Science in Information Technology Letters
Vol. 5., No. 2, November 2025, pp. 13-24
ISSN 2722-4139
http://pubs2.ascee.org/index.php/sitech

Comparative analysis of decision tree and random forest classifiers for structured data classification in machine learning

Agnes Nola Sekar Kinasih [a,1], Anik Nur Handayani [a,2,*], Jevri Tri Ardiansah [a,3], Nor Salwa Damanhuri [b,4]

[a] Department of Electrical Engineering and Informatic, Universitas Negeri Malang, Malang 65145, Indonesia
[b] Electrical Engineering Studies, Universiti Teknologi MARA (UiTM) Cawangan Pula, Malaysia
[1] agnes.nola.2405348@students.um.ac.id; [2] aniknur.ft@um.ac.id; [3] jevri.ardiansah.ft@um.ac.id;
[4] norsalwa071@uitm.edu.my
* Corresponding Author

ARTICLE INFO

Article history
Received September 22, 2024
Revised October 10, 2024
Accepted November 24, 2024

Keywords
Machine Learning
Random Forest
Decision Tree
Clustering

ABSTRACT

This study explores the application of machine learning techniques, specifically classification, to improve data analysis outcomes. The primary objective is to evaluate and compare the performance of Decision Tree and Random Forest classifiers in the context of a structured dataset. Using the Elbow Method for optimal clustering alongside decision tree and random forest for classification algorithms, this research investigates the effectiveness of each method in accurately categorizing data. The study employs K-Means clustering to segment the data and Decision Trees and Random Forests for classification tasks. Dataset used in this research was obtained from Kaggle consisting of 13 attributes and 1048575 rows, all of which are numeric. The key results show that Random Forest outperforms Decision Trees in terms of classification accuracy, precision, recall, and F1 score, providing a more robust model for data classification. The performance improvement observed in Random Forest, particularly in handling complex datasets, demonstrates its superiority in generalizing across varied classes. The findings suggest that for applications requiring high accuracy and reliability, Random Forest is preferable to Decision Trees, especially when the dataset exhibits high variability. This research contributes to a deeper understanding of how different machine learning models can be applied to real-world classification problems, offering insights into the selection of the most appropriate model based on specific data characteristics.

1. Introduction

In recent years, machine learning techniques have become integral to addressing complex classification challenges across various fields [1]. These techniques offer significant advantages in identifying underlying patterns and structures within large and high-dimensional datasets, thereby enabling more accurate predictions and data-driven insights [2], [3]. One such technique, clustering, plays a pivotal role in grouping data points based on their inherent similarities, facilitating a more

effective analysis of the data. The Elbow Method is a widely adopted approach for determining the optimal number of clusters, relying on the principle that the sum of squared distances between data points and their respective cluster centroids decreases with the addition of more clusters, but at a diminishing rate beyond a certain threshold [4]–[6].

Classification, a supervised learning task, builds upon clustering by assigning data to predefined categories or classes based on learned patterns [7], [8]. Among the most widely used classifiers are Decision Trees and Random Forests, which have demonstrated their effectiveness in a range of applications. Decision Trees provide a straightforward, interpretable approach to classification by recursively partitioning the data based on feature values [9]. However, their performance can degrade when faced with overly complex datasets, often leading to overfitting [10]. Random Forests, an ensemble learning method, address this issue by aggregating the predictions of multiple Decision Trees, thereby enhancing model robustness and improving generalization [11].

This research aims to explore and compare the performance of Decision Tree and Random Forest classifiers within the context of a structured dataset through the use of performance metrics containing accuracy, precision, recall, and f1-score. By assessing their classification accuracy and generalization capabilities, the study seeks to provide a deeper understanding of their strengths and limitations. The findings aim to inform decisions on which machine learning models are best suited for specific classification challenges, ultimately contributing to the optimization of machine learning workflows.

2. Method

The research process in this study is organized into stages as shown in Fig 1. It begins with data collection, followed by preprocessing, where the data undergoes normalization to ensure consistency. The next stage applies Principal Component Analysis (PCA) for dimensional reduction, which then K-Means clustering is performed to group the data, and classification is carried out using Decision Tree and Random Forest models. Finally, the models are evaluated based on their accuracy and performance metrics to determine effectiveness.

