**Recursive Journal of Informatics**
Vol. 1, No. 2, September 2023
p-ISSN 2963-5551 https://journal.unnes.ac.id/sju/index.php/rji e-ISSN 2986-6588

# Diagnosis of Heart Disease Using Optimized Naïve Bayes Algorithm with Particle Swarm Optimization and Gain Ratio

**Anisa Meidina[1], Zaenal Abidin[2]**
[1,2]Computer Science Department, Faculty of Mathematics and Natural Sciences,
Universitas Negeri Semarang, Indonesia

**Abstract.**
**Purpose:** This study aims to apply feature selection particle swarm optimization (PSO) and gain ratio to the naïve Bayes algorithm and gauging the level of accuracy before and after applying PSO feature selection and gain ratio to the naïve Bayes algorithm in the diagnosis of heart disease.
**Methods/Study design/approach:** Data collection is done by using taking the Cleveland dataset obtained from the UCI machine learning repository. The data used in this study were 303 samples. The data is processed using the preprocessing stage. The naïve Bayes algorithm is used for a classifier, while PSO and gain ratio for feature selection.
**Result/Findings:** The results of the study revealed that the classification accuracy of the naïve Bayes algorithm without the application of feature selection in the Cleveland dataset is 86.88%, while the results of the classification accuracy of the naïve Bayes algorithm after applying PSO and gain ratio in the Cleveland dataset is 93.44%. Application of PSO and gain ratio as feature selection algorithms can improve classification accuracy by 6.56%.
**Novelty/Originality/Value:** This study combines the PSO feature selection and gain ratio on the naïve Bayes algorithm using the Cleveland dataset. The research model that was carried out was enriched by carrying out the preprocessing stages, namely data cleaning, changing the number of class labels, data normalization, and data discretization. This study shows that using a combination of the PSO feature selection algorithm and the gain ratio gives better accuracy to the naïve Bayes algorithm in diagnosing heart disease.

**Keywords**: Diagnosis, Heart, Naïve Bayes, PSO, Gain Ratio
**Received** March 23, 2023 / **Revised** June 24, 2023 / **Accepted** September 14, 2023



## INTRODUCTION

Heart health is crucial for human well-being and quality of life. The heart is one of the most important organs in the human body as it pumps blood, which carries oxygen and nutrients to all the other organs and tissues. Without a healthy heart, the body cannot function properly, and everyday activities can become difficult. Heart or cardiovascular disease is the narrowing or blockage of blood vessels that cause a heart attack, chest pain, or stroke [1]. Heart disease is a disease with a high mortality rate of more than 12 million deaths worldwide due to heart disease [2]. The high number of people with heart disease also impacts on the country's economic losses. Therefore, we need a method of examination to find out whether a person has heart disease. Early diagnosis of heart disease is very important for better outcomes and faster recovery. Early diagnosis of heart disease can use a collection of medical record data to make a pattern for determining heart disease by utilizing data mining with a classification process.

Data mining classification algorithms can assist medical experts in diagnosing heart disease [3]. One of the successful data mining techniques used in diagnosing heart disease patients is the naïve Bayes method [4]. Naïve Bayes is one of the algorithms in data mining techniques that apply Bayes' theory in classification [5]. The naïve Bayes algorithm model has a very minimum error rate and is known for its simple, fast, and highly accurate calculations [6]. The naïve Bayes method is also considered to work better than the other classifier models because it has a better level of accuracy [7]. However, the naïve Bayes algorithm also has weaknesses, namely, the prediction of the probabilities of running is not optimal, and the lack of selecting features that are relevant to classification causes low accuracy [8].

*Corresponding author.
Email addresses: anisameidina15@students.unnes.ac.id (Meidina)
DOI: 10.15294/rji.v1i2.67278

Feature selection is a technique for reducing attribute dimensions. Dimensional reduction is made to obtain relevant and not redundant attributes to speed up the classification process and increase the accuracy of the classification algorithm [9]. Particle swarm optimization (PSO) has superior search performance for solving optimization problems with a faster and more stable convergence rate [10]. The Gain ratio can be used for feature selection and can handle high-dimensional datasets [11]. Both algorithms give good results when combined with several other classifier algorithms. As in research conducted by Utami [12] using PSO feature selection on the SVM and k-nearest neighbor algorithms, and the use of a gain ratio algorithm which can increase the accuracy of the k-nearest neighbors algorithm [13].

