IMPLEMENTATION OF FEEDFORWARD NEURAL NETWORK FOR CARDIOVASCULAR DISEASE PREDICTION WITH PERFORMANCE EVALUATION

Muhammad Rafli[1)*], Misbahuddin[2)], Bulkis Kanata[3)]

[1,2,3]*Electrical Engineering*
[1,2,3]*Faculty of Engineering*
[1,2,3]*University of Mataram*
E-mail : raflinsheon24@gmail.com[1)], misbahuddin@unram.ac.id[2)], uqikanata@unram.ac.id[3)]

Abstract

Disease is crucial to prevent more serious complications. This study implemented a Feedforward Neural Network (FNN) algorithm to build a cardiovascular disease risk prediction model using patient clinical data. The dataset used was sourced from open sources and underwent preprocessing stages such as one-hot encoding and normalization. The model architecture consists of two hidden layers with ReLU and dropout activation functions, and an output layer with a sigmoid function for binary classification. Training was conducted for 100 epochs with a data split ratio of 80:20. Evaluation was carried out using accuracy, precision, recall, F1-score, and confusion matrix metrics. The evaluation results showed that the model achieved a training accuracy of 92% and a testing accuracy of 88%, with an average F1-score of 87.2%. The Confidence Factor value also indicated a high level of confidence in each prediction. These results indicate that the FNN model is effective for cardiovascular disease risk prediction and has the potential to be used as a tool for rapid and accurate medical decision-making.

Keywords- *Cardiovascular disease, Feedforward neural network, risk prediction, confidence factor, Machine Learning*

1. INTRODUCTION

Cardiovascular disease is a non- communicable disease that is the leading cause of death worldwide. Approximately 80% of all deaths from this disease occur in low- to middle-income countries, including Indonesia [1]. This disease includes disorders of the heart and blood vessels such as coronary heart disease, heart failure, hypertension, and stroke, which can cause disability and a reduced quality of life [2]. Changes in people's lifestyles that follow social, economic and technological developments have given rise to unhealthy habits such as smoking, alcohol consumption, unbalanced diets, lack of physical activity and obesity [3]. This lifestyle shift contributes to the epidemiological transition from infectious diseases to degenerative diseases such as cardiovascular disease [4].

WHO data shows that in 2005, approximately 17.5 million deaths, or 30% of total global deaths, were caused by heart disease. This figure is expected to increase to nearly 25 million by 2030, primarily due to coronary heart disease and stroke [5]. In the United States, cardiovascular disease accounted for 34.3% of all deaths in 2006 [6]. challenge in treating cardiovascular disease is the difficulty of early detection. Many patients only become aware of their heart condition after symptoms have advanced. Conventional examinations, such as specialist consultations and laboratory tests, are expensive and time- consuming [7]. Therefore, an alternative technology-based approach is needed that is able to predict the risk of disease quickly, accurately, and efficiently. Some of the main risk factors for heart disease include a diet high in fat or carbohydrates, obesity, lack of exercise, smoking habits, and a family history of similar diseases [8].

Medical data-based risk detection and classification are becoming increasingly important to support prevention efforts and medical decision-making. One widely used approach is the application of Artificial Neural Networks (ANNs), an information processing system that mimics the workings of the human brain. ANNs are designed to identify complex patterns and relationships in data and have evolved into a part of the field of artificial intelligence [9]. Classification is a data analysis

DOI : https://doi.org/10.21063/jtif.2025.V13.2.97-104

technique that aims to determine the class of data based on certain features.

Various classification methods have been widely applied in the fields of machine learning, expert systems, and statistics [10]. On the other hand, prediction in the context of medical data is the process of estimating the likelihood of a health condition occurring based on a patient's historical data [11]. In the medical world, the accuracy of predictions is very important, because it is the basis for effective decision making, including in detecting the risk of cardiovascular disease [12]. Some clinical indicators that are often used in predicting heart disease include the type of chest pain, resting blood pressure, cholesterol levels, resting ECG results, maximum heart rate, and fasting blood sugar levels [13]. These factors can be analyzed through machine learning models to provide early and measurable estimates of disease risk [14]. Heart disease remains one of the leading causes of high mortality rates worldwide.

