Jurnal ELTIKOM :
Jurnal Teknik Elektro, Teknologi Informasi dan Komputer

Vol. 9, No. 1, June 2025, page. 57-68
ISSN 2598-3245 (Print), ISSN 2598-3288 (Online)
DOI: http://doi.org/10.31961/eltikom.v9i1.1476
Available online at http://eltikom.poliban.ac.id

PREDICTION OF TELKOMSEL 4G LTE CARD SALES USING THE K-NEAREST NEIGHBOR ALGORITHM

Alfiana Fontes Martins[1], Yasinta Oktaviana Legu Rema[1], Debora Chrisinta[1*], Alejandro Jr. V. Matute[2], Krisantus Jumarto Tey Seran[1]
[1)] Information Technology Program, Universitas Timor, Kefamenanu, Indonesia
[2)] College of Computer Studies, Laguna State Polytechnic University - Los Baños Campus, Laguna, Philippines
e-mail: alfianamartins89@gmail.com, rema.ivana@gmail.com, deborachrisinta@unimor.ac.id, alejandroventiro-matute@lspu.edu.ph, krisantusteyseran@unimor.ac.id

Received: 24 February 2025 – Revised: 22 April 2025 – Accepted: 1 May 2025

ABSTRACT

Accurate sales prediction is a critical challenge in business decision-making, as factors such as data imbalance, outliers, and overfitting may compromise the reliability of predictive models. This study aims to develop a precise model for predicting card sales using the K-Nearest Neighbor (KNN) algorithm and to offer recommendations for improving prediction quality by addressing issues related to data imbalance and overfitting. The KNN algorithm is applied to analyze a card sales dataset, with preprocessing steps that include detecting missing values, handling outliers, and converting the target attribute into a categorical format. The optimal value of k is identified using the elbow method to determine the model's best accuracy. Findings indicate that the KNN model with k = 1 achieves 100% accuracy, though it shows signs of overfitting, which may hinder its generalizability to new data. Handling outliers and transforming data contributed to improving the model's performance. However, to enhance robustness, further testing with different k values and the use of cross-validation are recommended. Moreover, balancing the dataset and incorporating external variables such as promotional activities or market trends could support more reliable future predictions.

Keywords*: card sales prediction, KNN, model accuracy.*

I. INTRODUCTION

THE advancement of information technology plays a vital role in modern life, as seen in its integration into daily activities. Among its fastest-growing sectors is communication technology. In 2023, data from the Indonesian Internet Service Providers Association (APJII) reported a surge in internet users, reaching 171 million. Each year, this number increases by 10.2 percent, or approximately 27 million users. The rapid growth in internet usage has driven greater demand for reliable telecommunication services, prompting mobile network providers to innovate and improve continuously to stay competitive. Recognizing this upward trend, telecommunications companies have identified it as a valuable business opportunity [1]. Telkomsel, one of Indonesia's largest mobile network operators, has established itself as a market leader. This success stems from Telkomsel's dedication to offering high-quality services, such as affordable starter packs, competitive call and SMS rates, wide network coverage, easy access to products, appealing promotions, and ongoing technological innovation. The company's ability to maintain excellent service standards while expanding its market share has contributed to strong customer loyalty and steady subscriber growth. Telkomsel also prioritizes service quality and network capacity, ensuring reliable connectivity for users across diverse regions [2]. As the market leader, Telkomsel continues to innovate, including the recent rollout of its 4G LTE (Long Term Evolution) service.

According to 2023 data on Telkomsel 4G LTE subscribers, the number of users increased consistently each month. The highest subscriber count was recorded in December, with 277,613 users, while the lowest was in January, with 98,674. This steady growth reflects a strong demand for Telkomsel's services and a broader reliance on mobile communication and internet connectivity. These figures

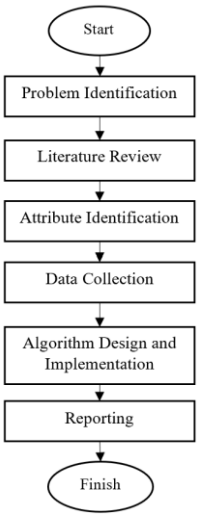


Figure 1. Research stages

highlight a substantial monthly rise in 4G LTE subscriptions. Given the upward trend, predictive analysis is essential to support continued sales growth in the coming years.