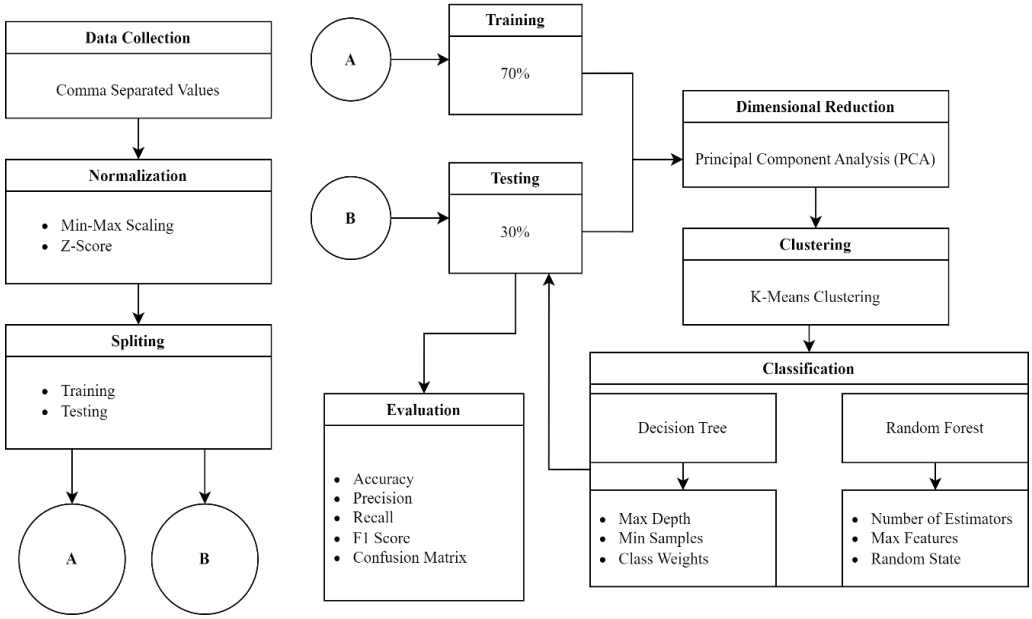

Fig. 1. Data Processing Flow

2.1. Data Collection

Dataset used for this research consists of sensor data collected from a Permanent Magent Synchronous Motor (PMSM) deployed on a test bench by LEA department at Paderborn University, accessed through Kaggle titled "Electric Motor Temperature" [12]. The dataset consists of 13 attributes as presented in Table 1 along with the descriptions and 1048575 rows, with each row representing one snapshot of sensor data at a certain time step.

Table 1. Attributes in Electric Motor Temperature

Attributes	Description
u_q	Voltage q-component measurement in dq-coordinates (in V)
coolant	Coolant temperature (in °C)
stator_winding	Stator winding temperature (in °C) measured with thermocouples
u_d	Voltage d-component measurement in dq-coordinates
stator_tooth	Stator tooth temperature (in °C) measured with thermocouples
motor_speed	Motor speed (in rpm)
i_d	Current d-component measurement in dq-coordinates
i_q	Current q-component measurement in dq-coordinates
pm	Permanent magnet temperature (in °C) measured with thermocouples and transmitted wirelessly via a thermography unit.
stator_yoke	Stator yoke temperature (in °C) measured with thermocouples
ambient	Ambient temperature (in °C)
torque	Motor torque (in Nm)
profile_id	Measurement session id. Each distinct measurement session can be identified through this integer id.

2.2. Data Preprocessing

Prior to data normalization, the 'profile_id' column was removed from the dataset, as it is an identifier rather than a feature relevant to the model. This step ensures that only the necessary features are included for normalization. Data Normalization is a widely used data preprocessing technique that scales or transforms data to ensure equal contribution of each feature, adjusting the data to meet consistent impact in analysis and modeling [13], [14]. The normalization techniques used in this research are:

- Min-Max Normalization, a normalization method that performs a linear transformation to rescale data from one range to another, ensuring a balanced comparison between the original and transformed values [15], [16]. This method uses the formula expressed in Equation (1) to transform the data to the target range which in this research is 0 to 1.

$$x = \frac{(x)-min\,(x)}{x-min\,(x)} \tag{1}$$

- Z-Score Normalization, this method is performed by rescaling the data to have zero mean and unit variance, achieved by subtracting the mean and dividing by the standard deviation, a technique that standardizes both the mean and variance of gradients across layers, ensuring consistent scaling [17]. The method is shown in Equation (2).

$$z = \frac{x-\mu}{\sigma} \tag{2}$$

The normalized data is subsequently divided into training and testing sets to ensure the model is trained on one subset and evaluated on another, facilitating unbiased performance assessment [18]. Data split show in Fig. 2.

