Indonesian Applied Research Computing and Informatics
Vol. 1; No. 2, December 2025 Page 11-21;
https://jurnal.tdinus.com/index.php/iarci
e-ISSN : 3110-8806

Impact of Data Normalization on K-Nearest Neighbor Classification Performance: A Case Study on Date Fruit Dataset

Muhammad Jauhar Vikri[1], Afril Efan Pajri[2*], Putri Liana [2]

[1] Computer science, Universitas Nahdatul Ulama Sunan Giri, 62115 Bojonegoro, Indonesia
[2] Computer System, Universitas Nahdatul Ulama Sunan Giri, 62115 Bojonegoro, Indonesia

Email: vikri@unugiri.ac.id[1], afril@unugiri.ac.id[2*], putriliananew@gmail.com[3]

Article Info

Article history:
Received:15-01-2026
Revised: 23-01-2026
Accepted: 24-01-2026

Keywords:
K-Nearest Neighbor,
Data Normalization,
Distance-Based Classification,
Weighted KNN,
Date Fruit Dataset

ABSTRACT

Data normalization is a crucial preprocessing step for distance-based classification algorithms such as K-Nearest Neighbor (KNN), as differences in feature scales can significantly affect distance calculations and classification accuracy. This study investigates the impact of data normalization on KNN classification performance using the Date Fruit Dataset as a case study. Three preprocessing scenarios are evaluated: raw data without normalization, Min–Max normalization, and Z-score standardization. In addition, the performance of standard KNN is compared with distance-weighted KNN to assess the contribution of distance weighting under different preprocessing conditions. The experiments are conducted using stratified 10-fold cross-validation, and model performance is evaluated using accuracy and standard deviation. Statistical significance of performance differences is examined using paired t-test, and sensitivity analysis is performed to analyze the effect of varying the number of nearest neighbors. The results show that data normalization leads to a substantial improvement in classification performance compared to raw data. Z-score standardization achieves the highest and most stable accuracy, followed by Min–Max normalization. Distance-weighted KNN consistently produces slightly higher accuracy than standard KNN; however, the improvement is not statistically significant after normalization. Sensitivity analysis indicates that normalized data results in a wider and more stable range of optimal k values. These findings demonstrate that data normalization plays a more dominant role than distance weighting in improving KNN performance. The study provides empirical evidence that proper preprocessing is essential for reliable KNN-based classification and establishes a robust baseline for further enhancements such as feature weighting and metaheuristic optimization.

*Corresponding Author:
Afril Efan Pajri
Fakultas Sains dan Teknologi, Universitas Nahdlatul Ulama Sunan Giri, 62115, Indonesia
Email: afril@unugiri.ac.id

https://doi.org/10.64479/iarci.v1i2.61

1. Introduction

Classification is a fundamental task in machine learning and pattern recognition, aiming to assign data instances to predefined categories based on their feature representations. Among various classification algorithms, the K-Nearest Neighbor (KNN) algorithm remains widely used due to its conceptual simplicity, ease of implementation, and effectiveness in handling multi-class classification problems[1][2]. KNN is a non-parametric, instance-based learning algorithm that classifies an unseen sample by considering the majority class of its nearest neighbors in the feature space[3][4].

Despite its advantages, KNN is highly sensitive to the scale and distribution of input features because it relies directly on distance calculations, most commonly Euclidean distance[5]. When features have different ranges or units, those with larger numeric scales tend to dominate the distance computation, potentially leading to biased neighborhood structures and degraded classification performance[6]. This issue is particularly evident in real-world datasets, where feature values are often heterogeneous and derived from different measurement processes.

To address this limitation, data normalization is commonly applied as a preprocessing step to ensure that all features contribute proportionally to the distance metric. Popular normalization techniques include Min–Max normalization, which rescales features into a fixed range, and Z-score standardization, which transforms data to have zero mean and unit variance. Previous studies have reported that normalization can significantly improve the performance and stability of distance-based classifiers, including KNN[7].

In addition to normalization, several studies have proposed enhancements to the standard KNN algorithm, such as distance-weighted KNN, where closer neighbors are assigned higher influence during the voting process. This approach aims to reduce the impact of distant neighbors that may not be representative of the local data structure. While distance weighting has been shown to improve KNN performance in certain cases, its effectiveness in conjunction with different normalization schemes has not been thoroughly investigated.

