleaning, explatory data analyst, data splitting dan modelling data. Pada proses selanjutnya membagi dataset menjadi data pelatihan dan data pengujian.

Langkah terakhir, menerapkan algoritma Decision Tree yang tersedia dalam pustaka scikit-learn di lingkungan pemrograman Python untuk mengklasifikasi data dan memprediksi data.

4. Hasil dan Pembahasan

Hasil analisis menggunakan Decison Tree pada dataset Framingham terhadap prediksi resiko penyakit jantung 10 tahun yang akan datang menunjukkan temuan yang relevan dikarenakan jumlah akurasi yang cukup besar secara keseluruhan. Dari berbagai Atribut yang dievaluasi, umur, tekanan darah, dan riwayat merokok muncul sebagai faktor yang paling signifikan dalam memprediksi resiko penyakit.

A. Proses Data Processing

Proses tersebut diawali dengan preprocessing yang terdiri atas pengumpulan data untuk menghasilkan data mentah (raw data) yang dibutuhkan oleh data mining, yang kemudian dilanjutkan dengan transformasi data untuk mengubah data mentah menjadi format yang dapat diproses oleh data mining, misalnya melalui filtrasi atau agregasi. Hasil transformasi data akan digunakan oleh analisis data untuk membangkitkan pengetahuan dengan menggunakan teknik seperti analisis statistik, machine learning, dan visualisasi informasi[14]. Dalam menganalisis dan mengkalasifikasi menggunakan Python, tahap awal melibatkan import dataset, lalu penggunaan library sangat penting untuk memuat DataFrame yang memungkinkan untuk menganalisis dan mengklasifikasi data lebih efisien

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
```

Gambar 1. Library yang digunakan pada process data processing dan klasifikasi

Gambar 1 secara grafis memperlihatkan beberapa library seperti numpy, pandas, matplotlib, dan sebagainya yang dimana fungsi library-library tersebut adalah untuk pengolahan data, visualisasi data, menginput data, dan menggunakan metode decision tree. Dengan adanya library tersebut, proses dalam data processing dan klasifikasi akan lebih efisien.

B. Data Understanding

Pada fase ini data yang akan diuji harus membuat target dari data rekam medis pasien dan fokus pada variabel atau sampel data yang akan diambil. Setelah itu sampel data rekam medis tersebut dicleaning target data dengan tujuan untuk mendapatkan data yang konsisten. [15] .Dalam memahami sebuah data untuk klasifikasi hal yang utama agar mendapatkan sebuah wawasan tentang dataset Framingham yang diolah. Analisis ini bertujuan untuk memahami struktur, karakteristik, dan potensi informasi yang terkandung dalam setiap atribut.

male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentTpp	diabetes	totChol	sysBP	diABP	BMI	heartRate	glucose	TenYearCVD
1	39	4.0	0	0.0	0.0	0	0	0	195.0	106.0	70.0	26.97	80.0	77.0	0
0	46	2.0	0	0.0	0.0	0	0	0	250.0	121.0	81.0	28.73	95.0	76.0	0
1	48	1.0	1	20.0	0.0	0	0	0	245.0	127.5	80.0	25.34	75.0	70.0	0
0	61	3.0	1	30.0	0.0	0	1	0	225.0	150.0	95.0	28.88	65.0	103.0	1
0	46	3.0	1	23.0	0.0	0	0	0	285.0	130.0	84.0	23.10	85.0	85.0	0
...
1	50	1.0	1	1.0	0.0	0	1	0	313.0	179.0	92.0	25.97	66.0	86.0	1
1	51	3.0	1	43.0	0.0	0	0	0	207.0	126.5	80.0	19.71	65.0	68.0	0
0	48	2.0	1	20.0	NaN	0	0	0	248.0	131.0	72.0	22.00	84.0	86.0	0
0	44	1.0	1	15.0	0.0	0	0	0	210.0	126.5	87.0	19.16	86.0	NaN	0
0	52	2.0	0	0.0	0.0	0	0	0	269.0	133.5	83.0	21.47	80.0	107.0	0

Gambar 2. Isi dataset Framingham

Dataset framing ham pada Gambar 2 menunjukkan terdiri dari 4236 baris dan 16 kolom, mencakup informasi beragam tentang subjek penelitian. Pada data tersebut menunjukkan tipe data yang berbeda antar atribut, selain itu terdapat juga nilai NaN.

D. Data Cleaning

Data cleansing atau data cleaning adalah kegiatan menganalisa kualitas data dengan cara memodifikasi, mengubah, atau menghapus data-data yang dianggap tidak perlu, tidak lengkap, data tidak akurat, atau memiliki format data atau file yang salah dalam basis data guna menghasilkan data berkualitas tinggi [16].

Data Cleaning atau pembersihan data adalah tahapan penting dalam penelitian ini yang bertujuan untuk memastikan integritas dan kualitas data.