Based on the explanation that has been explained, this study focuses on optimizing the classification results of the naïve Bayes algorithm using the particle swarm optimization (PSO) feature selection algorithm and gain ratio in selecting relevant attributes so as to produce the best accuracy in diagnosing heart disease.

## METHODS

In this study, the first step taken was to prepare the dataset to be used, then preprocess the data starting from cleaning the data, namely the process of handling missing values, then changing the number of class labels, data normalization, and data discretization. After that, a feature selection process is carried out and later, the results of the selected features will be used in the classification process. The classification algorithm used is naïve Bayes. The flowchart of the method used in this study is shown in Figure 1.

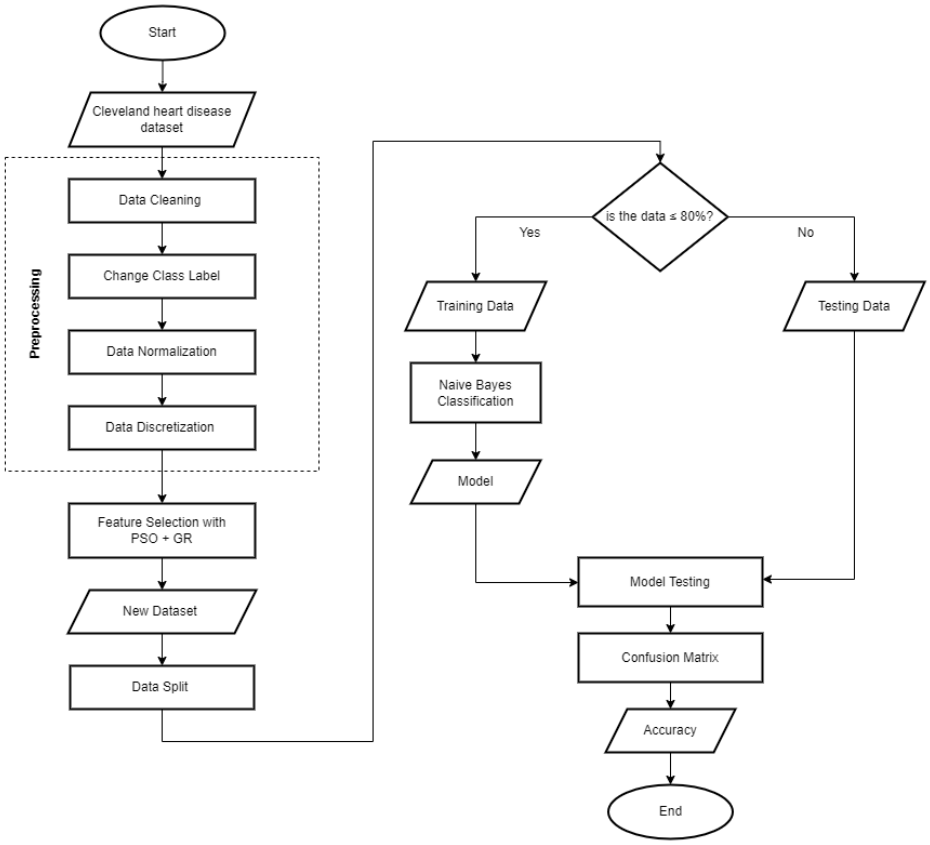


Figure 1. Research model

1) Data collection

The data used in this study is the Cleveland dataset obtained from the UCI machine learning repository via the archive.ics.uci.edu site. This data is public so that anyone can use it. This data consists of 303 samples with 13 attributes and 1 class label. The Cleveland dataset attributes, and their descriptions can be seen in Table 1.