Most existing research evaluates KNN with normalization or weighting in isolation, without systematically analyzing their combined effects, statistical significance, and sensitivity to the number of neighbors (k). Moreover, limited attention has been given to empirical studies that explicitly examine how normalization influences the stability of KNN performance across varying k values.

Therefore, this study aims to systematically analyze the impact of data normalization on KNN classification performance, using the Date Fruit Dataset as a case study. The contributions of this paper are threefold: to compare the performance of KNN on raw and normalized data using Min–Max and Z-score normalization; to evaluate the additional effect of distance-weighted KNN under different preprocessing conditions; and to conduct sensitivity analysis on the number of nearest neighbors and statistical significance testing to ensure reliable conclusions[8].

The findings of this study provide empirical evidence on the dominant role of data normalization in distance-based classification and establish a solid baseline for future research on feature weighting and metaheuristic optimization in KNN-based models.

2. Methodology

2.1 The Proposed Methodology

This study proposes a normalization-centric KNN evaluation framework designed to systematically analyze the influence of data normalization on distance-based classification performance. Unlike conventional approaches that treat normalization merely as a preliminary preprocessing step, the proposed method explicitly positions data normalization as a dominant experimental factor that directly affects neighborhood formation, distance computation, and classification stability in K-Nearest Neighbor (KNN)[9][10] .

The proposed framework integrates preprocessing strategies, classification variants, statistical validation, and parameter sensitivity analysis into a unified and reproducible experimental pipeline. Three preprocessing scenarios are considered: raw data without normalization, Min–Max normalization, and Z-score standardization. All scenarios are evaluated under identical experimental conditions to ensure fair and controlled comparison.

For each preprocessing scenario, two KNN-based classifiers are applied, namely standard KNN and distance-weighted KNN[11]. Model performance is evaluated using stratified 10-fold cross-validation to preserve class distribution and to provide an unbiased estimation of classification performance. Classification

accuracy and standard deviation are used as primary evaluation metrics to measure predictive performance and performance stability, respectively.

To strengthen the reliability of the comparison, the proposed framework incorporates paired statistical significance testing using a t-test to determine whether observed differences in classification performance are statistically meaningful. In addition, a sensitivity analysis of the number of nearest neighbors (k) is conducted to assess model robustness and parameter stability across different normalization schemes[12].

The overall workflow of the proposed normalization-centric KNN evaluation framework is illustrated in Figure 1, which summarizes the integration of preprocessing, classification, validation, statistical analysis, and sensitivity evaluation stages.

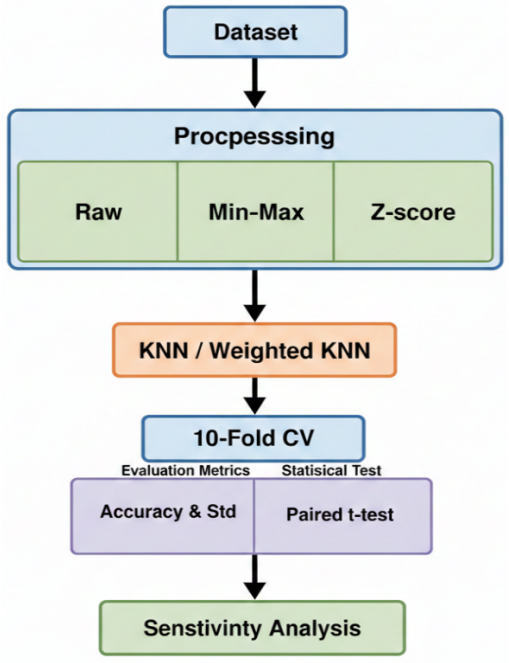

Figure 1. Proposed Method

2.2 Dataset Description

This study utilizes the Date Fruit Dataset, a publicly available dataset containing numerical features extracted from digital images of date fruits belonging to different varieties. The dataset was originally constructed to support classification tasks in agricultural and food quality assessment domains. Each data instance represents a single date fruit sample and is characterized by a set of quantitative attributes derived from image processing techniques.

