Journal of System and Computer Engineering (JSCE) ISSN (online): 2723-1240
Vol. 4 No. 1 Januari 2023

Analisa Penyakit Jantung Menggunakan Algoritma *Naïve Bayes*

Mahdiawan Nurkholifah[1], Andri Nofiar.Am[2*], Fenty Kurnia Oktorina[3]
[1]STMIK Amik Riau
[2,3]Politeknik Kampar

[1]2210031802058@sar.ac.id,[2]andrinofiar90@gmail.com, [3]fenty@poltek-kampar.ac.id

Abstrak

Penyakit jantung ialah pembunuh utama di dunia, membunuh hampir 2 juta orang Amerika setiap tahun. Temuan survei *Sample Regristration System* (SRS) memperlihatkan penyakit jantung ialah pemicu kematian terbesar pada semua umur setelah *stroke* (12,9%). Metode yang diterapkan pada penelitian ini ialah algortima *Naïve Bayes*. Tujuan dari penelitian ini ialah guna menentukan setiap yang terkena penyakit jantung terkena penyakit *stroke*. Dari hasil penelitian didapat dengan *splitting data* menggunakan 80:20 mendapatkan tingkat akurasi prediksi sebesar 83% untuk kasus prediksi *heart disease*. Pada hasil uji coba mengunakan data uji label yang didapatkan yakni *no stroke*.

Kata Kunci: *Heart Disease, Naïve Bayes*

Abstract

Every year more than 2 million Americans die from heart disease which is the number one killer in the world. The results of the Sample Registration System (SRS) survey show that heart disease is the highest cause of death at all ages after stroke, which is 12.9%. The method used in this study uses the Naïve Bayes algorithm. The purpose of this study is to determine if anyone with heart disease has a stroke. From the research results obtained by splitting data using 80:20 to get a prediction accuracy rate of 83% for heart disease prediction cases. In the trial results using the label test data obtained, namely no stroke.

Keywords: *Heart Disease, Naïve Bayes*

1. Pendahuluan

Dunia kesehatan memiliki database yang sangat banyak untuk dijadikan bahan penelitian bagi dunia pendidikan. Namun dataset yang tersedia biasanya hanya digunakan sebagai arsip misalnya data hasil cek Lab atau hasil pemeriksaan pasien yang menderita gejala atau penyakit tertentu (Nawawi dkk., 2019). Jantung ialah organ peredaran darah yang penting. Jantung memompa darah ke seluruh tubuh. Jika jantung rusak atau terganggu, seluruh organ lainnya akan terpengaruh. Pada tahun 2014, *survey Sample Registration System* (SRS) di Indonesia menemukan bahwa "penyakit jantung ialah pemicu kematian nomor dua setelah stroke (12,9%)" (Samosir dkk., 2021).

Penyakit jantung ialah pembunuh utama di dunia, membunuh hampir 2 juta orang Amerika setiap tahun (Widiastuti dkk., 2014). Penyakit jantung ialah serangkaian masalah yang menyerang jantung. Ini termasuk detak jantung tidak teratur, otot jantung lemah, kelainan jantung bawaan, penyakit pembuluh darah jantung, serta penyakit arteri koroner. Penyakit jantung mempunyai tingkat kematian yang tinggi di dunia. Bidang ilmu kedokteran sangat bergantung pada sarana otomatis berbasis komputer untuk diagnosis yang tepat, akurat, serta tepat waktu. Ini mengarah pada pemeliharaan data pasien setiap hari. Metode penambangan data diterapkan guna mengekstraksi informasi serta memprediksi penyakit di masa depan dari data yang disimpan (Derisma, 2020).

Data mining meningkat pesat seiring dengan teknologi informasi karena kebutuhan akan nilai tambah dari *database* berskala besar. Virgiawan dan Mukhlash (2013) mengemukakan *data mining* ialah proses mengekstraksi pola yang menarik (implisit, belum ditemukan, serta berpotensi untuk dimanfaatkan) dari data berukuran besar. *Data mining* diklasifikasikan menjadi *Decision Tree*, *Naïve Bayes*, *Support Vector Machine* (SVM), serta lainnya (Sabransyah dkk., 2017).