Table 1. Cleveland dataset attributes and their description

| Attributes | Description | Type |
|---|---|---|
| Age | Age | Numerik |
| Sex | Gender | Nominal |
| Cp | Chest pain | Nominal |
| Trestbps | Blood pressure | Numerik |
| Chol | Cholesterol | Numerik |
| Fbs | Blood sugar | Nominal |
| Restecg | Results electrocardiography | Nominal |
| Thalach | Heart rate maximum | Numerik |
| Exang | Induced angina | Nominal |
| Oldpeak | Depression | Numerik |
| Slope | Tilt ST segment | Nominal |
| Ca | Number of vessels main blood colored with fluoroscopy | Numerik |
| Thal | Heart rate type | Nominal |
| Num | Diagnosis heart disease | Nominal |

2) Data preprocessing

Data preprocessing is required so that the dataset is in accordance with the analytical method used. The preprocessing stages of being carried out are as follows.

a. Data cleaning

Data cleaning is used to identify and correct the data to be studied. Data correction was carried out because the data contained missing values or missing values. Fill in the missing value using the most frequent method or frequently appearing value [14].

b. Change class label

The class label on the Cleveland dataset consists of 5 classes, namely 0, 1, 2, 3, and 4. A value of 0 means there is no risk of heart disease, while values of 1, 2, 3, and 4 mean there is a risk of heart disease with different levels of risk. In this study, the number of class labels was changed, which was 5 into 2 classes 0 and 1, where class 0 indicated no heart disease and class 1 indicated heart disease [15].

c. Data normalization

Data normalization is done to balance data values. The normalization method used is Min-Max Normalization. Normalization is performed to change the data value into a data range between 0 to 1 [16].

d. Data discretization

Data discretization is used to change continuous features into discrete ones [17]. Discretization used in this study is a type of equal frequency, which divides data or determines intervals based on the same frequency. The use of discretization uses the scikit-learn library, namely KbinsDiscretizer with n_bins=3.

3) Feature selection using PSO

After the data is preprocessed, the data was applied features selection using PSO algorithm. The stages of feature selection using PSO are as follows [14]:

1. Initialize the PSO parameters, the speed, and position of each particle randomly set within a predetermined range.
2. Calculate the fitness of the particle using the fitness function formula in Equation 1.

$$Fitness = \frac{number\ of\ correctly\ classified\ instances}{number\ of\ instances} \quad (1)$$

3. Determine and update the pbest and gbest values of each particle based on the fitness function with Equation 2 for the pbest value and Equation 3 for the gbest value.

$$if(pos > pbest): pbest = pos \quad (2)$$

$$if(pos > gbest): gbest = pos \quad (3)$$

4. Update the velocity using Equation 4 [18].

$$v_i(t+1) = w.v_i(t) + c_1.r_1(x_i^p - x_i(t) + c_2.r_2(x^g - x_i(t) \quad (4)$$

5. Update the particle position with the sigmoid formula (S) of the updated velocity above with Equation 5. The PSO application uses binary digits to denote features. Selection of selected features is based on Equation 6, features that are not selected are denoted by 0, while selected features are denoted by 1.

$$S = \frac{1}{1+e^{-v_i(t+1)}} \quad (5)$$

$$x_i(t+1) = \begin{cases} 1, if\ rand(0,1) < Sigmoid(S) \\ 0, otherwise \end{cases} \quad (6)$$

The loop stops when it reaches the maximum iteration. The selected attribute is obtained by taking the highest fitness value (with the lowest cost) from all iterations.

4) Feature selection using gain ratio
Feature selection using the gain ratio is made by calculating the importance of all existing data attributes. The stages of feature selection using the gain ratio are as follows [13]:
1. Calculate the entropy value for each attribute in the dataset with Equation 7.

$$Entropy(S) = \sum_{i=1}^{n} -p_i * \log_2 p_i \quad (7)$$

2. Calculate the information gain value for each attribute with Equation 8.

$$Gain\ (S,A) = Entropy(S) - \sum_{i=1}^{n} \frac{|s_i|}{|s|} \times Entropy(S_i) \quad (8)$$

3. Calculate split information for each attribute with Equation 9 [19].

$$SplitInfo(S,A) = -\sum_{i=1}^{n} \frac{S_i}{S} log_2 \frac{S_i}{S} \quad (9)$$

4. Calculate the gain ratio of each attribute using Equation 10.

$$Gain\ Ratio(A) = \frac{Gain\ (S,A)}{SplitInfo(S,A)} \quad (10)$$

Attribute selection is made by sorting the gain ratio value of each attribute, the highest gain ratio value to be selected.