Efforts to predict this disease can be carried out by utilizing the Heart Disease Dataset from the UCI Repository through the application of classification algorithms such as Naive Bayes, Random Forest, and Neural Network. The research findings indicate that the Naive Bayes algorithm provides the best accuracy with a precision rate of 83% [15]. This study analyzes machine learning algorithms for heart failure prediction, a major cause of morbidity and mortality worldwide. Using a patient dataset with diverse clinical features, several classification models—including KNN, Decision Trees, SVM, Random Forest, and Gaussian Naive Bayes—were evaluated through preprocessing, feature selection, training, and validation. Performance was assessed using accuracy, precision, recall, F1-score, and ROC-AUC, with Random Forest achieving the best results. The findings highlight the potential of machine learning to support early intervention and personalized care in heart failure management [16].

Heart disease is a leading cause of death worldwide, requiring effective prediction methods. This study applies the Decision Tree algorithm to clinical data and shows that it achieves good accuracy in predicting heart disease risk. Analysis of key features also highlights major risk factors, indicating the algorithm's potential for clinical decision support [17].

Heart disease is a major cause of mortality worldwide, with risk factors such as cholesterol, diabetes, and high blood pressure. This study compares three machine learning algorithms—Support Vector Machine (SVM), Logistic Regression (LR), and Artificial Neural Network (ANN)—using the UCI public dataset under four train-test split scenarios (90:10, 80:20, 70:30, and 60:40). The best result was achieved by Logistic Regression with 86% accuracy under the 80:20 split [18]. This study presents an automated heart disease diagnosis model using an Optimal Artificial Neural Network (OANN) with feature selection via OBL-GWO. Tested on the Cleveland dataset, the model outperformed other classifiers and achieved 92.54% accuracy in predicting heart disease [19]. Urban development, population growth, and climate change contribute to flood disasters, making green open space a vital prevention measure. This study applies data mining, specifically the Naïve Bayes algorithm, to assess green open space development using existing data patterns. The model achieved an average accuracy of 53.94% and a maximum of 61% with an 80:20 data split [20].

2. METHODOLOGY

This study uses a computational method with a Feedforward Neural Network (FNN) algorithm to build a cardiovascular disease prediction model based on clinical data. The dataset used was obtained from open sources and includes features such as age, gender, blood pressure, cholesterol, heart rate, and ECG results. The data was processed using a one-hot encoding technique for categorical features, then divided into training data and test data with a ratio of 80:20. The FNN model was built using TensorFlow with two hidden layers, each using ReLU activation functions and dropout to prevent overfitting, as well as a sigmoid output layer for binary classification. The model was trained for 100 epochs using Adam optimization and a binary crossentropy loss function. Performance evaluation was carried out by measuring accuracy, loss, confusion matrix, and visualizing the accuracy and loss trends against training.

DOI : https://doi.org/10.21063/jtif.2025.V13.2.97-104

2.1 Data Collection and Data Sources

The data used in this study comes from the public heart.csv dataset available on the Kaggle platform. This dataset contains patient clinical information related to cardiovascular disease risk factors. Some key features in the data include age (Age), gender (Sex), resting blood pressure (RestingBP), cholesterol level (Cholesterol), maximum heart rate (MaxHR), the presence of exertional angina (ExerciseAngina), resting ECG (RestingECG), and ST-slope. The target label for this dataset is HeartDisease, with a value of 0 indicating no heart disease and a value of 1 indicating the presence of heart disease.

2.2 Pre-Processing Data

The initial stage in this research method is data pre-processing, which is carried out to prepare the dataset for use in the model training process. At this stage, categorical columns are converted into numeric form using the One-Hot Encoding method so that they can be processed by the neural network model. After that, the data is divided into two parts: training data and testing data with a ratio of 80:20 to ensure fair and representative model evaluation. Furthermore, normalization of input features is performed to equalize the value scale between features and speed up the model training process.