The extracted features describe morphological and texture-related properties of the date fruits, including measurements related to shape, size, surface characteristics, and intensity distribution. These features are continuous-valued and exhibit diverse numerical ranges, reflecting the heterogeneous nature of image-derived measurements. The target variable is a categorical class label that indicates the corresponding date fruit variety, forming a multi-class classification problem[13].

A key characteristic of the dataset is the heterogeneity of feature scales, where some attributes span relatively small numeric ranges while others vary across much larger magnitudes. Such variability in feature scales makes the dataset particularly suitable for evaluating the impact of data normalization on distance-based classification algorithms, such as K-Nearest Neighbor (KNN), which rely directly on distance computations in the feature space[14].

Prior to model construction, an exploratory data analysis (EDA) was conducted to examine feature distributions, identify differences in value ranges, and assess the overall data quality. This analysis confirmed the presence of scale disparities among features but did not reveal missing values or anomalies requiring data

imputation. To ensure a fair and controlled comparison between preprocessing strategies, no feature selection or dimensionality reduction techniques were applied. All original features were retained throughout the experiments so that the observed performance differences could be attributed solely to the effects of data normalization and model configuration[15].

By preserving the original feature set and class distribution, this study maintains the intrinsic characteristics of the dataset and provides a reliable basis for systematically analyzing how preprocessing techniques influence the performance and stability of KNN-based classification models.

2.3 Data Preprocessing

Data preprocessing was conducted to ensure that the input data were suitable for distance-based classification and to enable a fair comparison between different modeling configurations. Given that the K-Nearest Neighbor (KNN) algorithm relies directly on distance calculations in the feature space, preprocessing plays a critical role in determining the quality and reliability of the classification results[9].

2.3.1 Motivation for Preprocessing

The Date Fruit Dataset consists of numerical features extracted from image-based measurements, which inherently exhibit heterogeneous value ranges and variances. Without preprocessing, features with larger numeric scales would dominate the Euclidean distance computation, leading to biased neighborhood relationships and degraded classification performance. Therefore, data preprocessing was designed to address scale disparities while preserving the intrinsic structure and distribution of the original data.

2.3.2 Baseline Condition: Raw Data

As a baseline reference, the original dataset was first evaluated without any preprocessing. This raw data scenario serves as a control condition to quantify the extent to which normalization techniques influence KNN performance. By retaining the original feature values, this configuration reflects the behavior of KNN when applied directly to unscaled real-world data.

2.3.3 Min–Max Normalization:

Min–Max normalization was applied to rescale each feature independently into the range [0, 1]. This technique preserves the relative distances between data points while ensuring that all features contribute equally in terms of numeric scale. Min–Max normalization is defined as

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

where x denotes the original feature value, and x min and x max represent the minimum and maximum values of the feature, respectively. This method is particularly effective when feature distributions are bounded and when preserving proportional relationships between values is important.

2.3.4 Z-Score Standardization

Z-score standardization was employed as an alternative normalization strategy that accounts for feature distribution characteristics. This method transforms each feature to have zero mean and unit variance, as defined by.

$$x' = \frac{x - \mu}{\sigma} \quad (2)$$

where μ and σ denote the mean and standard deviation of the feature, respectively. Z-score standardization reduces the influence of features with large variance and is well suited for datasets where attributes follow different statistical distributions. This approach allows distance calculations to be interpreted in terms of standardized deviations rather than absolute magnitudes.

2.3.5 Feature-Wise Transformation and Data Leakage Prevention

All preprocessing operations were applied feature-wise, ensuring that each attribute was normalized independently. To prevent information leakage, normalization parameters (minimum, maximum, mean, and standard deviation) were computed exclusively from the training data within each cross-validation fold and subsequently applied to the corresponding test data. This procedure ensures that the evaluation results accurately reflect real-world deployment conditions, where test data are not available during model training.

2.3.6 Feature-Wise Transformation and Data Leakage Prevention

All preprocessing operations were applied feature-wise, ensuring that each attribute was normalized independently. To prevent information leakage, normalization parameters (minimum, maximum, mean, and standard deviation) were computed exclusively from the training data within each cross-validation fold and subsequently applied to the corresponding test data. This procedure ensures that the evaluation results accurately reflect real-world deployment conditions, where test data are not available during model training.