5) Split the data into training data and testing data
Split data is the stage of dividing data into training data and testing data. The distribution of data for training data and testing data with a ratio of 80:20 is the best ratio empirically [20]. The distribution of data is carried out in a consistent random state (random state) with the aim that each calculation has a fixed value.

6) Classification using naïve Bayes
The flow of the naïve Bayes classification algorithm are as follows [21]:
1. Prepare training data. The training data comprises 80% of the entire dataset.
2. Calculate the probability for each parameter. If the data attribute is categorical, calculate the probability value using Equation 11. Meanwhile, if the data attribute is numeric, then calculate the probability value using Gaussian density in Equation 12.

$$P(C|F) = \frac{P(C).P(F|C)}{P(F)} \quad (11)$$

$$P(X_i = x_i | C = c_j) = \frac{1}{\sqrt{2\pi{\sigma_{ij}}^2}} e^{-\frac{(x_i-\mu_{ij})^2}{2\sigma^2 ij}} \quad (12)$$

3. Calculate the final probability value of each class.
   The calculation process will stop when the final probability value of each class is calculated.

7) Evaluate using the confusion matrix
   The evaluation stage is carried out to test the model and calculate the resulting accuracy using the confusion matrix. The calculation is done by calculating the amount of correctly classified data and divided by the number of predictions made. The steps are as follows:
   1. Enter the classification test results in the confusion matrix table, as shown in Table 2.

Table 2. The confusion matrix

| Actual | Predicted | |
|---|---|---|
| | Positive | Negative |
| Positive | True Positive (TP) | False Negative (FN) |
| Negative | False Positive (FP) | True Negative (TN) |

   2. Calculate the accuracy value using Equation 13.

$$Accuracy = \frac{TP+TN}{(TP+TN+FP+FN)} \quad (13)$$

   3. Summarize the accuracy results obtained.

**RESULTS AND DISCUSSION**

This study uses the Cleveland dataset obtained through the UCI machine learning repository collected by Robert Detrano, M.D., Ph.D. from V.A. Medical Center, Long Beach, and Cleveland Clinic Foundation in 1988. This dataset consists of 303 data with 13 attributes and 1 class. The Cleveland dataset consists of numeric and nominal data. The amount of data on class 0 attributes (no heart disease) is 164 data, while the amount of data on class 1, 2, 3, and 4 (heart disease) is 139 data.

After the data is collected, data processing is then carried out through several processes before data classification. These stages are the preprocessing stage, namely data cleaning, changing the number of class labels, normalization, and data discretization. In the Cleveland dataset, there are six missing values in the ca and thal attributes which are marked with a question mark character "?". Missing values are first converted to empty values (NaN). Filling in the missing value is done by replacing the missing value with the most frequent value or the value that frequently appears in the ca and thal attributes. The results of filling in the missing value using the most frequent value are shown in Table 3.

Table 3. The most frequent value to fill in the missing value

| Attributes | Most frequent |
|---|---|
| *Ca* | 0 |
| *Thal* | 3 |

After that, change the number of class labels from 5 (0, 1, 2, 3, and 4) to 2 (0, 1). Class label 0 indicates the absence of heart disease and label class 1 indicates the presence of heart disease. The label class used as a classification can be seen in the num attribute shown in Table 4.

Table 4. Cleveland dataset with two class labels

| no | age | sex | cp | trestbps | chol | fbs | … | num |
|---|---|---|---|---|---|---|---|---|
| 1 | 63 | 1 | 1 | 145 | 233 | 1 | … | 0 |
| 2 | 67 | 1 | 4 | 160 | 286 | 0 | … | 1 |
| 3 | 67 | 1 | 4 | 120 | 229 | 0 | … | 1 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 4 | 37 | 1 | 3 | 130 | 250 | 0 | … | 0 |
| 5 | 41 | 0 | 2 | 130 | 204 | 0 | … | 0 |
| … | … | … | … | … | … | … | … | … |
| 303 | 38 | 1 | 3 | 138 | 175 | 0 | … | 0 |

After that, data normalization using min-max normalization to balance data values by mapping data into the range 0-1. The results of the normalization calculation are shown in Table 5.

