

# Analisa Perbandingan Metode Klasifikasi Machine Learning Untuk Mendeteksi Serangan Jantung

Joko Siswantoro<sup>1\*</sup>, Kenneth Manuel Lieyanto<sup>2</sup>, Venansius Mario Tando<sup>3</sup>

<sup>1\*</sup>Program Studi Teknik Informatika, Universitas Surabaya, Surabaya, Jawa Timur

<sup>2</sup>Program Studi Teknik Informatika, Universitas Surabaya, Surabaya, Jawa Timur

<sup>3</sup>Program Studi Teknik Informatika, Universitas Surabaya, Surabaya, Jawa Timur

Email: <sup>1\*</sup> joko\_siswantoro@staff.ubaya.ac.id, <sup>2</sup>s160419041@student.ubaya.ac.id,  
<sup>3</sup>s160419091@student.ubaya.ac.id

(Naskah masuk: dd mmm yyyy, direvisi: dd mmm yyyy, diterima: dd mmm yyyy)

## Abstrak

Pekembangan terkait ilmu pengetahuan dan teknologi terus meningkat. Kemajuan terkait teknologi membuat manusia untuk menikmati kemudahan. Tetapi dalam segala kemudahan tersebut, mendeteksi serangan jantung tidak termasuk dalam kemudahan ini. Serangan jantung merupakan penyebab kematian paling tinggi di dunia. Sekitar 15.2 juta orang mati karena serangan jantung di tahun 2019, merepresentasikan 27% dari total kematian. Angka kematian karena serangan jantung diperkirakan akan terus meningkat karena kurangnya pengetahuan atau informasi terkait penyakit ini. Oleh karena itu diperlukanlah sebuah metode untuk mendeteksi serangan jantung lebih dini.

**Kata Kunci:** klasifikasi, penyakit jantung, *machine learning*

## *Comparative Analysis of Classification Methods for Detecting Heart Attack*

### Abstract

*The continues development of science and technology is always advancing. Advancement related to techonology makes human enjoy convenience. But in all these convenience, detecting heart attack is not included in one of this convenience. Heart attacks are the leading cause of death globally. An estimated of 15.2 million people died from heart attack in 2019, representing 27% of global death. The number of death caused by heart diseases are expected to increase due to lack of knowledge or information related to this disease. Therefore, a method is needed to detect heart attacks early.*

**Keywords:** *classification, heart disease, machine learning*

---

## I. PENDAHULUAN

Serangan jantung merupakan salah satu penyakit yang paling paling mematikan di dunia[1]. Berdasarkan data dari WHO (*World Health Organization*) Sekitar 17.9 juta orang di dunia meninggal karena penyakit kardiovaskular pada tahun 2019. Angka ini merepresentasikan dari 32% dari penyebab kematian secara global dimana 85% dari kematian tersebut merupakan penyakit jantung, artinya sekitar 15.2 juta orang meninggal karena serangan jantung [1]. Ini

artinya serangan jantung sendiri merepresentasikan 27% dari penyebab kematian global. Di Indonesia sendiri pada tahun 2013 penderita penyakit jantung sebesar 61.682 [2]. Meski terdapat banyak cara seperti operasi, penyinaran, dna khemoterapi tetapi cara ini disertai isu-isu tertentu mulai dari masalah secara ekonomi, kualitas, keselamatan pasien dan kelalaian manusia. Maka dari itu diperlukanlah cara yang meningkatkan produktivitas dan mempercepat proses penanganan terkait penyakit jantung yaitu dengan memanfaatkan metode Machine Learning.

Ari, et.al. melakukan perancangan sistem klasifikasi penyakit jantung menggunakan algoritma klasifikasi Naïve Bayes dengan 5-fold cross validation. Akurasi yang dihasilkan 95% [2].