One method suitable for this analysis is data mining. This consistent increase in subscribers suggests a strong demand for Telkomsel's services, reflecting the growing reliance on mobile communication and data connectivity. involves analyzing large-scale datasets (big data) to uncover meaningful patterns [3], [4]. One commonly used classification technique in data mining is the K-Nearest Neighbor (KNN) algorithm, which classifies objects based on their proximity to the nearest training data points [5], [6], [7]. Due to its simplicity and effectiveness, KNN is often favored for predictive analysis, particularly when dealing with highly non-linear and complex data patterns. Its strengths include resilience to noisy training data and effectiveness in handling large datasets [8], [9]. Several previous studies have demonstrated the effectiveness of the KNN algorithm in predictive tasks across different domains. For instance, Choirun and Andri used KNN to predict pharmaceutical sales at Kimia Farma Atno Pharmacy in Palembang, achieving 100% accuracy [10]. Amalia applied KNN to forecast best-selling smartphone models, reaching an accuracy of 92.51% [11]. Similarly, WPR et al. predicted Unilever product sales using KNN, with an accuracy of 86.6% [12]. These studies support the applicability of KNN in sales prediction across various industries.

Building on the identified research gap and previous studies, this research aims to develop an accurate sales prediction model for Telkomsel 4G LTE starter packs using the KNN algorithm. Unlike earlier studies, this study introduces several novel elements. First, it addresses critical challenges often overlooked in prior work, such as data imbalance, outliers, and the risk of overfitting. Second, it implements improved data preprocessing techniques and hyperparameter tuning, specifically the selection of an optimal *k* value, to enhance model performance. Third, this study proposes a hybrid approach by integrating KNN with data balancing methods such as SMOTE (Synthetic Minority Oversampling Technique), promoting more equitable learning from imbalanced datasets. These innovations aim to produce a more robust, reliable, and generalizable prediction model suited to the dynamic and competitive telecommunications industry. By applying proper preprocessing and optimal *k* selection, this study seeks to improve the accuracy and practical relevance of KNN-based sales prediction in this sector.

II. RESEARCH METHOD

A. Research Stages

The research process (Figure 1) begins with problem identification, which focuses on accurately predicting the sales of Telkomsel 4G LTE cards in Kefamenanu City. This is followed by a literature review, which examines prior studies on the use of data mining techniques and the KNN algorithm in sales prediction. The aim is to understand the fundamental concepts, applied techniques, and the strengths and limitations of the KNN method. In the attribute identification stage, relevant variables for the sales prediction analysis are defined. These consist of two categories: target and predictor attributes. The

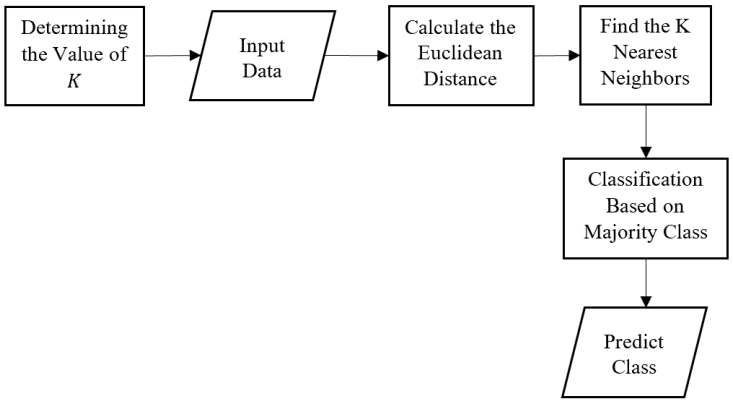


Figure 2. Stages of the KNN Algorithm

$$d(p,q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2} \quad (1)$$

target attribute has two classes—low and high. The low class corresponds to monthly sales of fewer than 1,000,000 units, while the high class refers to sales of 1,000,000 units or more. The predictor attributes include the number of AS, Simpati, Loop, and Halo cards sold each month. During the data collection stage, monthly sales data from 2020 to 2023 were gathered from a store in Kefamenanu City, East Nusa Tenggara Province. This dataset, sourced from internal company records, consists of historical sales of Telkomsel 4G LTE starter packs. The data were cleaned to remove duplicates, address missing values, and eliminate outliers. In the algorithm design and implementation stage, a predictive model was developed using the KNN algorithm and implemented in the system using the Python programming language. Finally, in the reporting stage, the research findings were compiled into a report detailing the methodology, experimental results, conclusions, and recommendations.