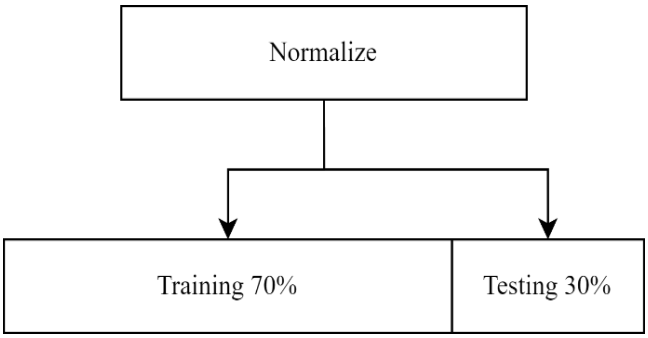

Fig. 2. Data Split

Data splitting is a crucial step in ensuring that the model is trained and tested on distinct subsets of the data, helping to prevent overfitting and ensuring generalization [19]. For this study, the dataset is split into 70% for training and 30% for testing. The training data is used to build the model, while the testing data is reserved for evaluating its performance. This method ensures that the model's performance is assessed on unseen data, providing a reliable measure of its accuracy.

2.3. Dimensionality Reduction and Clustering

Dimensionality reduction techniques are used to reduce the number of input variables in high-dimensional datasets, improving machine learning efficiency, reducing computational complexity, and mitigating the "curse of dimensionality" [20]. In this research, Principal Component Analysis (PCA) is applied after normalization as it produces better resuilts when dimensionality of the datasets is high by identifying a smaller set of principal components that capture the most important information in the data which reduces computational complexity, memory requirements, and improves algorithm efficiency, while preserving critical information [21], [22].

Following dimensionality reduction, K-Means clustering is applied to group the dataset into distinct clusters based on the patterns identified in the reduced feature space. K-means clustering algorithm partitions datasets into clusters by finding the minimum squared error between the various data points in the data set and the mean of a cluster, which are subsequently assigned to the nearest cluster centre [23]. K-means cluster algorithm can be expressed as Equation (3).

$$J(c) = \sum_{k=1}^{K} \quad \sum_{x_i} \quad \|x_i - \mu_k\|^2 \tag{3}$$

Where J(c) is the cost function, Ck denotes the set of points assigned to cluster k, and μk is the centroid of cluster k. The algorithm is iterated through necessary conditions for minimizing the k-means objective function, which continues until convergence [24]. This approach is applied to the PCA reduced data in this research to improve clustering efficiency and focus on the most significant features.

2.4. Classification

In this study, classification models were applied to the clusters generated through K-Means clustering, aiming to categorize the dataset into meaningful groups. Two classification algorithms were used for comparison, Decision Tree and Random Forest. These models were selected for their simplicity, interpretability, and effectiveness in handling diverse datasets [25], [26].

- Decision Tree

  The algorithm works by recursively splitting the dataset based on feature values that provides the best discriminatory power, resulting in a tree-like structure [27]. Each internal node represents a decision based on the input feature, and the leaves represent the specific classification outcome or value of the tested attribute. The tree is constructed using a greedy approach, selecting the best feature to split the data at each step [28]. In this study, the Decision Tree was trained using the normalized data and the K-Means cluster labels

- Random Forest

  Random Forest, an ensemble learning technique, builds multiple decision trees and aggregates their results to improve predictive performance and reduce overfitting [29]–[31]. Each tree in the forest is trained on a random subset of the data, and predictions of each are combined through majority voting to produce the final classification. The Random Forest model was used to compare its performance to the Decision Tree, with expectations of improved accuracy and reduced variance. This method is often preferred in situations where robustness and generalization are important [32].

2.5. Testing

The dataset was divided into training and testing subsets with a 7:3 ratio. The clustering phase utilized K-Means to partition dataset with Elbow method applied to determine the optimal number of clusters. The classification tested two models: Decision Tree and Random Forest, both evaluated through the use of a 5-fold cross validation [33], [34].

2.6. Evaluation

The performance of all used algorithm are evaluated based on confusion matrix. Table 2 shows the formula of each metric utilized [35].

Table 2. Performance Model Evaluation

Attributes	Formula	Description
Accuracy	$\frac{T_P + T_N}{T_P + T_N + F_P + F_N}$	Ratio of total number of correct classifications to the total number of all classifications
Precision	$\frac{T_P}{T_P + F_P}$	Ratio of true positive predictions to the total number of positive predicitons made
Recall	$\frac{T_P}{T_P + F_N}$	Ratio of true positive predictions to the total number of actual positive instances
F1-Score	$\frac{2T_P}{2T_P + F_P + F_N}$	Mean of precision and recall

3. Results and Discussion

In this research, dimensionality reduction, clustering, and classification model has been applied. The effectiveness of each is evaluated through elbow method and confusion matrix.

3.1. Dimensionality Reduction and Clustering Result

The application of Principal Component Analysis (PCA) allowed dimensionality reduction of the dataset, retaining ±41.67% and capturing the relevant features while maintaining data variance as shown on Fig. 3.