Srabanti dan Srishti menggunakan dua metode C4.5 dan ANN untuk melakukan prediksi terhadap *dataset heart disease* dan membuat model hybrid yang menggunakan kombinasi C4.5 dan ANN menemukan *accuracy* yang dipakai lebih dari masing-masing metode yaitu 78% [3].

diperlukan untuk melakukan percobaan-percobaan pada dataset ini [3].

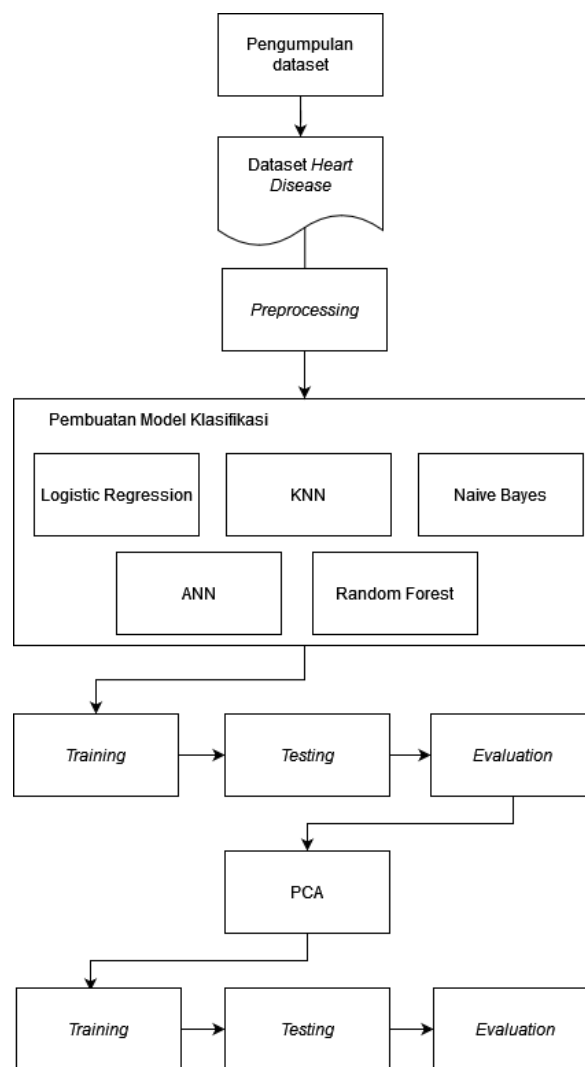
Anand, et.al membuat sistem klasifikasi dengan berbagai algoritma Naïve Bayes, K-Nearest Neighbors, SVM, Random Forest, dan XGBoost, Cat Boost, Light GBM, dan Ada Boost menunjukan bahwa Random Forest menghasilkan model terbaik dengan akurasi 97% tetapi tidak melakukan perbandingan hasil terhadap evaluasi model lainnya [4].

Nabila dan Hamid melakukan komparasi terhadap banyak penelitian (2012 – 2020) yang melakukan klasifikasi menggunakan *Cleveland heart disease dataset* dengan menggunakan berbagai algoritma klasifikasi menyimpulkan bahwa teknik data mining yang sama memberikan hasil yang berbeda dan masih

Berdasarkan penelitian yang sudah dilakukan sebelumnya, meski banyak yang berhasil membuat sebuah model prediksi yang cukup baik, tetapi belum terdapat penelitian yang bertujuan untuk membandingkan performa algoritma klasifikasi pada dataset *Cleveland heart disease* dengan menggunakan PCA (*Principal Component Analysis*). Penelitian ini bertujuan untuk membandingkan performa algoritma klasifikasi beserta performa setelah ketika menggunakan PCA terhadap dataset *Cleveland heart disease* apakah pasien memiliki penyakit jantung atau tidak. Algoritma klasifikasi yang akan digunakan dalam penelitian adalah Logistic Regression, KNN, Random Forest, ANN, Decision Tree, Naïve Bayes, dan XGBoost. Performa tiap-tiap algoritma yang akan dibandingkan adalah akurasi, precision, recall, dan f1 score. Performa algoritma diharapkan dapat berguna untuk peneliti-peneliti lain dalam menentukan algoritma yang digunakan dalam melakukan klasifikasi *heart disease*.