2.3 Making Model

The heart disease prediction model was designed by building an artificial neural network architecture using TensorFlow and Keras. The model was designed based on several important features in the dataset, such as age, gender, type of chest pain, blood pressure, cholesterol, and other clinical variables. Each feature was fed into the input layer and then combined using a Concatenate layer for further processing. The network structure consists of two consecutive hidden layers with 128 and 64 neurons, each using the ReLU activation function, and a 20% dropout rate was added to reduce the risk of overfitting. The output layer uses a single neuron with a sigmoid activation function to generate a binary classification prediction, which detects whether a patient is at risk of heart disease or not. This architecture was chosen for its ability to handle tabular data and recognize complex non-linear patterns in clinical data.

2.4 Training Model

After the model architecture was designed, the training process was carried out using pre-processed data. The dataset was divided into two parts: 80% training data and 20% testing data using a train-test split method to maintain evaluation objectivity. Categorical features were converted to numeric representations using the One-Hot Encoding technique, while numeric features were normalized to maintain a balanced scale. The model was compiled using the Adam optimizer algorithm with a binary crossentropy loss function, which is suitable for binary classification cases. The training process was carried out for 100 epochs with the help of the tf.data.Dataset module to efficiently manage data batches. During training, model performance was monitored through accuracy and loss metrics on both training and validation data. Training results were also visualized graphically to monitor trends in accuracy improvement and loss reduction, reflecting the model's ability to generalize to new data.

2.5 Evaluation Model

The trained model, saved in .h5 format, was then tested using previously separated test data. Evaluation was conducted to assess the model's generalization ability to new data using the Confusion Matrix metric and evaluation parameters such as precision, recall, and F1-score. These metrics are used to describe the model's ability to distinguish between positive classes (patients with heart disease) and negative classes (patients without heart disease). The architecture used is a Feedforward Neural Network with a sigmoid-activated output layer, resulting in a prediction value ranging from 0 to 1. This value is called the Confidence Factor (CF), which indicates the model's level of confidence in a positive classification. CF is interpreted as the probability that an individual is at risk of heart disease and is used as the basis for a binary classification process with a certain threshold, generally 0.5. This evaluation provides a comprehensive overview of the model's performance in the context of medical predictions that require high levels of accuracy and precision

DOI : https://doi.org/10.21063/jtif.2025.V13.2.97-104

3. RESULTS AND DISCUSSIONS

3.1 Pre-Processing and Sharing Dataset

```
data = pd.read_csv('dataset/heart.csv')
categorical_features = ['Sex',
'ChestPainType', 'RestingECG',
'ExerciseAngina', 'ST_Slope']
data[categorical_features] =
data[categorical_features].astype(str)
#Membagidatasettrain,test=train_test_spli
t(data,test_size=0.2,random_state=42)
```

The dataset was augmented by defining categorical features in the dataset, namely Sex, ChestPainType, RestingECG, ExerciseAngina, and ST_Slope, which are non-numeric features and require special processing. All of these features were converted to string data types using the astype(str) function, in preparation for the one-hot encoding process so that they can be converted into a numeric format that can be used by the machine learning model. After that, the dataset was split into two parts using the train_test_split function from scikit-learn, where 80% of the data was used as training data and the remaining 20% as test data. The use of the random_state=42 parameter ensures that the data split results are consistent every time the code is run.

3.2 Training Model

This stage is the process of training a model using a TensorFlow-based Dense Neural Network architecture suitable for tabular data. Each input feature has its own layer, then combined using Concatenate. The model consists of two hidden layers (128 and 64 neurons) with ReLU activation and 20% dropout to prevent overfitting. The output layer uses one neuron with sigmoid activation for binary classification (HeartDisease 0 or 1). The model is compiled using the Adam optimizer, binary_crossentropy loss function, and accuracy metric, then trained for 100 epochs using the training data and validated with the test data.

```
# Training model
History = model.fit(train_ds,
validation_data=test_ds, epochs- 100)
```

After compilation, the model is trained using the model.fit() function with the train_ds dataset and validated against test_ds for 100 epochs. During this process, the model weights are updated based on the loss and accuracy functions. The training results are stored in a history object, which can be used to analyze trends in accuracy and loss over time