Data Mining ialah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan di dalam *database*. *Data mining* merupakan proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terikat dari berbagai *database* besar (Utomo & Mesran, 2020). Penambangan data menyediakan seperangkat alat dan teknik yang dapat diterapkan pada data yang diproses ini untuk menemukan pola tersembunyi dan juga memberikan sumber pengetahuan tambahan kepada profesional kesehatan untuk membuat keputusan yang lebih akurat (Fiqriansyah dkk., 2022)

Keterbatasan metode *manual* mendorong peneliti guna merancang metode yang tidak bergantung seutuhnya pada manusia. Metode yang dibuat menganalisis media sosial melalui komputer. Komputer hanya memahami bahasa mesin, sementara konten media sosial menerapkan bahasa yang difahami manusia. Mengkuantifikasi data yang mampu memecahkan kesulitan ini. Data kuantitatif dikelompokkan berlandaskan pendekatan *machine learning* (Laksana Utama, 2018).

2. Metode Penelitian

Dataset pada penelitian ini adalah kumpulan data atau informasi pasien penyakit heart desease yang didapat dari situs *Kaggle Open Datasets* (kaggle.com) sebanyak 303 data, terdiri dari 165 data target 1 dan 138 data target 0, target 0 merupakan non-stroke dan target 1 merupakan stroke. Kumpulan data ini memiliki 14 atribut, yaitu: *Age, Sex, CP, Trestbps, Chol, Fbs, Restecg, Thalach, Exang, Oldpeak, Slope, Ca, Thal, Target.* Lebih jelasnya diperlihatkan pada tabel 2.1 berikut:

Tabel 1 Dataset heart disease

age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
63	1	3	145	233	1	0	150	0	2	0	0	0	1
37	1	2	130	250	0	1	187	0	4	0	0	1	1
41	0	1	130	204	0	0	172	0	1	2	0	1	1
56	1	1	120	236	0	1	178	0	1	2	0	1	1

57	0	0	120	354	0	1	163	1	1	2	0	1	1
57	1	0	140	192	0	1	148	0	0	1	0	0	1
56	0	1	140	294	0	0	153	0	1	1	0	1	1
44	1	1	120	263	0	1	173	0	0	2	0	2	1
52	1	2	172	199	1	1	162	0	1	2	0	2	1
57	1	2	150	168	0	1	174	0	2	2	0	1	1
54	1	0	140	239	0	1	160	0	1	2	0	1	1
48	0	2	130	275	0	1	139	0	0	2	0	1	1
49	1	1	130	266	0	1	171	0	1	2	0	1	1
64	1	3	110	211	0	0	144	1	2	1	0	1	1

Sumber: https://www.kaggle.com/datasets/zgeakyldz/heart-desease-data

Keterangan Kolom :

- “age : usia dalam tahun.
- sex : jenis kelamin.
- cp : Nyeri dada yang dialami (Nilai 1: angina tipikal, Nilai 2: angina atipikal, Nilai 3: nyeri non angina, Nilai 4: asimtomatik)
- trestbps: Tekanan darah seseorang (mm Hg saat masuk ke rumah sakit).
- chol: Pengukuran kolesterol dalam mg/dl.
- fbs : Gula darah (> 120 mg/dl, 1 = benar; 0 = salah).
- restecg: Pengukuran elektrokardiografi (0 = normal, 1 = memiliki kelainan gelombang ST-T, 2 = menunjukkan kemungkinan atau pasti hipertrofi ventrikel kiri menurut kriteria Estes).
- thalach: Detak jantung maksimum seseorang tercapai.
- exang: Latihan menginduksi angina (1 = ya; 0 = tidak).
- oldpeak: Depresi ST yang diinduksi oleh olahraga relatif terhadap istirahat ('ST' berkaitan dengan posisi pada plot EKG.
- slope : kemiringan segmen ST latihan puncak (Nilai 1 : miring ke atas, Nilai 2 : datar, Nilai 3 : miring ke bawah).
- ca: jumlah major vessels (0-3)
- thal: Kelainan darah yang disebut thalassemia (3 = normal; 6 = cacat tetap; 7 = cacat reversibel).
- target: heart disease (0 = non stroke, 1 = stroke)”