B. Stages of the KNN Algorithm

The KNN algorithm involves several key steps essential for building a predictive model (Figure 2). The process begins with selecting the optimal value of *K* by testing different values to determine which provides the best prediction accuracy. Next, the distance between the new data point and existing data points is calculated, typically using the Euclidean Distance formula (see (1)), which measures the straight-line distance between two points in a multi-dimensional space.

In (1), p and q represent two data points (vectors), each consisting of multiple attributes (features). For example, in a sales prediction context, p and q might represent two records containing values such as the number of cards sold, the month, or other relevant variables. The Euclidean formula computes how close these points are in the feature space.

After calculating the distances, the new data point is classified based on the majority class among its *K* nearest neighbors. This involves identifying the *K* closest points and assigning the most frequent class—such as “low” or “high” sales in the case of Telkomsel 4G LTE card predictions. Through these stages, the KNN algorithm predicts outcomes by referencing historical data and identifying the most similar patterns in the dataset.

C. Stages of KNN Algorithm Implementation

The KNN algorithm implementation process (Figure 3) begins with input data, which involves loading the dataset used for analysis. This dataset should include relevant information for prediction, such as Telkomsel 4G LTE card sales. The next step is preprocessing, which consists of two sub-stages: cleaning and transformation. In the cleaning stage, duplicate records are removed to avoid biased results. Missing values are then addressed through imputation or deletion, depending on the extent of missing data [13]. Outliers—values that deviate significantly from others—are identified and handled by either removing or adjusting them to prevent the model from being influenced by extreme, unrepresentative data. Following the cleaning process, categorical data in the target attribute (e.g., the "low" and "high" classes) is converted into numerical form using techniques such as label encoding or one-hot encoding, making the data compatible with the KNN algorithm [14].

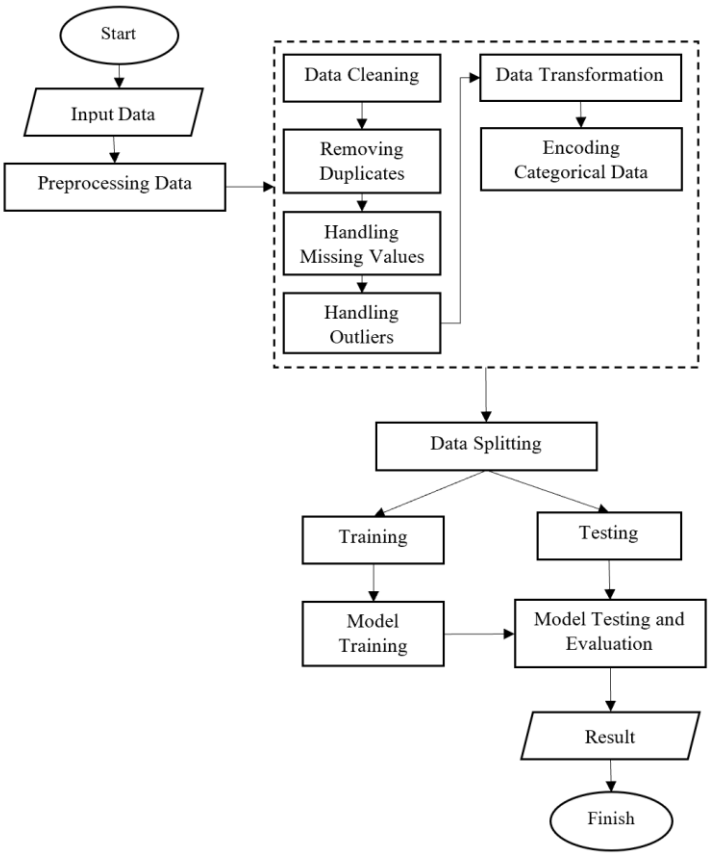


Figure 3. Stages of the KNN Algorithm Implementation

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$F1 - Score = \frac{2(Precision \times Recall)}{Precision + Recall} \tag{5}$$

Subsequently, the dataset is divided using an 80:20 split, where 80% is used for training the model (training set), and the remaining 20% is used for testing (test set) [15]. After the split, model training is conducted by applying the KNN algorithm to the training data. During this stage, the model learns to recognize patterns in the data and predict class labels based on the specified attributes. Once the model is trained, testing is carried out using the previously separated test data. The model's performance is then evaluated using several classification metrics. One common approach is the confusion matrix, which displays the number of correct and incorrect predictions compared to the actual class labels. Based on the confusion matrix (Table 1), four key evaluation metrics can be calculated: accuracy, precision, recall, and F1-score [16] using (1), (2), (3), and (4), respectively.