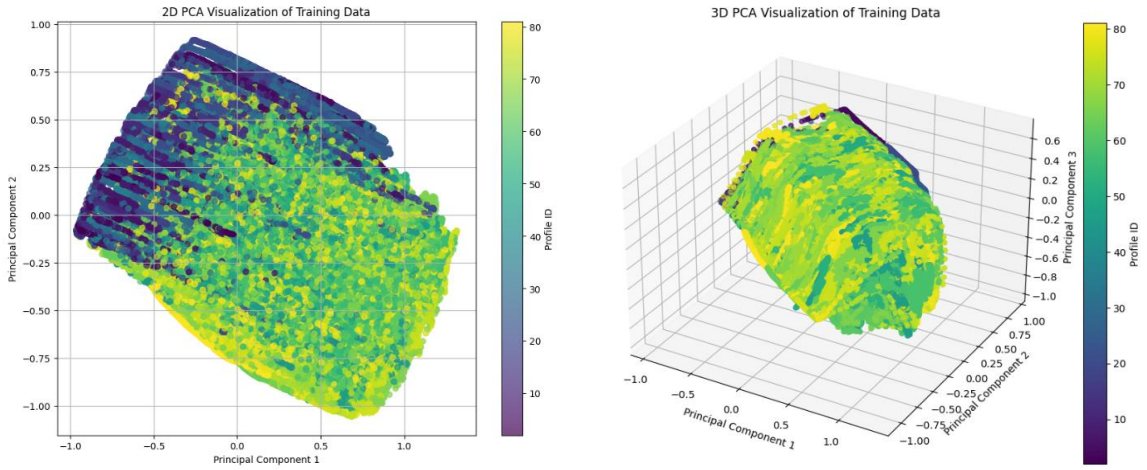

Fig. 3. 2D (Left) and 3D (Right) PCA Visualization

The elbow method has been employed to determine the optimal number of clusters for the K-means clustering algorithm. The Elbow Method, depicted in Fig. 4, revealed a significant drop in inertia up to K=3, after which the improvement plateaued. This indicates that three clusters optimally represent the dataset's structure.

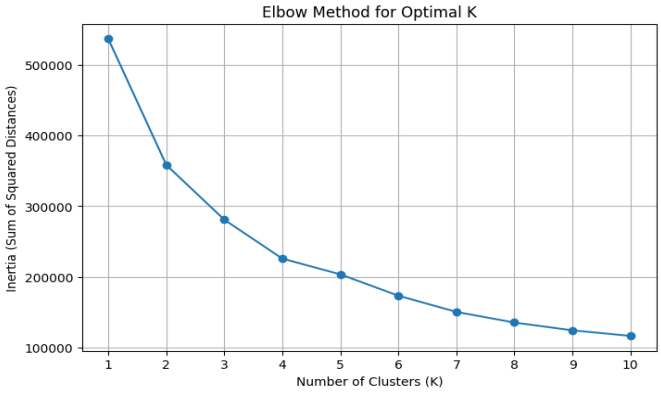

Fig. 4. Elbow Method for Optimal K

With K=3 selected, K-means clustering has been applied to the normalized dataset, resulting in Fig. 5, offering representation of how the data points are grouped based on features.

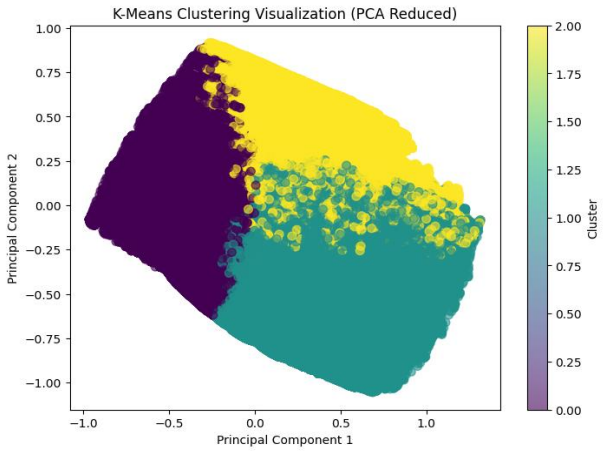

Fig. 5. K-Means Clustering

3.2. Decision Tree

The Decision Tree Classifier was trained using the dataset partitioned into clusters through the use of K-means clustering, achieving an accuracy of 94%. The model's performance was evaluated using a classification report containing accuracy, precision, recall, and F1 score, as well as by examining the confusion matrix. The results presented in Table 3 show a generally strong performance with average accuracy, precision, recall, and F1 scores all above 91%.