Table 5. Dataset after normalization

| no | age | sex | cp | trestbps | chol | fbs | … | num |
|---|---|---|---|---|---|---|---|---|
| 1 | 0,708333 | 1 | 1 | 0,481132 | 0,244292 | 1 | … | 0 |
| 2 | 0,791667 | 1 | 4 | 0,622642 | 0,365297 | 0 | … | 1 |
| 3 | 0,791667 | 1 | 4 | 0,245283 | 0,235160 | 0 | … | 1 |
| 4 | 0,166667 | 1 | 3 | 0,339623 | 0,283105 | 0 | … | 0 |
| 5 | 0,250000 | 0 | 2 | 0,339623 | 0,178082 | 0 | … | 0 |
| … | … | … | … | … | … | … | … | … |
| 303 | 0,187500 | 1 | 3 | 0,415094 | 0,111872 | 0 | … | 0 |

Then the last stage of preprocessing is used as data discretization. The discretization process will divide the range of values into three intervals from the results of normalizing the data. The subset of attributes in the dataset to be discretized are age, tresbps, chol, thalach, oldpeak, and ca. The implementation of discretization in Python programs is called using the scikit-learn library, namely KbinsDiscretizer with n_bins=3. Discretization results are shown in Table 6.

Table 6. Dataset after discretization

| no | age | sex | cp | trestbps | chol | fbs | … | num |
|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 1 | 2 | 1 | 1 | … | 0 |
| 2 | 2 | 1 | 4 | 2 | 2 | 0 | … | 1 |
| 3 | 2 | 1 | 4 | 0 | 1 | 0 | … | 1 |
| 4 | 0 | 1 | 3 | 1 | 1 | 0 | … | 0 |
| 5 | 0 | 0 | 2 | 1 | 0 | 0 | … | 0 |
| … | … | … | … | … | … | … | … | … |
| 303 | 0 | 1 | 3 | 2 | 0 | 0 | … | 0 |

After the preprocessing stage, the feature selection stage is carried out using the PSO algorithm and gain ratio in selecting the best feature set from the Cleveland dataset. The first feature selection process uses the PSO algorithm by determining several parameters, such as the number of particles, the number of iterations, the inertia weight, and the acceleration coefficient. The determination of this parameter follows Chiu's [22] research because it gives convergence results. Table 7 shows the parameters used in this study.

Table 7. PSO Parameters

| Parameter | Value |
|---|---|
| Number of Particles | 50 |
| Number of Iterations | 100 |
| Inertia Weight (w) | 0,72 |
| C1 | 1,49 |
| C2 | 1,49 |

Feature selection in the PSO algorithm is based on the fitness value of each feature (attribute). From this process, ten features are selected that have the highest fitness value. The results of the features selected from the PSO algorithm are not necessarily the best feature set. For this reason, the gain ratio algorithm is applied to the PSO feature selection results. The gain ratio calculates the weight of importance of each attribute (feature). Determining the number of selected features is by determining the value of k. Several

trials of the value of k were carried out with k = 9, 8, 7, and 6. From the results of the 10 experiments that have been carried out, the number k = 9 has the best classification accuracy. The feature selection results from the combination of the PSO algorithm and the gain ratio produce 9 selected features, namely sex, cp, tresbps, chol, fbs, restecg, thalach, exang and oldpeak.

The selected features are continued with the classification process. Dataset classification was carried out two times. The first classification was carried out using only the naïve Bayes algorithm, while the second classification was carried out by applying a combination of PSO feature selection and gain ratio to the naïve Bayes algorithm. From the classification process using naïve Bayes, an accuracy of 86.88% was obtained, while the classification process using the NB+PSO+GR combination obtained an accuracy of 93.44%. A comparison of the accuracy results obtained from the naïve Bayes algorithm and the NB+PSO+GR combination is shown in Figure 2.