```
df.isnull().sum()
male          0
age           0
education     105
currentSmoker 0
cigsPerDay    29
BPMeds        53
prevalentStroke 0
prevalentHyp  0
diabetes       0
totChol       50
sysBP         0
diaBP         0
BMI           19
heartRate     1
glucose       388
TenYearCHD    0
dtype: int64
```

Gambar 3. Melihat data yang kosong pada dataset Framingham

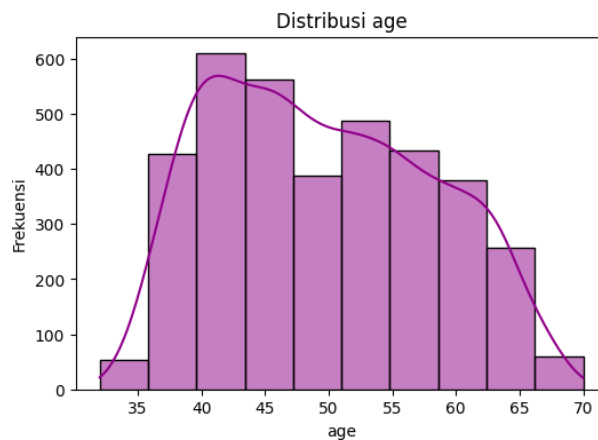
Gambar 3 memvisualisasikan distribusi nilai kosong atau missing values pada dataset Framingham. Analisis ini bertujuan untuk memahami sejauh mana nilai kosong memengaruhi integritas dataset dan membantu dalam pengambilan keputusan tentang penanganan nilai-nilai yang hilang. Nilai yang kosong dapat dihilangkan dengan beberapa perintah yang terdapat pada python, seperti `df.dropna(inplace=True)`