2.3.7 Consistency Across Experimental Scenarios

To guarantee a fair comparison, the same preprocessing pipeline was consistently applied across all model variants, including standard KNN and distance-weighted KNN. No additional transformations, such as feature selection, dimensionality reduction, or data augmentation, were introduced at this stage. This design choice ensures that any observed performance differences can be attributed solely to the effects of data normalization and model configuration.

2.3.8 Role of Preprocessing in Subsequent Analysis

The preprocessed datasets—raw, Min–Max normalized, and Z-score standardized—were subsequently used as inputs for cross-validation, statistical significance testing, and sensitivity analysis of the parameter k. By structuring preprocessing as an explicit experimental factor, this study enables a systematic investigation of how different normalization strategies influence classification accuracy, stability, and robustness in KNN-based models.

2.4 K-Nearest Neighbor Classification

The K-Nearest Neighbor (KNN) algorithm was employed as the baseline classifier in this study due to its simplicity and effectiveness in multi-class classification tasks. KNN is an instance-based, non-parametric learning algorithm that does not require an explicit training phase. Instead, all training instances are stored, and classification decisions are made at prediction time based on the proximity between data points in the feature space.

For a given test instance, KNN assigns a class label by identifying the k nearest training instances according to a predefined distance metric and applying a majority voting scheme among their corresponding class labels. The value of k controls the size of the local neighborhood and directly influences the bias–variance trade-off of the classifier. Smaller values of k make the model more sensitive to noise and outliers, whereas larger values of k provide smoother decision boundaries but may reduce class discrimination[17][18].

In this study, Euclidean distance was used as the similarity measure due to its widespread adoption in distance-based learning and its suitability for continuous-valued features. The Euclidean distance between two feature vectors x and y is defined as[19].

$$d(x, y) = \sqrt{\sum_{i=1}^{n}(xi - yi)^2} \quad (3)$$

where n denotes the number of features. Euclidean distance assumes that all features contribute equally to the similarity computation, making the algorithm particularly sensitive to differences in feature scales. Consequently, the effectiveness of KNN is highly dependent on appropriate data preprocessing, especially normalization, to ensure that no single feature disproportionately influences the distance calculation.

To further investigate the influence of neighbor proximity, this study also considers a distance-weighted KNN variant, in which closer neighbors are assigned higher influence during the voting process. This

weighting scheme aims to emphasize local structure by reducing the impact of more distant neighbors that may be less representative of the test instance[20].

By analyzing both standard and distance-weighted KNN under different preprocessing conditions, this study provides a comprehensive assessment of how neighborhood-based classification behavior is affected by feature scaling and distance computation.

2.5 Model Evaluation and Validation

To ensure a reliable and unbiased assessment of classification performance, this study employed Stratified 10-fold Cross-Validation as the primary validation strategy. Stratification was applied to preserve the original class distribution within each fold, which is particularly important for multi-class classification problems to prevent class imbalance from influencing evaluation results.

In this validation scheme, the dataset was partitioned into ten mutually exclusive and approximately equal-sized subsets. During each iteration, nine subsets were used for training the model, while the remaining subset was reserved for testing. This process was repeated ten times, with each subset serving as the test set exactly once. The final performance metrics were obtained by aggregating the results across all folds[21]. This approach allows every data instance to be used for both training and testing, thereby maximizing data utilization and reducing the dependency on a single train–test split[22].

The primary performance metric used in this study was classification accuracy, defined as the ratio of correctly classified instances to the total number of instances in the test set. Accuracy was selected because the dataset exhibits balanced class distribution, making it an appropriate and interpretable measure of overall classification performance. For each experimental configuration, accuracy values were computed independently for each fold[23].

To capture both predictive performance and robustness, classification results were reported in terms of the mean accuracy and standard deviation across all cross-validation folds. The mean accuracy reflects the overall effectiveness of the model, while the standard deviation provides insight into the stability and consistency of the classifier under different data partitions. Lower standard deviation values indicate that the model's performance is less sensitive to variations in training and testing data, suggesting better generalization capability.