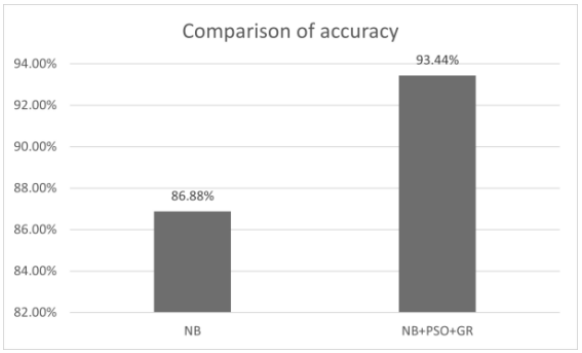

Figure 2. Comparison of accuracy results

Based on Figure 2, it is known that the application of PSO and gain ratio as a feature selection algorithm can increase classification accuracy by 6.56%. The PSO feature selection and gain ratio algorithms can improve the accuracy of the naïve Bayes algorithm in the classification of heart disease using the Cleveland dataset.

## CONCLUSION

The following conclusions can be drawn based on research related to the application of the PSO feature selection algorithm and gain ratio in the naïve Bayes algorithm for heart disease diagnosis in the Cleveland dataset. The Cleveland dataset used is carried out in the preprocessing stage by cleaning the data, changing the number of class labels, normalizing data, and discretizing data. The application of the PSO algorithm as a feature selection uses parameters $c_1, c_2 = 1{,}49$; $w = 0{,}72$; the number of particles = 50; and the number of iterations= 100, 10 features are selected that have the highest fitness value. Application of the gain ratio by calculating the value of the gain ratio for each feature. The combined application of PSO feature selection and gain ratio resulted in 9 selected features namely sex, cp, tresbps, chol, fbs, restecg, thalach, exang and oldpeak. The selected dataset is divided into training data and test data with a ratio of 80:20 to be implemented on the naïve Bayes algorithm.

The results of the evaluation using the confusion matrix in the form of accuracy obtained from the classification process using the naïve Bayes algorithm obtained an accuracy of 86.88%. Then the PSO feature selection method and gain ratio were applied to the naïve Bayes algorithm to obtain an accuracy of 93.44%. From this description, it can be concluded that the combination of the application of the PSO feature selection algorithm and the gain ratio has succeeded in increasing the accuracy of 6.56% in the diagnosis of heart disease. For future research, you can try an appropriate method to determine the best number of ranges or bins in the data discretization process and try other feature selection algorithms that have a shorter time efficiency and are able to get better accuracy.

## REFERENCES

[1] F. Babic, J. Olejar, Z. Vantova, and J. Paralic, "Predictive and descriptive analysis for heart disease diagnosis," in *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems, FedCSIS 2017,* Nov. 2017, pp. 155–163. doi: 10.15439/2017F219.

[2] D. Umadevi and M. Snehapriya, "A survey on prediction of heart disease using data mining