## II. METODOLOGI PENELITIAN

Metodologi penelitian menggunakan tahapan, dimulai dari pengumpulan dataset, melakukan data *preprocessing*, pembentukan model klasifikasi (Logistic Regression, KNN, Naïve Bayes, ANN, dan Random Forest), training tiap-tiap model klasifikasi, testing tiap-tiap model klasifikasi, kemudian melakukan perbandingan model evaluasi tiap-tiap model, setelah tahapan-tahapan tersebut, dalam penelitian juga akan membandingkan hasil perhitungan evaluasi setelah menggunakan PCA (*Principal Component Analysis*). Gambar 1 menunjukan diagram yang menunjukan tahapan metodologi yang akan dilakukan pada penelitian ini.



Gambar 1. Metodologi Penelitian

### A. Pengumpulan Dataset

Dataset yang digunakan dalam penelitian ini adalah data pasien *Cleveland heart disease dataset* yang diperoleh dari *UCI Machine Learning Repository* [5]. Tujuan dari penggunaan dataset yang diambil dari repository *UCI Machine Learning* ini adalah sifatnya yang publik sehingga dapat dibandingkan dengan penelitian berikutnya. Dataset dikumpulkan dengan format CSV (*Comma Separated Value*) berisikan 303 *record* yang terdiri dari 14 fitur (13 variabel dependen dan 1 variabel independen) dimana 5 atribut bertipe data kontinu dan 8 atribut bertipe data kategorik. Tabel data atribut dari dataset *heart disease* dapat dilihat pada Tabel 1.

Tabel 1. Deskripsi Data

Nama Atribut	Tipe Data	Deskripsi
Age	Kontinu	Umur pasien dalam satuan tahun
Sex	Biner	Kelamin pasien (1: Laki-laki, 2: perempuan)
CP	Kategorik (0-3)	Jenis nyeri dada pasien (1: Typical Angina, 2: Atypical Angina, 3: Non-anginal Pain, 4: Aymptomatic)
Trestbps	Kontinu	Tekanan darah pasien (dalam mm/Hg)
Chol	Kontinu	Serum kolesterol
FBS	Biner	Kadar gula darah
Restecg	Kategorik (0-2)	Hasil pengukuran elektrokardiografi (0: normal, 1: kelainan gelombang ST-T, 2: menunjukkan kemungkinan hipertrofi ventrikel kiri)
Thalach	Kontinu	Detak jantung maksimum (bpm)
Exang	Biner	Keluhan pasien ketika berolahraga (1: iya, 2: tidak)
Oldpeak	Kontinu	Depresi ST akibat olahraga relative terhadap insirahat
Slope	Kategorik (0-2)	Kemiringan segmen pada ST puncak (0: upslope, 2: downslope)
CA	Kategorik (0-3)	Jumlah pembuluh dari utama yang diwarnai dengan fluoroskopi
Thal	Kategorik (1-3)	(1-3: normal, 6: fixed defect, 7: reversable defect)
Target	Biner	Sakit atau tidaknya pasien (0: Pasien tidak memiliki penyakit, 1: Pasien memiliki penyakit)

## B. Data Preprocessing

Data preprocessing yang dilakukan pada tahapan ini adalah *one hot encoding*, dan *data standardization*. Rumus *standarization* dapat dilihat pada persamaan (1).

$$z = \frac{x - \mu}{\sigma} \quad (1)$$

## C. Pembuatan Model Klasifikasi

### 1) Logistic Regression

Konfigurasi parameter yang digunakan pada algoritma logistic regression adalah *solver*.