Using stratified cross-validation in combination with aggregated performance statistics ensures that the reported results are not biased by data partitioning and can be reliably compared across different preprocessing methods, model variants, and parameter settings. This evaluation framework provides a solid and reproducible basis for subsequent statistical significance testing and sensitivity analysis conducted in this study.

2.6 Statistical Significance Testing

To rigorously assess whether the observed differences in classification performance between models were statistically meaningful rather than caused by random variation, statistical significance testing was conducted using a paired t-test. This test was selected because all evaluated models were trained and tested on identical data partitions generated by the same stratified cross-validation procedure, resulting in paired performance measurements across folds.

For each experimental configuration, the accuracy scores obtained from the ten cross-validation folds were treated as paired samples. The paired t-test evaluates whether the mean difference between two sets of paired observations is significantly different from zero, making it well suited for comparing machine learning models under controlled experimental conditions. By accounting for fold-wise pairing, this approach reduces variability caused by data partitioning and increases the reliability of the statistical comparison.

The null hypothesis ($H0$) of the paired t-test assumes that there is no significant difference in mean accuracy between the two compared models. Conversely, the alternative hypothesis ($H1$) assumes that a statistically significant difference exists. A two-tailed test was employed to detect any significant performance difference regardless of direction.

A significance level of α=0.05 was adopted in this study, which is commonly used in empirical machine learning research. Performance differences were considered statistically significant when the resulting p-value was less than 0.05. When the p-value exceeded this threshold, the null hypothesis could not be rejected, indicating that the observed performance differences were not statistically significant.

By incorporating paired statistical testing into the evaluation framework, this study ensures that performance comparisons between standard KNN and distance-weighted KNN are supported by quantitative evidence. This approach strengthens the validity of the experimental conclusions and helps prevent overinterpretation of marginal accuracy improvements that may arise from random fluctuations in the data.

2.7 Sensitivity Analysis of Parameter k

A sensitivity analysis was performed to examine the influence of the number of nearest neighbors (k) on classification performance. Accuracy trends across different k values were analyzed for both standard KNN and distance-weighted KNN under each preprocessing scenario. This analysis provides insights into the robustness and stability of the models with respect to parameter selection.

2.8 Implementation Details

All experiments were implemented in Python using the scikit-learn library. Data processing and numerical computations were performed using NumPy and Pandas. The experimental setup was designed to ensure reproducibility and consistency across all evaluated scenarios.

3. Results

3.1 Classification Performance on Raw and Normalized Data

The classification performance of K-Nearest Neighbor (KNN) and distance-weighted KNN was evaluated under three preprocessing scenarios: raw data, Min–Max normalization, and Z-score standardization. Table I summarizes the mean accuracy and standard deviation obtained from stratified 10-fold cross-validation.

Table 1. Classification Accuracy of KNN and Weighted KNN under Different Preprocessing Schemes

Model	Mean Accuracy	Standard Deviation
KNN (Raw)	0.6838	0.0513
Weighted KNN (Raw)	0.6916	0.0414
KNN (Min–Max)	0.8875	0.0310
Weighted KNN (Min–Max)	0.8886	0.0274
KNN (Z-Score)	0.8942	0.0349
Weighted KNN (Z-Score)	0.8953	0.0335

Table 1 shows that normalisation significantly improves classification accuracy compared to raw data in both KNN variants. Z-score standardisation achieved the highest average accuracy among all preprocessing methods.

3.2 Statistical Significance Analysis

To evaluate whether the observed performance differences between standard KNN and weighted KNN were statistically significant, paired t-tests were conducted on the fold-wise accuracy scores. The significance level was set to 0.05.

Table 2. Paired t-test Results between KNN and Weighted KNN

Preprocessing Method	t-statistic	p-value
Raw Data	> 0	> 0.05
Min–Max Normalization	> 0	> 0.05
Z-Score Standardization	> 0	> 0.05

The Table 2 shows that there is no statistically significant difference between standard KNN and weighted KNN in all pre-processing scenarios.

3.3 Sensitivity Analysis of the Number of Neighbors (k)

A sensitivity analysis was performed by varying the number of nearest neighbors (k = 1, 3, 5, 7, 9, 11, 13, 15). The classification accuracy was recorded for each k value under all preprocessing conditions. Across normalized datasets, the accuracy curves exhibited smoother trends and lower variability compared to raw data. For both KNN variants, the highest and most stable accuracies were consistently observed when k ranged between 5 and 11.