Algoritma yang diterapkan pada riset ini yakni *Naïve Bayes*, ialah pengklasifikasi probabilistik dasar yang memperkirakan probabilitas dengan menjumlahkan frekuensi serta kombinasi nilai dari kumpulan data tertentu. Teknik ini menerapkan Teorema Bayes serta menganggap semua atribut ialah independen atau tidak saling ketergantungan yang diberikan oleh nilai pada variabel kelas. *Naïve Bayes* juga didefinisikan kategorisasi yang menerapkan metode probabilitas serta statistik Thomas Bayes guna memprediksi peluang masa depan berlandaskan pengalaman masa lalu (Putri dkk., 2021).
Pada penelitian (Handoko & Neneng, 2021) Persamaan *Naïve Bayes* adalah:

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)}$$

Keterangan :

- "P(Ci|X) = Probabilitas hipotesis Ci jika diberikan fakta atau record X (Posterior probability).
- P(X|Ci) = mencari nilai parameter yang memberi kemungkinan yang paling besar (likelihood).
- P(Ci) = Prior probability dari X (Prior probability) P(X) = Jumlah probability tuple yg muncul".

Tahapan algoritma *Naïve Bayes*:

1. Menghitung jumlah kelas / label P(H)
2. Menghitung jumlah kasus per kelas P(X|H)
3. Kalikan semua *variable* kelas P(X|H) * P(H)
4. Bandingkan hasil perkelas.

3. Hasil dan Pembahasan

3.1 Tahapan algoritma *Naïve Bayes*

Sebelum melakukan perhitungan pada *dataset* yang ada, maka dilakukan pembersihan data dalam dataset. Data yang dilakukan pembersihan yakni data berupa NaN yang mana nilai itu ialah nilai ca = 4 dan nilai th = 0.
Dengan formula untuk melakukan pembersihan data ialah :

- data ca < 0
- data th = 0

Setelah dilakukan pembersihan data maka sisa data yang didapat yakni 296 data.Untuk lebih jelasnya bisa menggunakan data uji dibawah ini untuk perhitungan *manual*:

- age : 43
- sex : 1
- cp : 3
- trestbps: 200
- chol: 233
- fbs : 1
- restecg: 0
- thalach: 103
- exang: 1
- oldpeak: 0,9
- slope : 1
- ca: 2
- thal: 2
- target: ?

1. Tahap Pertama : menghitung jumlah kelas / label P(H)

Jumlah record pada dataset = 296
Formula mencari prior kelas adalah *Prior* = jumlah kelas/total data

Tabel 2 jumlah kelas *pertarget* dan *prior*

Kelas	jumlah	prior
non stroke	136	0.506
stroke	160	0.595

2. Tahap Kedua : menghitung jumlah kasus perkelas P(X|H) dan kalikan semua *variable* kelas P(X|H) * P(H)
formula untuk mendapatkan nilai *prior* adalah *prior* kasus = jumlah kasus kelas/jumlah target kelas

Tabel 3 perhitungan semua *variable* kelas P(X|H) * P(H)

kelas P(X|H)	Jumlah Kelas	P(Non Stroke)	P(Stroke)
P(age 43 | non stroke)	3	0.022	
P(age 43 | stroke)	5		0.031
P(sex 1 | non stroke)	114	0.838	
P(sex 1 | stroke)	93		0.581
P(cp 1 | non stroke)	7	0.051	
P(cp 1 | stroke)	16		0.100
P(trestbps 200 | non stroke)	1	0.007	
P(trestbps 200 | stroke)	0		0.000
P(chol 233 | non stroke)	1	0.007	
P(chol 23 | stroke)	3		0.019
P(fbs 1 | non stroke)	22	0.162	
P(fbs 1 | stroke)	23		0.144
P(restecg 0 | non stroke)	79	0.581	
P(restecg 0 | stroke)	68		0.425
P(thalach 103 | non stroke)	2	0.015	
P(thalach 103 | stroke)	0		0.000
P(exang 1 | non stroke)	76	0.559	
P(exang 1 | stroke)	23		0.144
P(oldpeak 0.9 | non stroke)	2	0.015	
P(oldpeak 0.9 | stroke)	1		0.006
P(slope 1 | non stroke)	91	0.669	
P(slope 1 | stroke)	49		0.306
P(ca 2 | non stroke)	31	0.228	
P(ca 2 | stroke)	7		0.044
P(thal 2 | non stroke)	36	0.265	
P(thal 2 | stroke)	130		0.813
P(X|no stroke)		0.00000000000002359	
P(X|stroke)			0