```
df_cleaned_rows = df.dropna()
df.dropna(inplace=True)
df.isnull().sum()
male          0
age           0
education     0
currentSmoker 0
cigsPerDay    0
BPMeds        0
prevalentStroke 0
prevalentHyp  0
diabetes       0
totChol       0
sysBP         0
diaBP         0
BMI           0
heartRate     0
glucose       0
TenYearCHD    0
dtype: int64
```

Gambar 4. Hasil Menghilangkan nilai kosong pada dataset Framingham

E. Exploratory Data Analysis

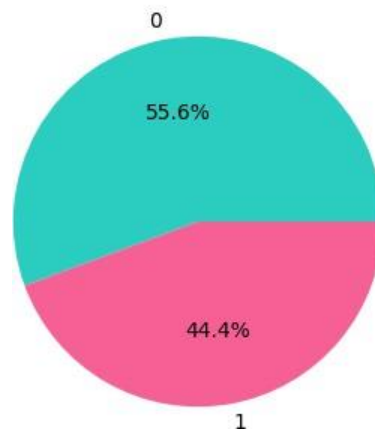
Tahap Exploratory Data Analysis merangkum karakteristik utama, melihat pola-pola dalam data, dan mengidentifikasi hubungan antar variabel. Tujuan utama dari EDA adalah untuk memahami struktur dataset, menemukan pola menarik atau anomali, dan membantu dalam pembuatan hipotesis yang dapat diuji lebih lanjut.



Gambar 5. Visualisasi data Framingham dengan subjek penelitian berupa age

Pada gambar 5 sumbu x menunjukkan rentang usia dan sumbu y menunjukkan frekuensi observasi pada setiap kelompok usia. Distribusi usia menunjukkan bahwa kelompok usia dengan frekuensi tertinggi berada di rentang 39-45 tahun, yang dapat diidentifikasi dengan tingginya bar pada interval tersebut. Ini menandakan bahwa usia-usia ini memiliki frekuensi observasi yang lebih tinggi daripada kelompok usia lainnya. Sebaliknya, frekuensi observasi paling rendah terjadi pada kelompok usia di bawah 36 tahun dan di atas 66 tahun. Hal ini terlihat dari tingginya bar pada interval di bawah 36 dan di atas 66, yang menunjukkan bahwa kelompok usia ini memiliki frekuensi yang lebih rendah dalam dataset.

Pie Chart untuk Kolom male



Gambar 6. Visualisasi data Framingham mengetahui jumlah male

Dalam pie chart di atas, proporsi distribusi antar kategori male (bernilai 1) dan non-male (bernilai 0) dapat jelas terlihat. Kategori male mendominasi dengan jumlah 44,4%, sementara kategori non-male menyumbang sebesar 56,5%. Proporsi ini memberikan gambaran tentang perbandingan relatif antara kategori male dan non-male dalam dataset.

Penemuan ini menunjukkan bahwa non-male memiliki presentase yang lebih tinggi dalam dataset dibandingkan dengan kategori male. Analisis lebih lanjut terkait perbandingan gender ini dapat memberikan wawasan lebih lanjut terhadap distribusi variabel gender dalam konteks studi ini.

F. Data Splitting

Untuk membangun dan mengevaluasi model klasifikasi data menggunakan decision tree dengan akurasi yang dapat diandalkan, dataset awal perlu dibagi menjadi dua subset: subset pelatihan (training set) dan subset pengujian (test set). Proses ini dikenal sebagai data splitting. Subset pelatihan digunakan untuk melatih model agar dapat memahami pola dan relasi dalam data. Di sisi lain, subset pengujian digunakan untuk menguji kinerja model pada data yang tidak pernah dilihat selama pelatihan.

```
] x = df.drop(columns = ['TenYearCHD'])
y = df['TenYearCHD']

print("x : ",x.shape)
print("y : ",y.shape)
```

Gambar 7. Code untuk membagi data set

Dataset yang digunakan dalam penelitian ini terdiri dari beberapa atribut yang potensial mempengaruhi risiko penyakit jantung dalam jangka waktu 10 tahun (TenYearCHD). Sebagai langkah pertama dalam persiapan data, dilakukan pemisahan dataset pada Gambar 7 menjadi dua bagian utama, yaitu atribut (features) dan label (target). Atribut-atribut ini diwakili oleh variabel x, yang mencakup semua kolom kecuali label TenYearCHD, sedangkan label tersebut diwakili oleh variabel y.

```
scaler = StandardScaler()
scaler.fit(x)
x = scaler.transform(x)
```

Gambar 8. Code proses standarisasi dataset

Dalam implementasi ini, gambar 8 digunakan metode standarisasi dengan menggunakan objek StandardScaler dari pustaka scikit-learn. Atribut numerik pada subset atribut (x) diukur dan dihitung reratanya, serta deviasi standarnya. Proses normalisasi dilakukan dengan mengurangi rerata dan membagi hasilnya dengan deviasi standar pada setiap atribut. Hasil normalisasi tersebut akan digunakan sebagai input pada model machine learning. Normalisasi atribut memastikan bahwa atribut-atribut dengan skala yang berbeda-beda dapat diperlakukan secara adil oleh model, meningkatkan interpretabilitas, dan menghindari dominasi oleh atribut dengan skala besar.