techniques," *Int. J. Sci. Res.*, vol. 6, no. 4, pp. 2228–2232, 2017.
[3] M. Shouman, T. Turner, and R. Stocker, "Applying k-nearest neighbour in diagnosing heart disease patients," *Int. J. Inf. Educ. Technol.*, vol. 2, no. 3, pp. 220–223, 2012.
[4] U. N. Dulhare, "Prediction system for heart disease using naive Bayes and particle swarm optimization.," *Biomed. Res.*, vol. 29, no. 12, pp. 2646–2649, 2018.
[5] M. Ridwan, H. Suyono, and M. Sarosa, "Penerapan data mining untuk evaluasi kinerja akademik mahasiswa menggunakan algoritma naive Bayes classifier," *J. EECCIS (Electrics, Electron. Commun. Control. Informatics, Syst.*, vol. 7, no. 1, pp. 59–64, 2013.
[6] M. Muhathir, M. H. Santoso, and R. Muliono, "Analysis naïve Bayes in classifying fruit by utilizing hog feature extraction," *J. Informatics Telecommun. Eng.*, vol. 4, no. 1, pp. 250–259, 2020, doi: 10.31289/jite.v4i1.3860.
[7] D. Xhemali, C. J. Hinde, and R. G. Stone, "Naïve Bayes vs. decision trees vs. neural networks in the classification of training web pages," *Int. J. Comput. Sci.*, vol. 4, no. 1, pp. 16–23, 2009.
[8] M. R. Fanani, "Algoritma naïve Bayes berbasis forward selection untuk prediksi bimbingan konseling siswa," *J. Disprotek*, vol. 11, no. 1, pp. 13–22, 2020.
[9] M. Arifin, "Ig-knn untuk prediksi customer churn telekomunikasi," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 6, no. 1, pp. 1–10, Apr. 2015, doi: 10.24176/simet.v6i1.230.
[10] I. Muzakkir, A. Syukur, and I. N. Dewi, "Peningkatan akurasi algoritma backpropagation dengan seleksi fitur particle swarm optimization dalam prediksi pelanggan telekomunikasi yang hilang," *Pseudocode*, vol. 1, no. 1, pp. 1–10, 2014.
[11] M. A. Al Karomi and Ivandari, "Optimasi algoritma naïve Bayes dengan information gain ratio untuk menangani dataset berdimensi tinggi," *IC-Tech J. Inform. Comput. Technol.*, vol. 14, no. 2, pp. 18–24, 2019.
[12] L. A. Utami, "Analisis sentimen opini publik berita kebakaran hutan melalui komparasi algoritma support vector machine dan k-nearest neighbor berbasis particle swarm optimization," *J. Pilar Nusa Mandiri*, vol. 13, no. 1, pp. 103–112, 2017.
[13] A. A. Nababan, O. S. Sitompul, and Tulus, "Attribute weighting based k-nearest neighbor using gain ratio," *J. Phys. Conf. Ser.*, vol. 1007, no. 1, pp. 1–6, Apr. 2018, doi: 10.1088/1742-6596/1007/1/012007.
[14] O. Herdiana, S. Maulani, and E. A. Firdaus, "Strategi pemasaran produk industri kreatif menggunakan algoritma k-means clustering berbasis particle swarm optimization," *NUANSA Inform.*, vol. 15, no. 2, pp. 1–13, Aug. 2021, doi: 10.25134/nuansa.v15i2.4394.
[15] M. S. Amin, Y. K. Chiam, and K. D. Varathan, "Identification of significant features and data mining techniques in predicting heart disease," *Telemat. Informatics*, vol. 36, pp. 82–93, 2019, doi: 10.1016/j.tele.2018.11.007.
[16] V. M. Purnama, W. Astuti, and A. Adiwijaya, "Analisis perbandingan klasifikasi microarray menggunakan naïve Bayes dan support vector machine (SVM) untuk deteksi kanker dengan feature extraction PCA," *eProceedings Eng.*, vol. 8, no. 5, pp. 9974–9986, 2021.
[17] H. Rahmawan, "Penentuan rekomendasi pelatihan pengembangan diri bagi pegawai negeri sipil menggunakan algoritma c4.5 dengan principal component analysis dan diskritisasi," *J. Tekno Kompak*, vol. 14, no. 1, pp. 5–10, Feb. 2020, doi: 10.33365/jtk.v14i1.531.
[18] X. Xu, H. Rong, M. Trovati, M. Liptrott, and N. Bessis, "CS-PSO: chaotic particle swarm optimization algorithm for solving combinatorial optimization problems," *Soft Comput.*, vol. 22, no. 3, pp. 783–795, 2018, doi: 10.1007/s00500-016-2383-8.
[19] I. Yulianti, R. A. Saputra, M. S. Mardiyanto, and A. Rahmawati, "Optimasi akurasi algoritma C4.5 berbasis particle swarm optimization dengan teknik bagging pada prediksi penyakit ginjal kronis optimization of C4.5 algorithm based on particle swarm optimization with bagging technique on prediction of chronic Kidney Dise," *Techno.Com*, vol. 19, no. 4, pp. 411–421, 2020, [Online]. Available: https://archive.ics.uci.edu/ml/
[20] A. Gholamy, V. Kreinovich, and O. Kosheleva, "Why 70/30 or 80/20 relation between training and testing sets: a pedagogical explanation," *Dep. Tech. Reports (CS).*, pp. 1–6, 2018.
[21] M. Sabransyah, Y. N. Nasution, and F. D. T. Amijaya, "Aplikasi metode naive Bayes dalam prediksi resiko penyakit jantung," *EKSPONENSIAL*, vol. 8, no. 2, pp. 111–118, 2017.
[22] C. Y. Chiu, Y. F. Chen, I. T. Kuo, and H. C. Ku, "An intelligent market segmentation system using k-means and particle swarm optimization," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 4558–4565, 2009, doi: 10.1016/j.eswa.2008.05.029.