Berdasarkan data dari tabel diatas didapatkan nilai P(X|Stroke) = 0 sedangkan P(X|Non Stroke) = 0.00000000000002359

3. Tahap Ketiga : membandingkan hasil per kelas

Tabel 3 Perbandingan hasil perkelas

Target Stroke	Target Non Stroke
P(X| Stroke) * P(Stroke)	P(X| Non Stroke) * P(Non Stroke)
= 0 * (160/269)	=0.00000000000002359*(136/269)
= 0	=0.00000000000001193

Berdasarkan data dari tabel perbandingan di atas maka hasil prediksi dari data uji adalah *Non Stroke*, karena hasil target *Non Stroke* lebih besar dari hasil *Stroke*.

3.2 Klasifikasi perhitungan *code python*

Dataset yang sebelumnya berformat xlsx, disimpan dalam bentuk csv. Agar *dataset* benar-benar dipisahkan menggunakan koma (,) maka terlebih dahulu data yang tersimpan dipastikan tidak menggunakan pemisah koma namun menggunakan titik untuk data *float*. Sedangkan untuk data Ribuan tidak menggunakan pemisah. Setelah semua data telah sesuai dengan kebutuhan, selanjutnya data sets secara *programmatic* siap di analisis menggunakan bahasa python.

Import liblary

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.pipeline import Pipeline
from sklearn.compose import ColumnTransformer
```

Gambar 1 *Import liblary*

Import data csv

```
data = pd.read_csv('heart_desease_test.csv')
data.head()
```

Gambar 2 *Import data csv*

Setelah dilakukan tahap *import* untuk memastikan bahwa dokumen telah ter-*import* dengan tepat dapat mengecek 5 data teratas data set dapat dilihat pada gambar

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1

Gambar 3 menampilkan 5 data teratas

Melihat jumlah baris dan kolom pada *dataset*

```
print("Heart data shape is:", data.shape[0], "x", data.shape[1])
```

```
Heart data shape is: 303 x 14
```

Gambar 4 menampilkan jumlah baris dan kolom

Cek *dataset* apabila ada nilai null

```
data.isnull().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 menampilkan hasil data null

Tahapan *preprocessing* data

Pada *dataset* data #93, 159, 164, 165 dan 252 memiliki ca=4 yang salah. Dalam *dataset* asli, mereka adalah NaN. data #49 dan 282 memiliki th = 0, juga salah. Mereka juga merupakan NaN dalam kumpulan data asli. Melakukan penghapusan data yang dianggap NaN dalam data asli. Serta menampilkan jumlah data setelah dilakukan pembersihan data

```
data = data[data['ca'] < 4] #drop the wrong ca values
data = data[data['thal'] > 0] # drop the wong thal value
print("Heart data shape is:", data.shape[0], "x", data.shape[1])
```

```
Heart data shape is: 296 x 14
```

Gambar 6 menampilkan jumlah data baru.