Table 3. Decision Tree Classification Report

	Precision	Recall	F1-Score
0	0.94	0.92	0.93
1	0.96	0.97	0.96
2	0.92	0.94	0.93
accuracy			0.94

These results suggest that the Decision Tree classifier is fairly accurate in identifying and distinguishing between the different clusters. However, certain clusters may be harder to predict, as reflected in the slight variation in the metrics across different clusters. In addition to the classification report, the confusion matrix is presented in Fig. 6.

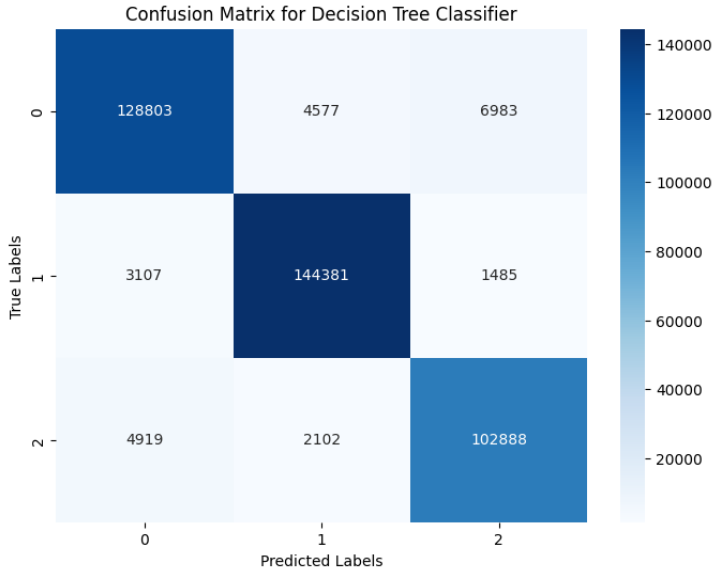

Fig. 6. Decision Tree Confusion Matrix

From the decision tree confusion matrix in Fig. 6, it appears that decision tree performs best for class 1 while misclassifications primarily occur between class 0 and class 2, indicating overlap in features or patters between the two classes

3.3. Random Forest

The Random Forest Classifier was trained using the dataset with clusters identified by K-means, intended as a comparison to the Decision Tree model. The model's performance was evaluated in the same manner as decision tree by using classification report containing accuracy, precision, recall, and F1 score as well as by examining the confusion matrix. The results presented in Table 4 show a generally strong performance with average accuracy, precision, recall, and F1 scores all above 94%.

Table 4. Decision Tree Classification Report

	Precision	Recall	F1-Score
0	0.95	0.96	0.96
1	0.98	0.97	0.98
2	0.95	0.95	0.95
accuracy			0.96

These results however, especially the accuracy suggests that random forest is superior by exceeding the overall score by 2-3%. The difference is slim however it is still superior regardless. Confusion matrix presented in Fig. 7 in order to provide further information for comparison against decision tree.

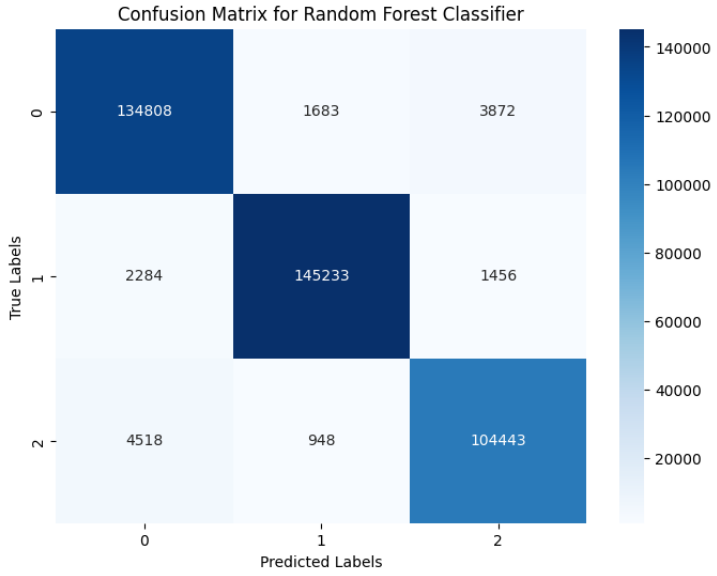

Fig. 7. Random Forest Confusion Matrix

Random Forest confusion matrix presented in Fig. 7 displays the consistent outperformance against Decision Tree across all classes, as indicated by the higher number of correctly classified instances and fewer missclassifications, further elaborated in Table 5.