Table 2. Parameter Logistic Regression

Parameter	Deskripsi
<i>Solver</i>	<i>lbfgs, newton-cg, liblinear, sag, saga</i>

### 2) KNN

Konfigurasi parameter yang digunakan pada algoritma KNN adalah jumlah *neighbors*, *power parameter* dan algoritma perhitungan *distance*.

Tabel 3. Parameter KNN

Parameter	Deskripsi
Jumlah <i>neighbors</i>	1 - 31
<i>Power parameter</i>	1, 2
<i>Distance metric</i>	<i>Euclidean, Manhattan, Chebyshev, Minkowski</i>

### 3) Naïve Bayes

Pada algoritma Naïve Bayes tidak terdapat banyak parameter yang dapat diubah sehingga tidak dilakukan hyperparameter tuning.

### 4) ANN

Konfigurasi parameter pada algoritma ANN adalah *activation function*, dan *solver*.

Tabel 4. Parameter ANN

Parameter	Deskripsi
<i>Activation function</i>	<i>Relu, tanh, logistic, indentity</i>
<i>Solver</i>	<i>Adam, SGD (Stochastic Gradient Descent)</i>

### 5) Random Forest

Konfigurasi parameter pada algoritma random forest adalah *criterion*.

Tabel 5. Parameter Random Forest

Parameter	Deskripsi
<i>Criterion</i>	<i>Gini, Entropy</i>

## D. Training

Pada training dan testing dataset yang digunakan dibagi menjadi dua bagian dengan proporsi 70% training dan 30% testing. Pembagian data training dan testing dilakukan secara acak.

Training dilakukan dengan menggunakan algoritma Logistic Regression, KNN, Naïve Bayes, ANN, dan

Random Forest dengan parameter yang dijelaskan pada subbab sebelumnya.

#### E. Testing

Tahapan testing dilakukan untuk melakukan validasi terhadap tiap model yang sudah dibuat pada tahapan training. Testing dilakukan untuk algoritma Logistic Regression, KNN, Naïve Bayes, ANN, dan Random Forest

#### F. Evaluasi Model

Pada tahapan evaluasi model, model performa dievaluasi dengan menggunakan perhitungan pada tahapan testing. *Metric* performa yang digunakan antara lain adalah *accuracy* dan *f1 score*. Tiap model kemudian akan dibandingkan satu sama lain dengan rata-rata performa.

Persamaan (2) menunjukkan perhitungan *accuracy*. *Accuracy* digunakan untuk menghitung total dari *True Positive* (TP) dan *True Negative* (TN) dibagi dengan total dari TP, TN, *False Positive* (FP), *False Positive* (FP), dan *False Negative* (FN).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

Persamaan (2) menunjukkan perhitungan dari *precision*. *Precision* dihitung dengan cara membagi TP dengan total dari TP dan FP

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

Persamaan (3) menunjukkan perhitungan dari *recall*. *Recall* dihitung dengan cara membagi TP dengan total dari TP dan FN.

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

Persamaan (4) menunjukkan perhitungan dari *f1 score*. *F1 score* dihitung dengan cara membagi antara perkalian dari hasil *precision* dengan *recall* dan penambahan *precision* dan *recall*.

$$F1\ Score = \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

Persamaan (5) menunjukkan perhitungan dari rata-rata performa dengan cara menggunakan rata-rata dari *accuracy* data *testing* dan *f1 score* data *testing*.

$$\mu_{performance} = \frac{Accuracy + f1\ score}{2} \quad (6)$$

#### G. PCA

PCA adalah sebuah teknik *unsupervised* yang menilai ketertarikan antar atribut. Tujuan utama pada PCA adalah *dimension reduction*. Pada tahapan ini atribut-atribut akan digantikan oleh sebuah PC dimana atribut-atribut berkorelasi yang tidak berkontribusi dalam pengambilan keputusan akan dihilangkan.

### III. HASIL DAN PEMBAHASAN

Pada tahapan ini akan dijelaskan mengenai hasil uji coba klasifikasi terhadap *dataset heart disease* menggunakan algoritma Logistic Regression, KNN, Naïve Bayes, ANN, dan Random Forest. Kemudian akan dijelaskan pula mengenai perbandingan dari tiap algoritma tersebut.