Menghitung jumlah label target *stroke* dan *non stroke*

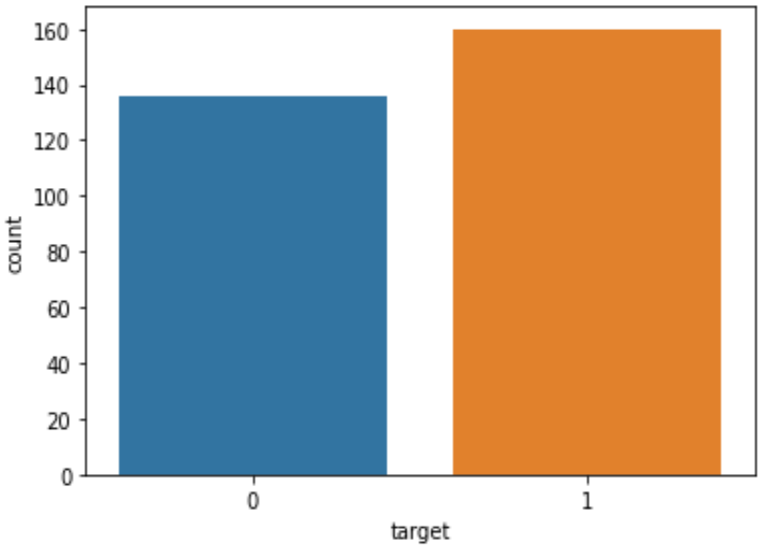
```
total = len(data["target"])
stroke = data["target"].sum()
non_stroke = len(data["target"]) - stroke
print("Total label no stroke:", non_stroke)
print("Total label stroke:", stroke)
```

```
Total label no stroke: 136
Total label stroke: 160
```

Gambar 7 jumlah label stroke dan no stroke

Menampilkan jumlah label stroke dan no stroke dengan bar chart

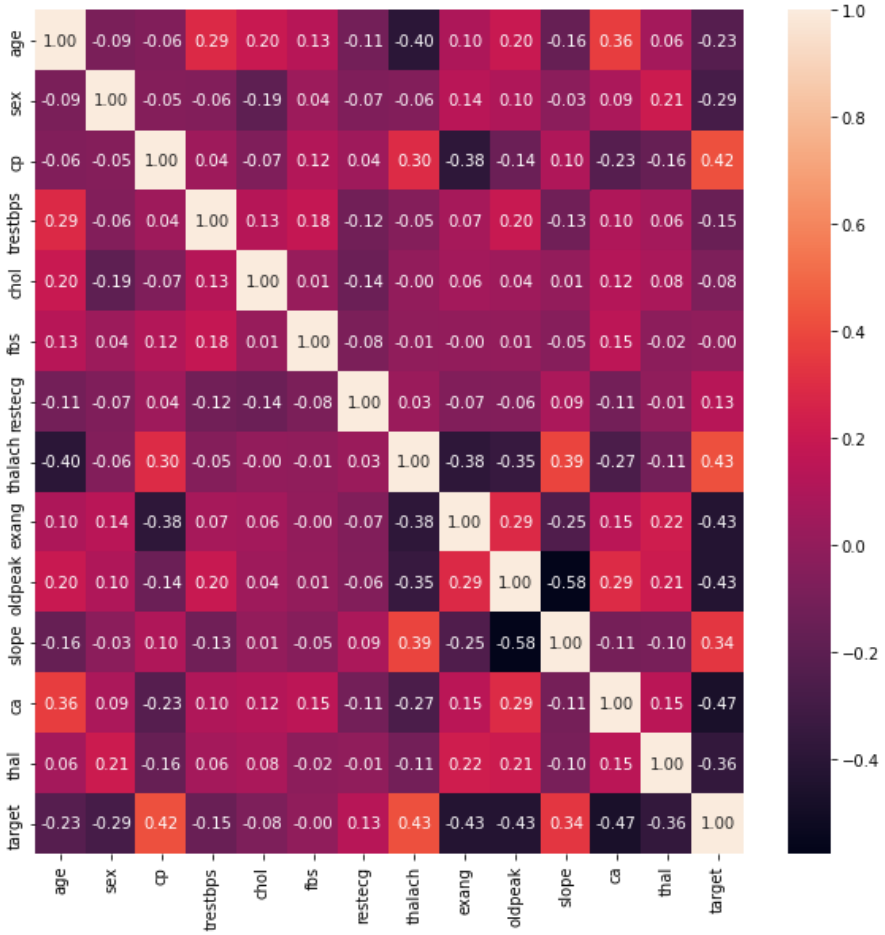
```
sns.countplot(data["target"])
```



Gambar 8 *chart* jumlah *stroke* dan *no stroke*

Melihat *korelasi* antar *variabel*

```
plt.figure(figsize=(10, 10))
sns.heatmap(data.corr(), annot=True, fmt='.2f')
```