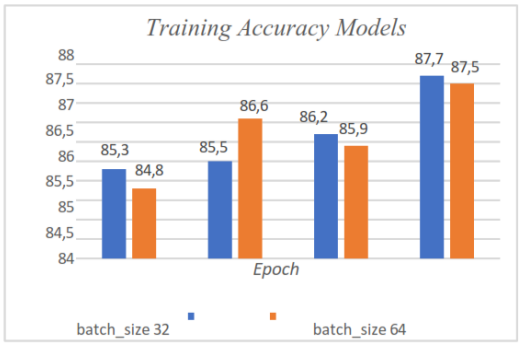

Figure 1. Accuracy Graph on Various Model Training

Based on Figure 1, it can be seen that accuracy tends to increase with increasing number of epochs, both when using batch sizes of 32 and 64. This indicates that the training process is running optimally over time. Batch size 32 showed slightly superior performance compared to batch size 64, especially at epochs 25, 75, and 100. At epoch 50, batch size 64 briefly recorded the highest accuracy. This difference indicates that batch size can affect the effectiveness of model training, although both still provide fairly good accuracy results.

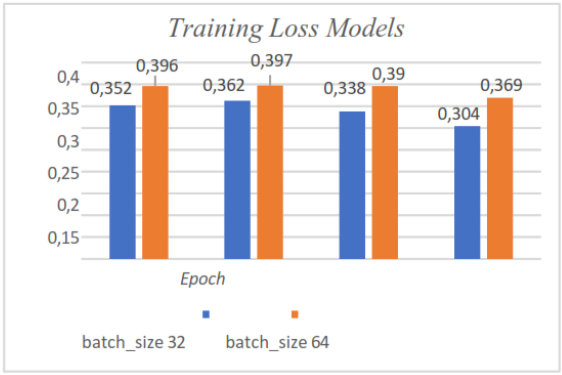

Figure 2. Loss Graph on Various Model Training

DOI : https://doi.org/10.21063/jtif.2025.V13.2.97-104

Figure 2 shows that the loss value is used to measure the extent of the model's prediction error relative to the actual label, with values closer to 0 indicating improved performance. Training results show that the loss value varies depending on the combination of batch size and epoch. Although the loss value for batch size 32 is lower at epoch 25 (0.352) than for batch size 64 (0.396), the overall trend shows that batch size 64 tends to produce more stable loss values. Batch size 64 ranges from 0.396–0.397 in epochs 25 to 75, while batch size 32 experiences a significant decrease, reaching its lowest loss value of 0.319 in epoch 100. This indicates that batch size 32 is more sensitive to increasing epochs, while batch size 64 is more stable but tends to decrease slowly. 3.3 Model Evaluation After training was completed, the model was tested using 20% of the test data (183 data sets) to evaluate its performance in binary classification (classes 0 and 1). The test results showed an accuracy of 88%, indicating that the model generalizes well to new data and has reliable classification performance.

3.3 Evaluation Model

After training was completed, the model was tested using 20% of the test data (183 datasets) to evaluate its performance in binary classification (classes 0 and 1). The test results showed an accuracy of 88%, indicating that the model was able to generalize well to new data and had reliable classification performance.

```
# Prediksi model (hasil berupa
probabilitas 0-1)
prediction = model.predict(input_data)
# Tampilkan hasil prediksi
print(f"Prediksi:
{prediction[0][0]:.4f}")

# Jika ingin mengonversi ke label 0
(Sehat) atau 1 (Penyakit Jantung)
predicted_label = 1 if prediction[0][0]
>= 0.5 else 0

print(f"Pasien diprediksi memiliki
penyakit jantung: {predicted_label}")
```

The source code above processes patient data and generates a probability value between 0 and 1, reflecting the risk level of heart disease. The closer the value is to 1, the higher the predicted risk; conversely, a value closer to 0 indicates a more healthy patient. This probability is presented as quantitative information for healthcare professionals. With a threshold of 0.5, the system classifies results into two labels:≥ 0.5 as positive (at risk), and < 0.5 as negative. This approach provides a combination of detailed risk estimation and clear classification to support rapid decision-making. The test results are then evaluated using a confusion matrix to further measure the model's performance.