#### A. Hasil Uji Coba Algoritma Logistic Regression

Perbandingan solver pada algoritma Logistic Regression menunjukkan bahwa yang terbaik adalah *libliner* yang menghasilkan nilai  $\mu_{performance}$  sebesar 87.52%.

Tabel 6. Hasil Uji Coba Algoritma Logistic Regression

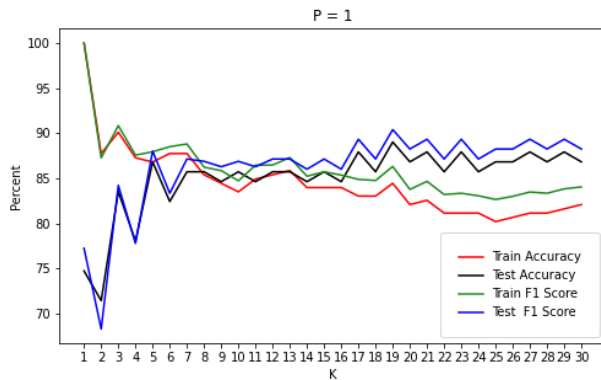
Solver	Akurasi		F1 Score	
	Train	Test	Train	Test
Liblinear	86.79	86.81	88.14	88.24
Newton-cg	86.79	85.71	88.14	88.38
Lbfgs	86.79	85.71	88.14	88.38
Sag	86.79	85.71	88.14	88.38
Saga	86.79	85.71	88.14	88.38

#### B. Hasil Uji Coba Algoritma KNN

Pada algoritma KNN parameter yang terbaik adalah nilai *n neighbors* dengan nilai 19 dengan *p* 1 yang menghasilkan nilai  $\mu_{performance}$  sebesar 89.69%.

Tabel 7. Hasil Uji Coba Algoritma KNN

Distance Metric	Best K	Best P	Akurasi		F1 Score	
			Train	Test	Train	Test
Euclidean	19	1	84.43	89.01	86.31	90.38
Manhattan	17	1	83.02	85.71	84.75	87.38
Chebyshev	27	1	70.28	85.71	74.29	76.64
Minkowski	19	2	84.43	89.01	86.31	90.38



Gambar 2 KNN dengan model terbaik

### C. Hasil Uji Coba Algoritma Naïve Bayes

Pada algoritma Naïve Bayes tidak terdapat parameter yang diubah. Algoritma Naïve Bayes menghasilkan  $\mu_{performance}$  sebesar 85.31%.

Tabel 8. Hasil Uji Coba Algoritma Naïve Bayes

Akurasi		F1 Score	
Train	Test	Train	Test
85.38	84.62	86.81	86.00

### D. Hasil Uji Coba Algoritma ANN

Pada algoritma ANN parameter yang terbaik adalah *activation function Identity* dengan *solver SGD* menghasilkan  $\mu_{performance}$  sebesar 89.69%.

Tabel 9. Hasil Uji Coba Algoritma ANN

Activation Function	Solver	Best HLZ	Akurasi		F1 Score	
			Train	Test	Train	Test
Relu	adam	4	92.45	87.91	93.10	89.32
	sgd	5	88.67	86.81	89.56	88.00
Tanh	adam	7	89.15	86.81	90.29	88.46
	sgd	2	87.73	86.81	88.88	88.23
Logistic	adam	2	88.20	86.81	89.45	88.23
	sgd	9	85.84	87.91	87.28	88.88
Identity	adam	2	88.20	85.71	89.45	87.61
	sgd	7	86.79	87.91	87.82	89.10

### E. Hasil Uji Coba Algoritma Random Forest

Pada algoritma Random Forest parameter yang terbaik adalah *criterion gini* menghasilkan  $\mu_{performance}$  sebesar 85.31%.