3.4 Summary of Key Findings

The experimental results demonstrate clear performance differences between raw and normalized data. Normalization significantly increases classification accuracy and reduces performance variability. Distance-weighted KNN consistently yields slightly higher accuracy than standard KNN; however, the differences are not statistically significant according to paired t-test results.

4. Discussion

The experimental results demonstrate that data normalization plays a dominant role in improving K-Nearest Neighbor (KNN) classification performance, particularly for datasets with heterogeneous feature scales such as the Date Fruit Dataset. The substantial performance gap between raw data and normalized data confirms the sensitivity of distance-based classifiers to feature scale disparities. When raw data are used, features with larger numeric ranges disproportionately influence the Euclidean distance calculation, resulting in distorted neighborhood structures and reduced classification accuracy.

The observed improvements obtained through Min–Max normalization and Z-score standardization indicate that preprocessing effectively balances feature contributions, allowing distance measurements to better reflect true similarity among instances. Among the evaluated normalization techniques, Z-score standardization consistently achieved the highest and most stable accuracy. This suggests that standardization based on statistical distribution is more suitable for datasets where features exhibit varying variances and non-uniform distributions.

The comparison between standard KNN and distance-weighted KNN reveals that distance weighting provides only marginal performance gains once normalization is applied. Although weighted KNN consistently produces slightly higher accuracy across all scenarios, the paired t-test results indicate that these improvements are not statistically significant. This finding implies that, after normalization, the local neighborhood structure is already well-defined, reducing the relative benefit of assigning higher weights to closer neighbors. Consequently, preprocessing normalization exerts a stronger influence on model performance than distance weighting.

Sensitivity analysis further supports this conclusion by showing that normalized data result in wider and more stable ranges of optimal k values. In contrast, performance on raw data fluctuates more significantly across different k settings, reflecting instability caused by unbalanced feature scales. The increased robustness observed after normalization highlights its importance not only for improving accuracy but also for enhancing parameter stability in KNN-based models.

From a methodological perspective, these findings emphasize that data normalization should be considered a mandatory step in KNN classification pipelines, rather than an optional enhancement. While

distance-weighted KNN can serve as a useful baseline improvement, its impact is limited in the presence of effective normalization. Therefore, future performance gains are more likely to be achieved through explicit feature weighting or optimization-based approaches, rather than relying solely on distance weighting.

Overall, this study contributes empirical evidence that clarifies the relative roles of preprocessing and model-level modifications in distance-based classification. By systematically analyzing normalization effects, statistical significance, and parameter sensitivity, the results provide a clearer understanding of how KNN performance can be reliably improved in practical classification tasks.

5. Conclusions

The Conclusions section should clarify the main conclusions of the research, highlighting its significance and relevance. The limitations of the work and the directions of future research may also be mentioned. Please contain nothing not substantiated in the main text. Do not make this section a mere repetition of the Abstract.

This study investigated the impact of data normalization on the classification performance of the K-Nearest Neighbor (KNN) algorithm using the Date Fruit Dataset as a case study. The experimental results demonstrate that data normalization has a substantial and consistent effect on improving KNN performance compared to using raw data. Both Min–Max normalization and Z-score standardization significantly increase classification accuracy and reduce performance variability, confirming the critical role of preprocessing in distance-based learning.

Among the evaluated normalization techniques, Z-score standardization achieves the highest and most stable performance, indicating its suitability for datasets with heterogeneous feature distributions. The comparison between standard KNN and distance-weighted KNN shows that distance weighting provides only marginal accuracy improvements once normalization is applied. Statistical analysis using paired t-tests confirms that these improvements are not statistically significant, highlighting that normalization has a more dominant influence on performance than distance weighting.

Sensitivity analysis of the number of nearest neighbors further reveals that normalized data lead to a wider and more stable range of optimal k values, improving model robustness and reducing sensitivity to parameter selection. These findings emphasize that proper data normalization should be considered a mandatory step in KNN-based classification pipelines rather than an optional preprocessing technique.