```
] x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=42)
```

Gambar 9. Code Pemisahan Dataset Framingham

Selanjutnya, dataset dibagi menjadi subset pelatihan (training set) dan subset pengujian (test set) untuk mengevaluasi kinerja model. Pemisahan ini penting untuk memastikan bahwa model yang dikembangkan mampu menggeneralisasi dengan baik pada data yang belum pernah dilihat sebelumnya. Dataset yang telah disiapkan sebelumnya, yaitu subset atribut (x) dan subset label (y), dibagi dengan proporsi tertentu:

- a. x_train dan y_train adalah subset pelatihan yang akan digunakan untuk melatih model.
- b. x_test dan y_test adalah subset pengujian yang digunakan untuk menguji kinerja model.

Proporsi pembagian Gambar 9 ditentukan oleh parameter test_size, dalam contoh ini diatur sebesar 0.2 (20%) untuk subset pengujian. Selain itu, penambahan parameter random_state pada nilai tertentu (dalam contoh di atas, 42) memastikan bahwa pembagian dataset dapat direproduksi dengan cara yang sama setiap kali dilakukan. Hal ini penting untuk konsistensi hasil eksperimen dan memungkinkan orang lain

mereproduksi hasil. Pemisahan dataset ini menjadilangkah kunci dalam evaluasi model, sehingga dapat diukur sejauh mana model mampu menghasilkan prediksi yang generalisasinya baik pada data baru.

G. Proses Klasifikasi data menggunakan Decision Tree

```
clf = DecisionTreeClassifier()
clf.fit(x_train, y_train)

y_pred = clf.predict(x_test)

decision_tree_acc = accuracy_score(y_test, y_pred)

print(classification_report(y_test, y_pred))
print('Akurasi Model Decision Tree : {:.2f}%'.format(decision_tree_acc * 100))
```

Gambar 10 Klasifikasi dengan Decision Tree

Gambar 10 memberikan gambaran tentang seberapa baik model Decision Tree dapat mengklasifikasikan data pengujian. Classification report memberikan informasi rinci tentang precision, recall, dan f1-score untuk setiap kelas, sementara akurasi menyajikan evaluasi keseluruhan performa model.

Implementasi dan evaluasi model ini merupakan langkah penting dalam proses penelitian ini, memberikan pemahaman tentang sejauh mana model dapat diandalkan dalam melakukan prediksi risiko penyakit jantung pada 10 tahun mendatang. Hasil-hasil ini memberikan gambaran tentang seberapa baik model Decision Tree dapat mengklasifikasikan data pengujian. Classification report memberikan informasi rinci tentang precision, recall, dan f1-score untuk setiap kelas, sementara akurasi menyajikan evaluasi keseluruhan performa model.

	precision	recall	f1-score	support
0	0.84	0.84	0.84	610
1	0.21	0.22	0.22	122
accuracy			0.73	732
macro avg	0.53	0.53	0.53	732
weighted avg	0.74	0.73	0.74	732

Akurasi Model Decision Tree : 73.36%

Gambar 11. Hasil Akurasi dengan Decision Tree

Pada Gambar 11 Hasil evaluasi performa model Decision Tree pada dataset Framingham ditunjukkan dalam classification report dan akurasi model berikut:

- Precision mengukur seberapa baik model dalam mengidentifikasi instance positif, dan dalam kasus ini, memiliki nilai 0.84 untuk kelas 0 (tidak ada risikopenyakit jantung) dan 0.21 untuk kelas 1 (risiko penyakit jantung).
- Recall mengukur sejauh mana model dapat mendeteksi semua instance positif. Nilai recall adalah 0.84 untuk kelas 0 dan 0.22 untuk kelas 1.
- F1-score adalah rata-rata harmonis antara precision dan recall, memberikan nilai 0.84 untuk kelas 0 dan 0.22 untuk kelas 1.

Akurasi model Decision Tree mencapai 73.36%, yang mengindikasikan seberapa baik model dapat melakukan prediksi secara keseluruhan. Meskipun akurasi mencapai tingkat yang cukup tinggi, perlu diingat bahwa evaluasi lebih lanjut mungkin diperlukan untuk memahami kinerja model dalam kasus kelas minoritas (kelas 1) yang memiliki recall dan f1-score yang rendah..

H. Proses Data Testing

Untuk menguji performa model Decision Tree yang telah dikembangkan, sebuah data pengujian baru diinput ke dalam model. Data pengujian ini mencakup atribut-atribut klinis dari seorang subjek, seperti jenis kelamin, usia,kebiasaan merokok, dan parameter-parameter lainnya. Datapengujian disusun dalam bentuk DataFrame menggunakan pustaka pandas di Python.