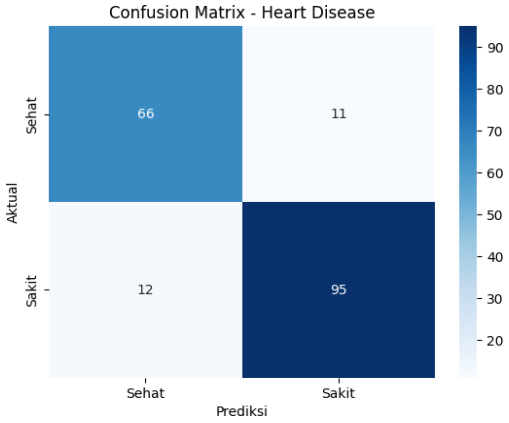

Figure 3. Confusion Matrix of Test Data

Figure 3 shows a confusion matrix obtainedfrom 184 test data sets, showing the distributionof model predictions across two classes: Healthy and Sick. This matrix illustrates the model's accuracy in classifying each class and the location of errors. Several classification errors are evident, such as predicting healthy patients as sick, and vice versa. These errors are likely caused by the similarity of characteristics between classes, particularly in patients with mild symptoms, making it difficult for the model to accurately distinguish between healthy and sick patients. Information on the confusion matrix can be found in Table 1.

Table 1. Confusion Matrix description for all classes

Actual/ Prediction	Healthy	Sick	Total Actual
Healthy	66	11	77
Sick	12	95	107
Total Prediction	78	106	184

Table 1 shows that the model performed quite well in classifying two classes: Healthy and

DOI : https://doi.org/10.21063/jtif.2025.V13.2.97-104

Sick. A True Positive (TP) value of 95 and a True Negative (TN) value of 66 indicates a fairly high model accuracy. However, there were still classification errors, namely 11 False Positive (FP) and 12 False Negative (FN). These errors could be caused by the similarity of symptoms between classes, especially in patients with mild symptoms. To evaluate the model's performance in more detail, precision, recall, and F1-score metrics were used, calculated based on the TP, FP, and FN values. These values provide an overview of how well the model can recognize sick and healthy cases equally and accurately.

Sickness Class Calculation

True Positive (TP) = 95 Sick predicted Sick

False Positive (FP) = 11 Healthy predicted Sick

False Negative (FN) = 12 Sick predicted Healthy

Health Class Calculation

True Positive (TP) = 66 Healthy predicted Healthy

False Positive (FP) = 12 Sick predicted Healthy

False Negative (FN) = 11 Healthy predicted Sick

The calculation results can be seen as a whole in Table 2

Table 2. Calculation Result Parameters

Class	Precision	Recall	F1-Score
Healthy	89,6	88,8	89,2
Sick	84,6	85,7	85,2
Average	**87,1**	**87,3**	**87,2**

Table 1 shows that the model performed quite well in classifying two classes: Healthy and Sick. A True Positive (TP) value of 95 and a True Negative (TN) value of 66 indicates a fairly high model accuracy. However, there were still classification errors, namely 11 False Positive (FP) and 12 False Negative (FN). These errors could be caused by the similarity of symptoms between classes, especially in patients with mild symptoms. To evaluate the model's performance in more detail, precision, recall, and F1-score metrics were used, calculated based on the TP, FP, and FN values. These values provide an overview of how well the model can recognize sick and healthy cases equally and accurately.

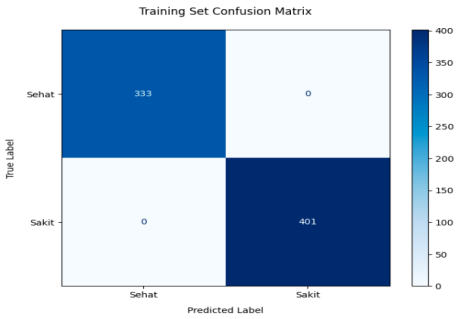

Figure 4. Confusion Matrix of Taining Data

Figure 4 shows the confusion matrix of the model's prediction results on 734 training data sets, consisting of 333 healthy data sets and 401 sick data sets. The model successfully classified all data sets with perfect accuracy without any errors (false positives or false negatives). Despite the high accuracy, these results also indicate the potential for overfitting, where the model overfits to the training data. Therefore, further evaluation with test data is still needed to measure the model's generalization ability to new data.