Tabel 10. Hasil Uji Coba Algoritma Random Forest

Criterion	Akurasi		F1 Score	
	Train	Test	Train	Test
Gini	100	84.62	100	86
Entropy	100	83.52	100	84.85

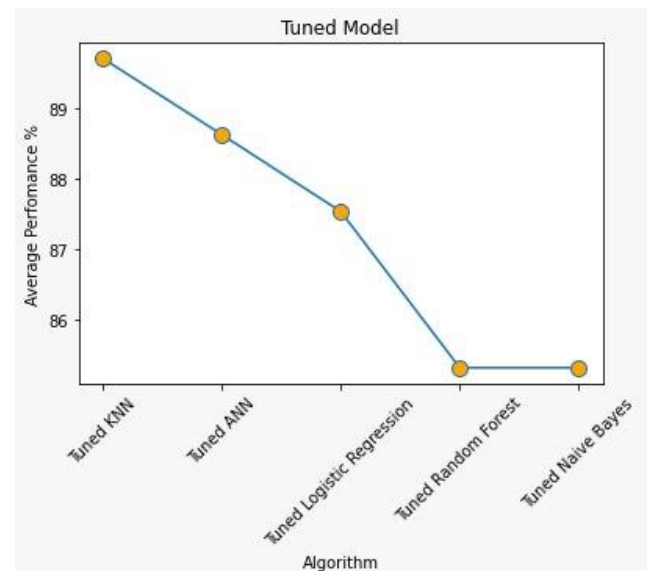
### F. Perbandingan Performa

Tabel 11. menunjukan hasil perbandingan *metric* performa dari algoritma KNN, ANN, Logistic Regression, Random Forest, dan Naïve Bayes. Untuk algoritma KNN yang menggunakan parameter telah mendapatkan performa terbaik. Untuk akurasi dari data test sendiri mencapai 89.01% dan memiliki *f1 score* untuk data test sebesar 90.38%.

Performa KNN terhadap ANN sendiri sebenarnya tidak signifikan, yaitu sebesar 1.18%.

Table 11. Hasil Klasifikasi Terbaik Tiap Model

Algoritma	Akurasi		F1 Score		$\mu$ Perform
	Train	Test	Train	Test	
KNN	84.43	89.01	86.31	90.38	89.7
Neural Network	92.45	87.91	93.10	89.32	88.62
Logistic Regression	86.79	86.81	88.14	88.24	87.52
Random Forest	100	84.62	100	86	85.31
Naïve Bayes	85.38	84.62	86.81	86	85.31



Gambar 3 Perbandingan average performance tiap model

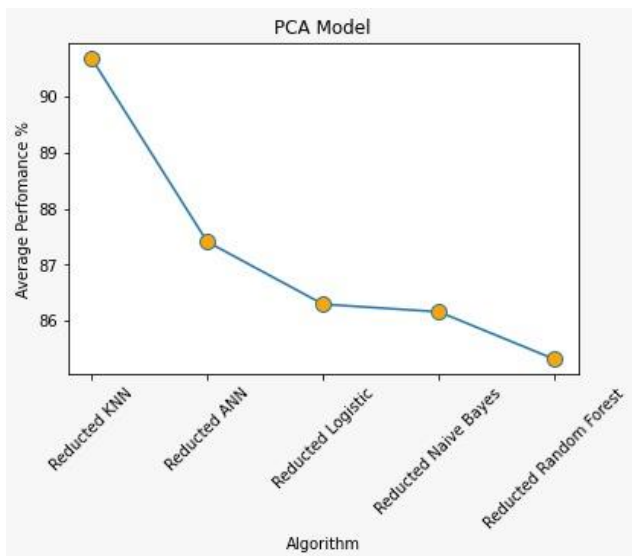
### G. Perbandingan Performa Setelah PCA