Selecting appropriate evaluation metrics—such as accuracy, precision, recall, and F1-score—is essential for a comprehensive assessment of the K-Nearest Neighbor (KNN) model's performance. Although accuracy is often used as a primary metric, it may be less informative when dealing with imbalanced datasets. In such cases, precision and recall help assess how accurately and thoroughly the model identifies specific classes, which is crucial when misclassification could affect business outcomes, such as inventory management or promotional strategies. The F1-score provides a balanced view between

TABLE 2
TRANSFORMATION OF THE TARGET ATTRIBUTE

index	Observation Object	Simpati	As	Loop	Halo	Sales Price	Description Before Label Encode	Description After Label Encode
0	1	62	48	14	17	2759000	High	0
1	2	14	3	2	2	881000	Low	1
2	3	10	19	5	4	763000	Low	1
3	4	31	39	9	4	2395000	High	0
4	5	38	32	31	6	3748000	High	0
5	6	45	52	20	7	3876000	High	0
6	7	49	20	15	4	4094000	High	0
7	8	70	26	13	11	4573000	High	1
8	9	9	2	1	1	690000	Low	0
9	10	57	21	13	2	5011000	High	0

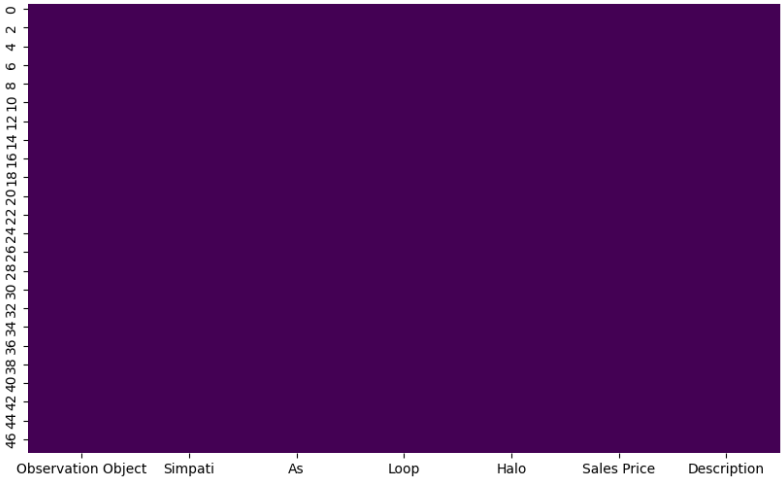


Figure 4. Results of Missing Value Detection

precision and recall, especially valuable when class distribution is uneven. Using all four metrics allows for a more objective and reliable evaluation of the KNN model's performance in predicting Telkomsel 4G LTE starter pack sales.

III. RESULTS AND DISCUSSION

A. Result

Based on the results of missing value detection in the dataset (Figure 4), the visualization appears entirely in purple. This indicates that there are no missing values in the dataset used for this study. The inspection was performed using a heatmap, where yellow typically represents the presence of missing values, and purple signifies complete data. Since no missing values are present, the dataset can proceed directly to the next preprocessing stage without requiring imputation or data removal. Previous studies have noted that missing values can impact the quality of predictive models, particularly in machine learning applications [17]. Common imputation techniques include using the mean, median, or predictive modeling approaches [18]. However, in this case, since the dataset is complete, the imputation stage is unnecessary, allowing the analysis to focus on data transformation and modeling. As a result, the integrity of the dataset remains intact, which supports improved prediction accuracy [19].