Gambar 9 *plot korelasi* antar *variabel*

Melakukan *splitting data* , dengan nilai *seed* nya 0 dan nilai *test size* nya 20% dan juga *size* 30%

```
from sklearn.naive_bayes import GaussianNB
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, plot_confusion_matrix, classification_re
port
from sklearn.metrics import recall_score, accuracy_score,roc_curve, auc

seed = 0
test_size = 0.2// ubah ke 0.3 untuk ke 70:30

X = data.drop(["target"], axis=1)
y = data["target"]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = test_size, random_stat
e=seed)
```

Menginisialisaikan algoritma klasifikasi *naive bayes* serta melakukan pelatihan model.

```
clf = GaussianNB()
clf.fit(X_train, y_train)
```

Melihat nilai *score*

```
clf.score(X_train,y_train)
```

```
0.847457627118644
```

Gambar 10 hasil *score* yang diperoleh

Melihat hasil prediksi dari dataset yang diolah

```
pred = clf.predict(X_test)
accuracy = accuracy_score(y_test, pred)
pred_proba = clf.predict_proba(X_test)[:, 1]
fpr, tpr, thresholds = roc_curve(y_test, pred_proba)
roc_auc = auc(fpr, tpr)
print(classification_report(y_test,pred))
```

```
              precision    recall  f1-score   support

           0       0.89      0.78      0.83        32
           1       0.78      0.89      0.83        28

    accuracy                           0.83        60
   macro avg       0.84      0.84      0.83        60
weighted avg       0.84      0.83      0.83        60
```

Gambar 11 tampilan *report clasification* 80:20

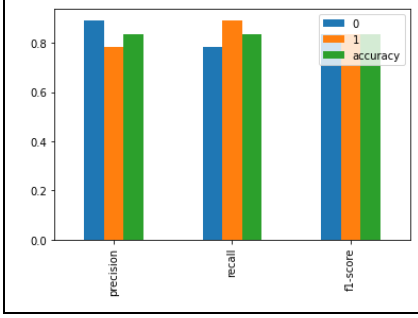
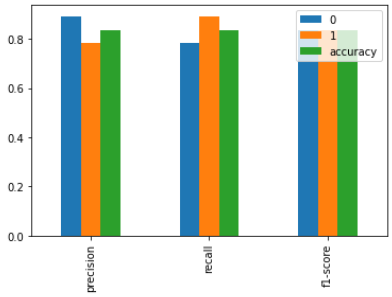
	precision	recall	f1-score	support
0	0.87	0.76	0.81	45
1	0.78	0.89	0.83	44
accuracy			0.82	89
macro avg	0.83	0.82	0.82	89
weighted avg	0.83	0.82	0.82	89

Gambar 12 tampilan *report clasification* 70:30

Visualisasi clasifiacation report dengan *bar chart*

```
report = classification_report(y_test,pred, output_dict=True)
reporttb = pd.DataFrame(report).transpose()
reporttb.drop('support', inplace=True, axis=1)
reporttb.iloc[:3, :10].T.plot(kind='bar')
plt.show()
```

Tabel 4 Perbandingan hasil *test size*

Test size 80:20	Test size 70:30
	

Melakukan proses pengujian menggunakan data uji

```
X_pred = pd.read_csv("data_uji_heart_disease.csv")
X_pred
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal
0	43	1	3	200	233	0	0	103	1	0.9	1	2	2

Gambar 13 tampilan data uji

```
X_pred["target"] = clf.predict(X_pred)
X_pred
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	43	1	3	200	233	0	0	103	1	0.9	1	2	2	0

Gambar 14 tampilan hasil data uji

Tabel 5 hasil perbandingan setiap variable pada 0 dan 1

Spliting data	Accuracy	precision		recall		f1-score	
		0	1	0	1	0	1
80 : 20	83%	89%	78%	78%	89%	83%	83%
70 : 30	82%	87%	78%	76%	89%	81%	83%