Tabel 12. menunjukan perbandingan performa dengan menggunakan PCA. Dengan menggunakan PCA ini, komputasi akan menjadi lebih cepat karena dimensi pada data training akan dikurangi tanpa mengurangi keaslian data tersebut. Dalam perbandingan di atas, algoritma KNN mendapat performa tertinggi yaitu 90.69% dengan pengurangan dimensi hingga 9 komponen. Dalam metode pengurangan dimensi ini tidak selalu memberikan akurasi yang lebih besar daripada tidak

mengurangi dimensi dengan analisis komponen utama. Seperti algoritma ANN, Logistic Regression, Naïve Bayes dan Random Forest menunjukan bahwa performa mereka justru mengalami penurunan sekitar 1 % dibandingkan tidak menggunakan analisis komponen utama.

Tabel 12. Perbandingan Performa Algoritma dengan PCA

Algoritma	n-PC's	Akurasi		F1 Score		$\mu_{perform}$
		Train	Test	Train	Test	
KNN	9	83.02	90.11	84.48	91.26	90.69
Neural Network	6	84.43	86.81	85.84	88	87.41
Logistic Regression	6	83.02	85.71	84.35	86.87	86.29
Naïve Bayes	8	80.66	85.71	82.1	86.6	86.16
Random Forest	6	85.38	84.62	86.81	86	85.31



Gambar 4 Perbandingan average performance model dengan PCA

#### IV. KESIMPULAN

Berdasarkan hasil uji coba yang dilakukan dapat disimpulkan bahwa baik algoritma KNN, Neural Network, Logistic Regression, Naïve Bayes, maupun Random Forest merupakan algoritma yang cukup baik untuk digunakan untuk melakukan klasifikasi pada *dataset heart disease*. Dalam penelitian ini baik dilakukan atau tidak PCA terhadap *dataset* KNN, ANN, dan Logistic Regression secara berurutan merupakan algoritma dengan rata-rata *accuracy* dan *f1 score* tertinggi.

Pada algoritma ANN dan Logistic Regression *accuracy* dan *f1 score* berkurang setelah menerapkan PCA meski tidak signifikan, pada algoritma KNN dan Naïve Bayes rata-rata *accuracy* bertambah tidak signifikan, sedangkan pada algoritma Random Forest tidak berubah.

Rekomendasi untuk penelitian berikutnya adalah membandingkan performa dengan k-fold validation, atau membandingkan performa model-model hybrid seperti SVM-ANN [6].

#### REFERENSI

- [1] "Cardiovascular diseases (CVDs)." [https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-\(cvds\)](https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)) (accessed Nov. 21, 2021).
- [2] M. Ari Bianto, "Perancangan Sistem Klasifikasi Penyakit Jantung Menggunakan Naïve Bayes Designing a Heart Disease Classification System Using Naïve Bayes," *Citec Journal*, vol. 6, no. 1, 2019.
- [3] N. Kausar and H. Ghous, "LC INTERNATIONAL JOURNAL OF STEM A COMPARATIVE ANALYSIS ON CLEVELAND AND STATLOG HEART DISEASE DATASETS USING DATA MINING TECHNIQUES." [Online]. Available: [www.lcjstem.com](http://www.lcjstem.com)
- [4] A. Anand, H. Anand, S. S. Rautaray, M. Pandey, and M. K. Gourisaria, "Analysis and prediction of chronic heart diseases using machine learning classification models," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 5, pp. 8479–8487, Sep. 2020, doi: 10.30534/ijatcse/2020/227952020.
- [5] "UCI Machine Learning Repository: Heart Disease Data Set." <https://archive.ics.uci.edu/ml/datasets/Heart+Disease> (accessed Nov. 21, 2021).
- [6] S. Janani, "A Review of Heart Disease Prediction System using Machine Learning Algorithms in Data Mining," *International Journal for Research in Engineering Application & Management (IJREAM)*, vol. 06, pp. 2454–9150, 2020, doi: 10.35291/2454-9150.2020.0689.