```

# testing pertama

new_data = pd.DataFrame({
    'male': [1],
    'age': [20],
    'education': [3],
    'currentSmoker': [1],
    'cigsPerDay': [10],
    'BPMed': [0],
    'prevalentStroke': [0],
    'prevalentHyp': [0],
    'diabetes': [0],
    'totChol': [60],
    'sysBP': [101],
    'diaBP': [76],
    'BMI': [25],
    'heartRate': [80],
    'glucose': [90]
})

new_data_scaled = scaler.transform(new_data)
prediction = clf.predict(new_data_scaled)
print("Prediksi TenYearCHD : ", prediction[0])

Prediksi TenYearCHD : 0

```

Gambar 12. Contoh Data untuk Testing

Data pengujian Gambar 12 kemudian diinputkan ke dalam model Decision Tree untuk mendapatkan prediksi risiko penyakit jantung pada 10 tahun mendatang. Hasil prediksi dapat digunakan untuk mengevaluasi sejauh mana model dapat mengklasifikasikan data baru berdasarkan pembelajaran dari dataset pelatihan.

Hasil prediksi pada Gambar 12 dari model Decision Tree menunjukkan bahwa subjek yang diuji memiliki prediksisisiko rendah terkena penyakit jantung pada periode 10 tahun mendatang. Dengan nilai prediksi 0, model menyatakan bahwa berdasarkan atribut-atribut klinis yang dimasukkan, subjek cenderung memiliki faktor-faktor yang mendukung kesehatan jantung yang baik. Interpretasi ini penting untuk membantu memahami implikasi prediksi model dalam konteks kesehatan masyarakat dan dapat digunakan dalam rekomendasi tindak lanjut yang sesuai.

5. Kesimpulan

Dalam penelitian ini, model Decision Tree digunakan untuk memprediksi risiko penyakit jantung berdasarkan dataset Framingham. Model ini mencapai akurasi 74% pada data pengujian, dengan evaluasi lebih lanjut menggunakan precision, recall, dan f1-score. Sebelumnya, dilakukan tahap data preprocessing, analisis eksploratori data, dan pemisahan data. Visualisasi seperti histogram digunakan untuk menganalisis distribusi variabel numerik. Model diuji menggunakan data baru, memberikan prediksi risiko penyakit jantung pada subjek baru. Kesimpulannya, model Decision Tree dapat menjadi alat yang berguna dalam mendukung prediksi risiko penyakit jantung, meskipun evaluasi lebih lanjut mungkin diperlukan untuk kelas minoritas. Temuan ini memiliki implikasi penting dalam konteks kesehatan masyarakat dan memberikan wawasan kepada praktisi kesehatan dan peneliti di bidang ini.

Daftar Pustaka

- [1] M. A. Bianto, K. Kusriani, dan S. Sudarmawan, "Perancangan Sistem Klasifikasi Penyakit Jantung Menggunakan Naïve Bayes," *Creat. Inf. Technol. J.*, vol. 6, no. 1, hal. 75, 2020, doi: 10.24076/citec.2019v6i1.231.
- [2] A. Riani, Y. Susianto, dan N. Rahman, "Implementasi Data Mining Untuk Memprediksi Penyakit Jantung Menggunakan Metode Naive Bayes," *J. Innov. Inf. Technol. Appl.*, vol. 1, no. 01, hal. 25–34, 2019, doi: 10.35970/jinita.v1i01.64.