Table 3. Confidence Factor Values

Data	Confidence Factor	Prediction (0/1)	Original Label
1	0,09	0	0
2	0,78	1	1
3	0,98	1	1
4	0,93	1	1
5	0,09	0	0

Table 3 shows that the model produced accurate predictions for all five tested datasets. The first two datasets (Data 1 and 5) had low Confidence Factor (CF) values of 0.09, and the model predicted class 0 (Healthy), consistent with their original labels. Meanwhile, the other three datasets (Data 2, 3, and 4) had high CFs, ranging from 0.78 to 0.98, and were predicted as class 1 (Sick), also consistent with their original labels. This indicates that the model was able to classify the data accurately and with an appropriate level of confidence in each prediction. The high CF for the Sick class and the low for the Healthy class indicate that the model is not only accurate but also confident in determining the class of each dataset. However, this analysis only included five datasets, so further evaluation with

DOI : https://doi.org/10.21063/jtif.2025.V13.2.97-104

a larger dataset is needed to ensure the model's consistency and generalizability.

4. CONCLUSIONS

This study successfully developed a cardiovascular disease classification model using the Feedforward Neural Network algorithm with relatively high accuracy. The model was able to distinguish healthy and sick patients well, as evidenced by a test accuracy value of 88% and an average precision, recall, and F1-score of 87%. Furthermore, the Confidence Factor value indicated a strong level of confidence in the prediction results. There were still several classification errors that needed to be minimized, especially in cases with mild symptoms. The results of this study indicate that the FFNN approach has great potential for application in medical decision support systems for early detection of heart disease risk. For further research, the use of a larger dataset and additional features is recommended to improve the model's stability and accuracy across various clinical conditions.

ACKNOWLEDGEMENTS

With all respect and sincerity, I would like to express my deepest gratitude to:

Dr. Ir. Misbahuddin, ST., MT., IPU., & Mrs. Bulkis Kanata, ST., MT. as my supervisors who have provided guidance, direction, and time during the process of compiling this thesis. Your diligence and patience in guiding me is something I really appreciate and will never forget.

REFERENCES

[1] I. Rosidawati and H. Aryani, "GAMBARAN TINGKAT RISIKO PENYAKIT KARDIOVASKULAR BERDASARKAN SKOR KARDIOVASKULAR JAKARTA," *Healthc. Nurs. J.*, vol. 4, no. 1, pp. 252–259, Jan. 2022, doi: 10.35568/HEALTHCARE.V4I1.1852.

[2] C. Boukhatem, H. Y. Youssef, and A. B. Nassif, "Heart Disease Prediction Using Machine Learning," in *2022 Advances in Science and Engineering Technology International Conferences, ASET 2022*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/ASET53988.2022.9734880.

[3] S. McLean *et al.*, "The impact of telehealthcare on the quality and safety of care: A systematic overview," *PLoS One*, vol. 8, no. 8, 2013, doi: 10.1371/journal.pone.0071238.

[4] M. Lestari, "Penerapan Algoritma Klasifikasi Nearest Neighbor (K-NN) untuk Mendeteksi Penyakit Jantung," *Fakt. Exacta*, vol. 7, no. September 2010, pp. 366–371, 2014.

[5] N. Nuraeni, "Klasifikasi Data Mining Untuk Prediksi Penyakit Kardiovaskular," *J. Tek. Inf. dan Komput.*, vol. 7, no. 1, p. 161, 2024, doi: 10.37600/tekinkom.v7i1.1276.

[6] D. Pradana, M. Luthfi Alghifari, M. Farhan Juna, and D. Palaguna, "Klasifikasi Penyakit Jantung Menggunakan Metode Artificial Neural Network," *Indones. J. Data Sci.*, vol. 3, no. 2, pp. 55–60, 2022, doi: 10.56705/ijodas.v3i2.35.

[7] A. Sepharni, I. E. Hendrawan, and C. Rozikin, "Klasifikasi Penyakit Jantung dengan Menggunakan Algoritma C4.5," *STRING (Satuan Tulisan Ris. dan Inov. Teknol.*, vol. 7, no. 2, p. 117, 2022, doi: 10.30998/string.v7i2.12012.