Table 5. Decision Tree Random Forest Comparison

	Decision Tree	Random Forest	Percentage Improvement
0	91.76%	96.04%	4.28%
1	96.92%	97.49%	0.57%
2	93.61%	95.03%	1.41%

The comparison highlights that the Random Forest model demonstrates superior performance over the Decision Tree model, with the largest improvement observed for class 0 at 4.28% and the smallest improvement for class 1 at 0.57%. The notable improvement in class 0 suggests that the Random Forest model is particularly effective at handling the inherent variability or potential noise within this class, leading to fewer misclassifications. For class 1, while the improvement is marginal, it still reflects the model's ability to maintain a high level of accuracy. Overall, the consistent improvement across all classes, even if slight, underscores the robustness of the Random Forest model. This robustness

likely stems from its ensemble approach, which reduces overfitting and enhances generalization, making it better suited to handle complex patterns and diverse data distributions compared to a single Decision Tree. Additionally, the reduced misclassification rates across classes indicate that the Random Forest model is better equipped to minimize errors, improving reliability and predictive consistency.

The comparison highlights that the Random Forest model demonstrates superior performance over the Decision Tree model, with the largest improvement observed for class 0 at 4.28% and the smallest improvement for class 1 at 0.57%. The notable improvement in class 0 suggests that the Random Forest model is particularly effective at handling the inherent variability or potential noise within this class, leading to fewer misclassifications. For class 1, while the improvement is marginal, it still reflects the model's ability to maintain a high level of accuracy. Overall, the consistent improvement across all classes, even if slight, underscores the robustness of the Random Forest model. This robustness likely stems from its ensemble approach, which reduces overfitting and enhances generalization, making it better suited to handle complex patterns and diverse data distributions compared to a single Decision Tree. Additionally, the reduced misclassification rates across classes indicate that the Random Forest model is better equipped to minimize errors, improving reliability and predictive consistency.

4. Conclusion

This study demonstrated the effectiveness of combining K-means clustering and decision tree compared with random forest classification for analyzing complex datasets. The Elbow Method applied to the K-Means algorithm revealed that the optimal number of clusters was 3, as indicated by the sharp drop in inertia followed by diminishing returns. The Decision Tree model performed reasonably well, with class-wise accuracies of 91% at the minimum while Random Forest model, achieved 95% minimum, showing improvement of at least 4% in class-wise accuracy compared to Decision Tree. From the accuracy percentage, it can be concluded that the Random Forest model outperformed the Decision Tree, achieving higher accuracy and better overall performance, as reflected in its confusion matrix, where fewer misclassifications were observed across all clusters. This improvement can be attributed to the ensemble nature of Random Forest, which helps mitigate overfitting and enhances generalization. Ultimately, the findings indicate that using K-Means for clustering, followed by Random Forest for classification, is a robust approach for analyzing complex datasets. The study highlights the potential of ensemble learning methods, such as Random Forest in enhancing predictive performance, particularly in scenarios involving varied class distributions. By demonstrating the improvements achieved over decision tree and random forest model through comparison, this research contributes to the growing body of evidence supporting ensemble methods as a robust solution for classification tasks. Additionally, the findings highlight opportunities for further exploration in both clustering and classification approaches. Future research could focus on exploring other clustering algorithms, such as DBSCAN or Hierarchical Clustering, and applying more advanced models like Gradient Boosting or Neural Networks. Additionally, hyperparameter tuning and feature engineering could further enhance the clustering and classification outcomes, providing opportunities for refining these methods in future studies. These advancements would not only improve predictive performance but also contribute to developing more adaptable and efficient machine learning pipelines for real-world applications.

Acknowledgment

The authors would like to express their sincere gratitude to Universitas Negeri Malang for their valuable support in facilitating this research. We appreciate the access to resources, collaboration opportunities, and academic contributions that have enriched our study. We acknowledge all parties who contributed, directly or indirectly, to this research. Their support and dedication are greatly appreciated.

Declarations

Author contribution. All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

Funding statement. None of the authors have received any funding or grants from any institution or funding body for the research.

Conflict of interest. The authors declare no conflict of interest.

Additional information. No additional information is available for this paper

References

[1] P. Carracedo-Reboredo *et al.*, "A review on machine learning approaches and trends in drug discovery," *Comput. Struct. Biotechnol. J.*, vol. 19, pp. 4538–4558, 2021, doi: 10.1016/j.csbj.2021.08.011.

[2] L. E. Lwakatare, A. Raj, I. Crnkovic, J. Bosch, and H. H. Olsson, "Large-scale machine learning systems in real-world industrial settings: A review of challenges and solutions," *Inf. Softw. Technol.*, vol. 127, p. 106368, Nov. 2020, doi: 10.1016/j.infsof.2020.106368.

[3] H. Sun, H. V. Burton, and H. Huang, "Machine learning applications for building structural design and performance assessment: State-of-the-art review," *J. Build. Eng.*, vol. 33, p. 101816, Jan. 2021, doi: 10.1016/j.jobe.2020.101816.