- [3] A. Riski, "Analisis Komparasi Algoritma Klasifikasi Data Mining Untuk Prediksi Penderita Penyakit Jantung," *J. Tek. Inform. Kaputama*, vol. 3, no. 1, hal. 22–28, 2019, [Daring]. Tersedia pada: <https://jurnal.kaputama.ac.id/index.php/JTIK/article/view/141/156>.
- [4] L. Qadrini, A. Sepperwali, dan A. Aina, "Decision Tree Dan Adaboost Pada Klasifikasi Penerima Program Bantuan Sosial," *J. Inov. Penelit.*, vol. 2, no. 7, hal. 1959–1966, 2021.
- [5] I. Sutoyo, "Implementasi Algoritma Decision Tree Untuk Klasifikasi Data Peserta Didik," *J. Pilar Nusa Mandiri*, vol. 14, no. 2, hal. 217, 2018, doi: 10.33480/pilar.v14i2.926.
- [6] I Komang Setia Buana, "Implementasi Aplikasi Speech to Text untuk Memudahkan Wartawan Mencatat Wawancara dengan Python," *J. Sist. dan Inform.*, vol. 14, no. 2, hal. 135–142, 2020, doi: 10.30864/jsi.v14i2.293.
- [7] S. Ratna, "Pengolahan Citra Digital Dan Histogram Dengan Phyton Dan Text Editor Phycharm,"

- Technol. J. Ilm.*, vol. 11, no. 3, hal. 181, 2020, doi: 10.31602/tji.v11i3.3294.
- [8] S. Rahman *et al.*, *Python : Dasar Dan Pemrograman Berorientasi Objek*. 2023.
- [9] W. N. Santosa dan B. Baharuddin, "Penyakit Jantung Koroner dan Antioksidan," *KELUWIH J. Kesehatan. dan Kedokt.*, vol. 1, no. 2, hal. 98–103, 2020, doi: 10.24123/kesdok.v1i2.2566.
- [10] M. 123Studi Dona, Hendri Maradona, "SISTEM PAKAR DIAGNOSA PENYAKIT JANTUNG DENGAN METODE CASE BASED REASONING (CBR)," vol. 3, no. 1, hal. 1–12, 2021.
- [11] A. H. Nasrullah, "Implementasi Algoritma Decision Tree Untuk Klasifikasi Data Peserta Didik," *J. Pilar Nusa Mandiri*, vol. 7, no. 2, hal. 217, 2021.
- [12] A. Shiddiq, R. K. Niswatin, dan I. N. Farida, "Ahmad Shiddiq Analisa Kepuasan Konsumen Menggunakan Klasifikasi Decision Tree Di Restoran Dapur Solo (Cabang Kediri)," *Gener. J.*, vol. 2, no. 1, hal. 9, 2018, doi: 10.29407/gj.v2i1.12051.
- [13] A. N. Syahrudin dan T. Kurniawan, "Jurnal Dasar Pemograman Python STMIK," *J. Dasar Pemrograman Python STMIK*, hal. 1–7, 2018.
- [14] M. Ayub, "Proses Data Mining dalam Sistem Pembelajaran Berbantuan Komputer," hal. 21–30.
- [15] R. Ordila, R. Wahyuni, Y. Irawan, dan M. Yulia Sari, "PENERAPAN DATA MINING UNTUK PENGELOMPOKAN DATA REKAM MEDIS PASIEN BERDASARKAN JENIS PENYAKIT DENGAN ALGORITMA CLUSTERING (Studi Kasus : Poli Klinik PT.Inecda)," *J. Ilmu Komput.*, vol. 9, no. 2, hal. 148–153, 2020, doi: 10.33060/jik/2020/vol9.iss2.181.
- [16] Dedi Darwis, Nery Siskawati, dan Zaenal Abidin, "Penerapan Algoritma Naive Bayes untuk Analisis Sentimen Review Data Twitter BMKG Nasional," *J. TEKNO KOMPAK*, vol. 15, no. 1, hal. 131–145, 2020.