In the outlier detection stage using a boxplot, five data points were identified as outliers within the Simpati, As, and Loop attributes (Figure 5). Outliers are data points with values that significantly deviate from the rest of the dataset and can negatively affect model performance. To address this, the rows containing outliers were removed. After removal, a new boxplot was generated to confirm that no outliers remained. This step is essential, as unaddressed outliers can distort the predictive model and reduce its accuracy and reliability. Previous research has shown that removing outliers can enhance the performance of machine learning-based predictive models [20]. Additionally, proper outlier handling reduces data variability and enables the model to focus more effectively on relevant patterns [21].

After the data cleaning stage, the next step is to transform the target attribute into a categorical format (Table 2). This process involves converting the original numeric or textual values of the target attribute

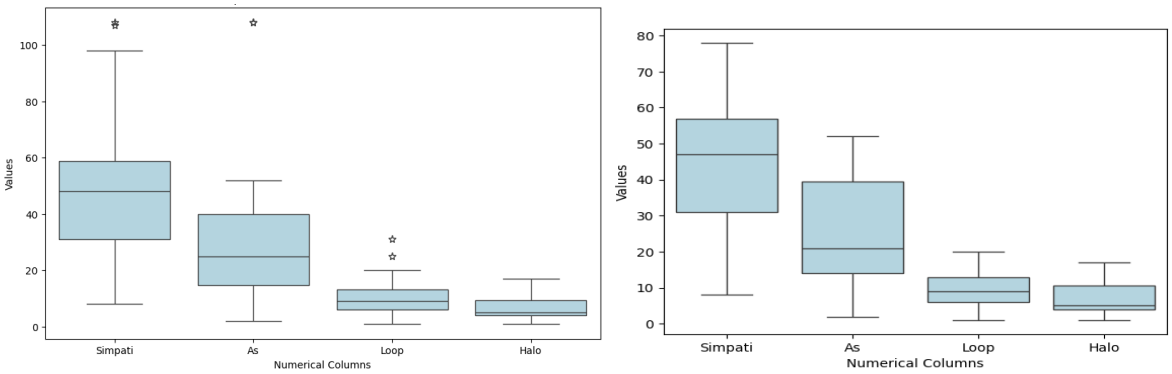


Figure 5. Results of Outlier Detection

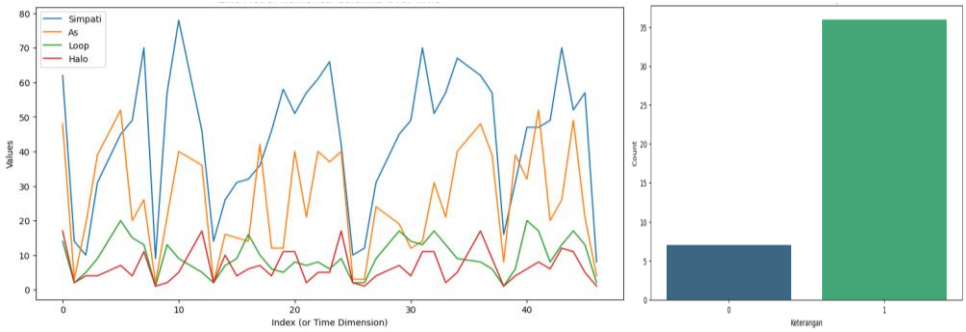


Figure 6. Card Sales Chart and Class Distribution in Sales Categories

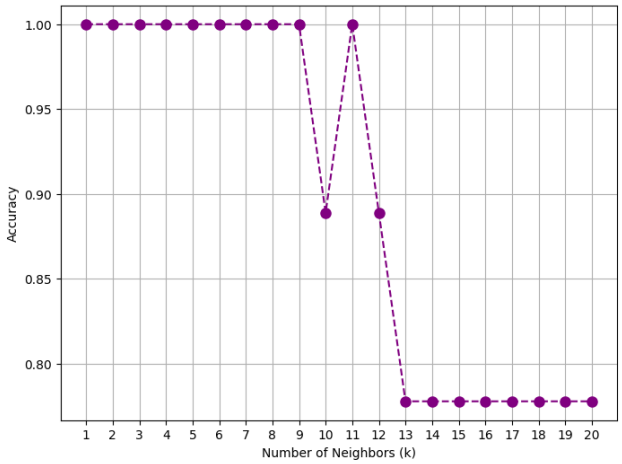


Figure 7. Elbow Method Graph for Determining Optimal k Value

into structured categories that are easier to analyze. Such transformation is essential in machine learning applications, especially for classification algorithms like KNN. By categorizing the target attribute, the model can more effectively classify data based on predefined classes. For instance, the target attribute may be divided into two categories, "low" and "high," representing sales levels based on a specific threshold. This categorization enhances the model's ability to recognize patterns and relationships between the predictor attributes and the classes within the target attribute. It also contributes to improved model performance, as categorical data is more appropriate for classification tasks and enables the algorithm to predict labels with greater accuracy.