[8] A. M. A. Rahim, Inggrid Yanuar Risca Pratiwi, and Muhammad Ainul Fikri, "Klasifikasi Penyakit Jantung Menggunakan Metode Synthetic Minority Over-Sampling Technique Dan Random Forest Clasifier," *Indones. J. Comput. Sci.*, vol. 12, no. 5, pp. 2995–3011, 2023, doi: 10.33022/ijcs.v12i5.3413.

[9] F. F. Utama, B. Warsito, and S. Sugito, "MODEL FEED FORWARD NEURAL NETWORK (FFNN) DENGAN ALGORITMA PARTICLE SWARM SEBAGAI OPTIMASI BOBOT (Studi Kasus : Harga Daging Sapi dari Bank Dunia Periode Januari 2007 – Desember 2018)," *J. Gaussian*, vol. 8, no. 1, pp. 117–126, 2019, doi: 10.14710/j.gauss.v8i1.26626.

[10] A. Riski, "Analisis Komparasi Algoritma Klasifikasi Data Mining Untuk Prediksi Penderita Penyakit Jantung," *J. Tek. Inform. Kaputama*, vol. 3, no. 1, pp. 22–28, 2019, [Online].

Available: https://jurnal.kaputama.ac.id/index.php/JTIK/article/view/141/156

[11] D. Larassati, A. Zaidiah, and S. Afrizal, "Sistem Prediksi Penyakit Jantung Koroner Menggunakan Metode Naive Bayes," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 7, no. 2, pp. 533–546, 2022, doi: 10.29100/jipi.v7i2.2842.

[12] W. S. Dharmawan, "Komparasi Algoritma Klasifikasi Svm-Pso Dan C4.5-Pso Dalam Prediksi Penyakit Jantung," *I N F O R M a T I K a*, vol. 13, no. 2, p. 31, 2022, doi: 10.36723/juri.v13i2.301.

[13] A. Wijayadhi, M. Makmun Effendi, and S. Budi Rahardjo, "Prediksi Penyakit Jantung Dengan Algoritma Regresi Linier," *Bull. Inf. Technol.*, vol. 4, no. 1, pp. 15–28, 2023, doi: 10.47065/bit.v4i1.463.

[14] H. M. Nawawi, J. J. Purnama, and A. B. Hikmah, "Komparasi Algoritma Neural Network Dan Naïve Bayes Untuk Memprediksi Penyakit Jantung," *J. Pilar Nusa Mandiri*, vol. 15, no. 2, pp. 189–194, 2019, doi: 10.33480/pilar.v15i2.669.

[15] D. Derisma, "Perbandingan Kinerja Algoritma untuk Prediksi Penyakit Jantung dengan Teknik Data Mining," *J. Appl. Informatics Comput.*, vol. 4, no. 1, pp. 84–88, 2020, doi: 10.30871/jaic.v4i1.2152.

[16] I. Osei and A. B. Adomako, "Using Machine Learning to Predict Heart Failure: A Comparative Analysis of Various Classification Algorithms," *Int. J. Res. Sci. Innov.*, vol. XI, no. I, pp. 336–354, 2024, doi: 10.51244/ijrsi.2024.1101026.

[17] L. Nafi'ah and Z. Fatah, "Implementasi Algoritma Decision Tree Untuk Pendeteksian Penyakit Jantung," *JUSIFOR J. Sist. Inf. dan Inform.*, vol. 3, no. 2, pp. 160–165, 2024, doi: 10.70609/jusifor.v3i2.5729.

[18] F. Handayani, "Komparasi Support Vector Machine, Logistic Regression Dan Artificial Neural Network Dalam Prediksi Penyakit Jantung," *J. Edukasi dan Penelit. Inform.*, vol. 7, no. 3, p. 329, 2021, doi: 10.26418/jp.v7i3.48053.

[19] R. Thanga Selvi and I. Muthulakshmi, "An optimal artificial neural network based big data application for heart disease diagnosis and classification model," *J. Ambient Intell. Humaniz. Comput.*, vol. 12, no. 6, pp. 6129–6139, 2021, doi: 10.1007/s12652-020-02181-x.

[20] N. Pandiangan and M. L. C. Buono, "Implementation of Naive Bayes for selection of green space areas," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 343, no. 1, 2019, doi: 10.1088/1755-1315/343/1/012241.

DOI : https://doi.org/10.21063/jtif.2025.V13.2.97-104