[4] H. Humaira and R. Rasyidah, "Determining The Appropiate Cluster Number Using Elbow Method for K-Means Algorithm," Feb. 2020, doi: 10.4108/eai.24-1-2018.2292388.

[5] F. Sutomo *et al.*, "Optimization Of The K-Nearest Neighbors Algorithm Using The Elbow Method On Stroke Prediction," *J. Tek. Inform.*, vol. 4, no. 1, pp. 125–130, Feb. 2023, doi: 10.52436/1.jutif.2023.4.1.839.

[6] M. Cui, "Introduction to the K-Means Clustering Algorithm Based on the Elbow Method," *Accounting, Audit. Financ.*, vol. 1, no. 1, pp. 5–8, Oct. 2020. [Online]. Available at: http://www.clausiuspress.com/article/592.html.

[7] J. Deng and J.-G. Yu, "A simple graph-based semi-supervised learning approach for imbalanced classification," *Pattern Recognit.*, vol. 118, p. 108026, Oct. 2021, doi: 10.1016/j.patcog.2021.108026.

[8] S.-C. Huang, A. Pareek, M. Jensen, M. P. Lungren, S. Yeung, and A. S. Chaudhari, "Self-supervised learning for medical image classification: a systematic review and implementation guidelines," *npj Digit. Med.*, vol. 6, no. 1, p. 74, Apr. 2023, doi: 10.1038/s41746-023-00811-0.

[9] L. L. Custode and G. Iacca, "Evolutionary Learning of Interpretable Decision Trees," *IEEE Access*, vol. 11, pp. 6169–6184, 2023, doi: 10.1109/ACCESS.2023.3236260.

[10] M. M. Mijwil and R. A. Abttan, "Utilizing the Genetic Algorithm to Pruning the C4.5 Decision Tree Algorithm," *Asian J. Appl. Sci.*, vol. 9, no. 1, pp. 2321–0893, Feb. 2021, doi: 10.24203/ajas.v9i1.6503.

[11] I. D. Mienye and Y. Sun, "A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects," *IEEE Access*, vol. 10, pp. 99129–99149, 2022, doi: 10.1109/ACCESS.2022.3207287.

[12] "Electric Motor Temperature," *Kaggel*, 2021. [Online]. Available at: https://www.kaggle.com/datasets/wkirgsn/electric-motor-temperature/data.

[13] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Appl. Soft Comput.*, vol. 97, p. 105524, Dec. 2020, doi: 10.1016/j.asoc.2019.105524.

[14] L. Huang, J. Qin, Y. Zhou, F. Zhu, L. Liu, and L. Shao, "Normalization Techniques in Training DNNs: Methodology, Analysis and Application," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 8, pp. 10173–10196, Aug. 2023, doi: 10.1109/TPAMI.2023.3250241.

[15] H. Henderi, "Comparison of Min-Max normalization and Z-Score Normalization in the K-nearest neighbor (kNN) Algorithm to Test the Accuracy of Types of Breast Cancer," *IJIIS Int. J. Informatics Inf. Syst.*, vol. 4, no. 1, pp. 13–20, Mar. 2021, doi: 10.47738/ijiis.v4i1.73.

[16] M. C. Bagaskoro, F. Prasojo, A. N. Handayani, E. Hitipeuw, A. P. Wibawa, and Y. W. Liang, "Hand image reading approach method to Indonesian Language Signing System (SIBI) using neural network and multi layer perseptron," *Sci. Inf. Technol. Lett.*, vol. 4, no. 2, pp. 97–108, Nov. 2023, doi: 10.31763/sitech.v4i2.1362.

[17] J. Yun, H. Kim, S. Cho, and H. Kang, "ZNorm: Z-Score Gradient Normalization for Accelerating Neural Network Training," *arxiv Artif. Intell.*, pp. 1–12, 2024, [Online]. Available at: http://arxiv.org/abs/2408.01215.

[18] Q. H. Nguyen *et al.*, "Influence of Data Splitting on Performance of Machine Learning Models in Prediction of Shear Strength of Soil," *Math. Probl. Eng.*, vol. 2021, no. 1, p. 4832864, Jan. 2021, doi: 10.1155/2021/4832864.

[19] I. O. Muraina, "Ideal Dataset Splitting Ratios in Machine Learning Algorithms: General Concerns for Data Scientists and Data Analysts," *7th Int. Mardin Artuklu Sci. Res. Conf.*, no. February, pp. 496–504, 2022, [Online]. Available at: https://www.researchgate.net/publication/358284895.

[20] W. Jia, M. Sun, J. Lian, and S. Hou, "Feature dimensionality reduction: a review," *Complex Intell. Syst.*, vol. 8, no. 3, pp. 2663–2693, Jun. 2022, doi: 10.1007/s40747-021-00637-x.