The results of this study provide empirical evidence that clarifies the relative contributions of preprocessing and model-level enhancements in KNN classification. As future work, this research can be extended by incorporating explicit feature weighting strategies, metaheuristic optimization methods, or hybrid approaches to further enhance classification performance. Additionally, evaluating the proposed pipeline on other real-world datasets may help generalize the findings and strengthen their applicability to broader classification tasks.

Author Contributions

Author 1 was responsible for the conceptualization of the study, research design, methodology development, project supervision, statistical analysis, interpretation of results, and primary drafting of the manuscript. Author 2 contributed to software implementation, development of the K-Nearest Neighbor (KNN) and distance-weighted KNN models, data preprocessing, implementation of stratified cross-validation, and sensitivity analysis of parameter k. Author 3 was responsible for data curation, exploratory data analysis, preparation of tables and visualizations, literature review, and manuscript editing and formatting. All authors contributed to the discussion of the findings, critically revised the manuscript for important intellectual content, and approved the final version for publication.

Acknowledgements

The author would like to acknowledge the providers of the Date Fruit Dataset for making the dataset publicly available, which enabled the experimental evaluation in this study. The author also acknowledges the technical support provided by the computing facilities used during data processing and experimentation. Any remaining errors are the sole responsibility of the author.

Conflicts of Interest

The author declares no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

[1] Y. Dimas Pratama and A. Salam, "Comparison of Data Normalization Techniques on KNN Classification Performance for Pima Indians Diabetes Dataset," *J. Appl. Informatics Comput.*, vol. 9, no. 3, pp. 693–706, 2025, doi: 10.30871/jaic.v9i3.9353.

[2] M. Yusran, K. Sadik, A. M. Soleh, and C. Suhaeni, "Effect of Feature Normalization and Distance Metrics on K-Nearest Neighbors Performance for Diabetes Disease Classification," *J. Math. Comput. Stat.*, vol. 8, no. 2, pp. 341–354, 2025, doi: 10.35580/jmathcos.v8i2.8012.

[3] S. Zhang, "Challenges in KNN Classification," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 10, pp. 4663–4675, 2022, doi: 10.1109/TKDE.2021.3049250.

[4] Y. Manzali, K. A. Barry, R. Flouchi, Y. Balouki, and M. Elfar, "A feature weighted K-nearest neighbor algorithm based on association rules," *J. Ambient Intell. Humaniz. Comput.*, vol. 15, no. 7, pp. 2995–3008, 2024, doi: 10.1007/s12652-024-04793-z.

[5] J. Manurung, H. Saragih, M. A. Prabukusumo, and E. A. Firdaus, "Optimizing the performance of the K-Nearest Neighbors algorithm using grid search and feature scaling to improve data classification accuracy," vol. 14, no. 2, pp. 260–268, 2025.

[6] I. Niño-Adan, I. Landa-Torres, E. Portillo, and D. Manjarres, "Influence of statistical feature normalisation methods on K-Nearest Neighbours and K-Means in the context of industry 4.0," *Eng. Appl. Artif. Intell.*, vol. 111, p. 104807, 2022, doi: https://doi.org/10.1016/j.engappai.2022.104807.

[7] M. Pagan, M. Zarlis, and A. Candra, "Investigating the impact of data scaling on the k-nearest neighbor algorithm," *Computer Science and Information Technologies*, vol. 4, no. 2. pp. 135–142, 2023, doi: 10.11591/csit.v4i2.pp135-142.

[8] I. ul Haq, D. M. Khan, M. Hamraz, N. Iqbal, A. Ali, and Z. Khan, "Optimal -k nearest neighbours based ensemble for classification and feature selection in chemometrics data," *Chemom. Intell. Lab. Syst.*, vol. 240, p. 104882, 2023, doi: https://doi.org/10.1016/j.chemolab.2023.104882.

[9] H. Vega-Huerta *et al.*, "K-Nearest Neighbors Model to Optimize Data Classification According to the Water Quality Index of the Upper Basin of the City of Huarmey," *Appl. Sci.*, vol. 15, no. 18, 2025, doi: 10.3390/app151810202.