4. Kesimpulan

Berlandaskan temuan riset serta pembahasan di atas, maka kesimpulan penelitian dengan menggunakan algoritma *Naïve Bayes* bahwa dengan *splitting data* menggunakan *test size* 80:20 mendapatkan tingkat akurasi prediksi sebesar 83% sedangkan *test size* 70:30 mendapatkan tingkat akurasi prediksi sebesar 82% untuk kasus prediksi *heart disease*. Maka *test size* 80:20 mendapatkan hasil *accuracy* yang lebih baik dari pada *test size* 70:30. Dan dapat dijadikan acuan *accuracy test size* untuk perbandingan algoritma *machine learning* lainnya.

Daftar Pustaka

Derisma. (2020). Perbandingan Kinerja Algoritma untuk Prediksi Penyakit Jantung dengan Teknik Data Mining. *Journal of Applied Informatics and Computing*, *4*(1), 84–88. https://doi.org/10.30871/jaic.v4i1.2152

Fiqriansyah, R., Akbar, F., Andiko, V. C., Ahmad, K. G., Rasywir, E., Meisak, D., Pratama, Y., & Feranika, A. (2022). *Penerapan Algoritma Naïve Bayes Untuk Mengetahui Pasien Penyakit Gagal Jantung Jurnal Informatika Dan Rekayasa Komputer (JAKAKOM)*. *2*(September).

Handoko, M. R., & Neneng. (2021). Sistem Pakar Diagnosa Penyakit Ispa Menggunakan Metode Naive Bayes Classifier Berbasis Web. *CSRID (Computer Science Research and Its Development Journal)*, *10*(3), 127. https://doi.org/10.22303/csrid.10.3.2018.127-138

Laksana Utama, P. K. (2018). Identifikasi Hoax pada Media Sosial dengan Pendekatan Machine Learning. *Widya Duta: Jurnal Ilmiah Ilmu Agama dan Ilmu Sosial Budaya*, *13*(1), 69. https://doi.org/10.25078/wd.v13i1.436

Nawawi, H. M., Purnama, J. J., & Hikmah, A. B. (2019). Komparasi Algoritma Neural Network Dan Naïve Bayes Untuk Memprediksi Penyakit Jantung. *Jurnal Pilar Nusa Mandiri*, *15*(2), 189–194. https://doi.org/10.33480/pilar.v15i2.669

Putri, H., Purnamasari, A. I., Dikananda, A. R., Nurdiawan, O., & Anwar, S. (2021). Penerima Manfaat Bantuan Non Tunai Kartu Keluarga Sejahtera Menggunakan Metode NAÏVE BAYES dan KNN. *Building of Informatics, Technology and Science (BITS)*, *3*(3), 331–337. https://doi.org/10.47065/bits.v3i3.1093

Sabransyah, M., Nasution, Y. N., & Tisna, D. (2017). Aplikasi Metode Naive Bayes dalam Prediksi Risiko Penyakit Jantung Naive Bayes Method for a Heart Risk Disease Prediction Application. *Jurnal EKSPONENSIAL*, *8*, 111–118.

Samosir, A., Hasibuan, M. S., Justino, W. E., & Hariyono, T. (2021). Komparasi Algoritma Random Forest, Naïve Bayes dan K- Nearest Neighbor Dalam klasifikasi Data Penyakit Jantung. *Prosiding Seminar Nasional Darmajaya*, *1*(0), 214–222. https://jurnal.darmajaya.ac.id/index.php/PSND/article/view/2955

Utomo, D. P., & Mesran, M. (2020). Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung. *Jurnal Media Informatika Budidarma*, *4*(2), 437. https://doi.org/10.30865/mib.v4i2.2080

Widiastuti, N. A., Santosa, S., & Supriyanto, C. (2014). Algoritma Klasifikasi Data Mining Naïve Bayes Berbasis Particle Swarm Optimization Untuk Deteksi Penyakit Jantung. *Nature Methods*, *7*(1), 11. https://doi.org/10.1038/nmeth.f.284