[21] J. P. Bharadiya, "A Tutorial on Principal Component Analysis for Dimensionality Reduction in Machine Learning," *Int. J. Innov. Res. Sci. Eng. Technol.*, vol. 8, no. 5, pp. 2028–2032, 2023, [Online]. Available at: https://www.researchgate.net/publication/371306692.

[22] G. T. Reddy *et al.*, "Analysis of Dimensionality Reduction Techniques on Big Data," *IEEE Access*, vol. 8, pp. 54776–54788, 2020, doi: 10.1109/ACCESS.2020.2980942.

[23] A. M. Ikotun, A. E. Ezugwu, L. Abualigah, B. Abuhaija, and J. Heming, "K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data," *Inf. Sci. (Ny).*, vol. 622, pp. 178–210, Apr. 2023, doi: 10.1016/j.ins.2022.11.139.

[24] K. P. Sinaga and M.-S. Yang, "Unsupervised K-Means Clustering Algorithm," *IEEE Access*, vol. 8, pp. 80716–80727, 2020, doi: 10.1109/ACCESS.2020.2988796.

[25] S. Mishra, P. Shukla, and R. Agarwal, "Analyzing Machine Learning Enabled Fake News Detection Techniques for Diversified Datasets," *Wirel. Commun. Mob. Comput.*, vol. 2022, no. 1, pp. 1–18, Mar. 2022, doi: 10.1155/2022/1575365.

[26] Y. Wu and Y. Chang, "Ransomware Detection on Linux Using Machine Learning with Random Forest Algorithm," *Authorea Preprints*. Authorea, pp. 1–12, Jun. 07, 2024, doi: 10.36227/techrxiv.171778770.06550236/v1.

[27] Z. Azam, M. M. Islam, and M. N. Huda, "Comparative Analysis of Intrusion Detection Systems and Machine Learning-Based Model Analysis Through Decision Tree," *IEEE Access*, vol. 11, pp. 80348–80391, 2023, doi: 10.1109/ACCESS.2023.3296444.

[28] M. Janota and A. Morgado, "SAT-Based Encodings for Optimal Decision Trees with Explicit Paths," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12178 LNCS, Springer, 2020, pp. 501–518, doi: 10.1007/978-3-030-51825-7_35.

[29] M. Mafarja *et al.*, "Classification framework for faulty-software using enhanced exploratory whale optimizer-based feature selection scheme and random forest ensemble learning," *Appl. Intell.*, vol. 53, no. 15, pp. 18715–18757, Aug. 2023, doi: 10.1007/S10489-022-04427-X/TABLES/12.

[30] D. Elavarasan and P. M. D. R. Vincent, "A reinforced random forest model for enhanced crop yield prediction by integrating agrarian parameters," *J. Ambient Intell. Humaniz. Comput.*, vol. 12, no. 11, pp. 10009–10022, Nov. 2021, doi: 10.1007/s12652-020-02752-y.

[31] C. Dunn, N. Moustafa, and B. Turnbull, "Robustness Evaluations of Sustainable Machine Learning Models against Data Poisoning Attacks in the Internet of Things," *Sustainability*, vol. 12, no. 16, p. 6434, Aug. 2020, doi: 10.3390/su12166434.

[32] D. Sun, J. Xu, H. Wen, and Y. Wang, "An Optimized Random Forest Model and Its Generalization Ability in Landslide Susceptibility Mapping: Application in Two Areas of Three Gorges Reservoir, China," *J. Earth Sci.*, vol. 31, no. 6, pp. 1068–1086, Dec. 2020, doi: 10.1007/s12583-020-1072-9.

[33] A. Shebl, D. Abriha, M. Dawoud, M. Ali Hussein Ali, and Á. Csámer, "PRISMA vs. Landsat 9 in lithological mapping – a K-fold Cross-Validation implementation with Random Forest," *Egypt. J. Remote Sens. Sp. Sci.*, vol. 27, no. 3, pp. 577–596, Sep. 2024, doi: 10.1016/j.ejrs.2024.07.003.

[34] I. K. Nti, O. Nyarko-Boateng, and J. Aning, "Performance of Machine Learning Algorithms with Different K Values in K-fold CrossValidation," *Int. J. Inf. Technol. Comput. Sci.*, vol. 13, no. 6, pp. 61–71, Dec. 2021, doi: 10.5815/ijitcs.2021.06.05.

[35] R. Yacouby and D. Axman, "Probabilistic Extension of Precision, Recall, and F1 Score for More Thorough Evaluation of Classification Models," in *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, Nov. 2020, pp. 79–91, doi: 10.18653/v1/2020.eval4nlp-1.9.