[10] S. M. Gbashi, P. A. Adedeji, O. O. Olatunji, and N. Madushele, "Optimal feature selection for a weighted k-nearest neighbors for compound fault classification in wind turbine gearbox," *Results Eng.*, vol. 25, p. 103791, 2025, doi: https://doi.org/10.1016/j.rineng.2024.103791.

[11] Y. Liu, Y. Zhang, X. Wang, and X. Qu, "Evidential K-Nearest Neighbors with Cognitive-Inspired Feature Selection for High-Dimensional Data," *Big Data Cogn. Comput.*, vol. 9, no. 8, 2025, doi: 10.3390/bdcc9080202.

[12] M. Kim *et al.*, “Fault Detection Method via k-Nearest Neighbor Normalization and Weight Local Outlier Factor for Circulating Fluidized Bed Boiler with Multimode Process,” *Energies*, vol. 15, no. 17, 2022, doi: 10.3390/en15176146.

[13] S. Fedushko, M. Greguš, and R. Kulhánek, “Developing an Application for Articles Classification Using the KNN Algorithm,” *Procedia Comput. Sci.*, vol. 265, pp. 578–583, 2025, doi: https://doi.org/10.1016/j.procs.2025.07.222.

[14] A. Alsirhani, M. H. Siddiqi, A. M. Mostafa, M. Ezz, and A. A. Mahmoud, “A Novel Classification Model of Date Fruit Dataset Using Deep Transfer Learning,” *Electronics*, vol. 12, no. 3, 2023, doi: 10.3390/electronics12030665.

[15] K. Albarrak, Y. Gulzar, Y. Hamid, A. Mehmood, and A. B. Soomro, “A Deep Learning-Based Model for Date Fruit Classification,” *Sustainability*, vol. 14, no. 10, 2022, doi: 10.3390/su14106339.

[16] M. Pagan, M. Zarlis, and A. Candra, “Investigating the impact of data scaling on the k-nearest neighbor algorithm,” *Comput. Sci. Inf. Technol.*, vol. 4, no. 2, pp. 135–142, 2023, doi: 10.11591/csit.v4i2.pp135-142.

[17] E. Kartal, F. Çalışkan, B. B. Eskişehirli, and Z. Özen, “p-adic distance and k-Nearest Neighbor classification,” *Neurocomputing*, vol. 578, p. 127400, 2024, doi: https://doi.org/10.1016/j.neucom.2024.127400.

[18] C. Gong, J. Demmel, and Y. You, “Scalable Evidential K-Nearest Neighbor Classification on Big Data,” *IEEE Trans. Big Data*, vol. 10, no. 3, pp. 226–237, 2024, doi: 10.1109/TBDATA.2023.3327220.

[19] A. A. Amer, S. D. Ravana, and R. A. A. Habeeb, “Effective k-nearest neighbor models for data classification enhancement,” *J. Big Data*, vol. 12, no. 1, p. 86, 2025, doi: 10.1186/s40537-025-01137-2.

[20] A. A. S. R. de Sousa, J. da Silva Coelho, M. R. Machado, and M. Dutkiewicz, “Multiclass Supervised Machine Learning Algorithms Applied to Damage and Assessment Using Beam Dynamic Response,” *J. Vib. Eng. Technol.*, vol. 11, no. 6, pp. 2709–2731, 2023, doi: 10.1007/s42417-023-01072-7.

[21] A. Gyasi-Agyei, “A comparative assessment of machine learning models and algorithms for osteosarcoma cancer detection and classification,” *Healthc. Anal.*, vol. 7, p. 100380, 2025, doi: https://doi.org/10.1016/j.health.2024.100380.

[22] S. Szeghalmy and A. Fazekas, “A Comparative Study of the Use of Stratified Cross-Validation and Distribution-Balanced Stratified Cross-Validation in Imbalanced Learning.,” *Sensors (Basel).*, vol. 23, no. 4, Feb. 2023, doi: 10.3390/s23042333.

[23] M. B. - and D. B. B. -, “A Comprehensive Review of Cross-Validation Techniques in Machine Learning,” *Int. J. Sci. Technol.*, vol. 16, no. 1, pp. 1–4, 2025, doi: 10.71097/ijsat.v16.i1.1305.