The sales chart shows that Simpati cards have the highest sales among all card types, followed by As, Loop, and Halo cards (Figure 6). This suggests that Simpati cards are in higher demand, potentially due to factors such as pricing, promotions, or customer preferences favoring this product. Additionally, the chart reveals an imbalance in the sales categories, with the "high" category appearing more frequently than the "low" category. The high category refers to sales that exceed a specific threshold, while the low category includes sales with lower volumes. This imbalance indicates that most card sales fall into the higher sales range, while lower-volume sales are relatively infrequent. Such a pattern may result from

TABLE 3
DESCRIPTIVE STATISTICS

	Simpati	As	Loop	Halo
Min	8	2	1	1
Max	78	52	20	17
Mean	44	26	9	7
Count	43	43	43	43

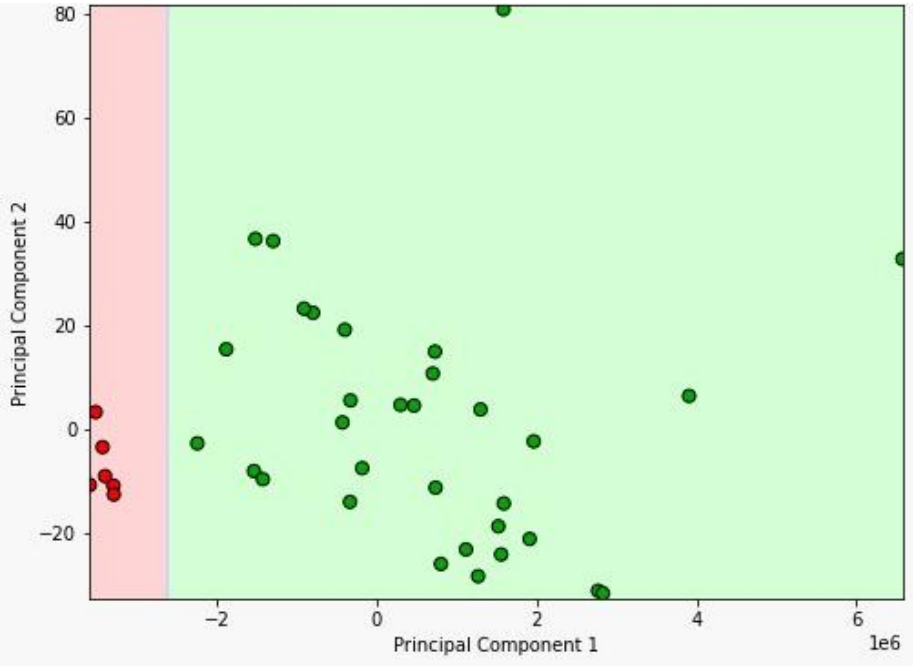


Figure 8. Decision Boundary of the KNN model

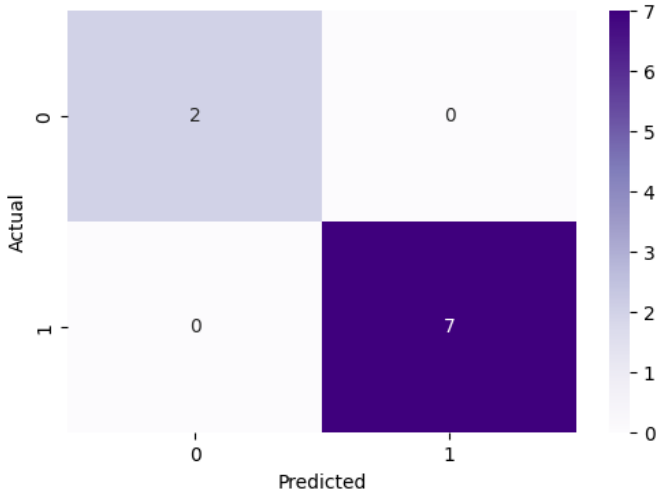
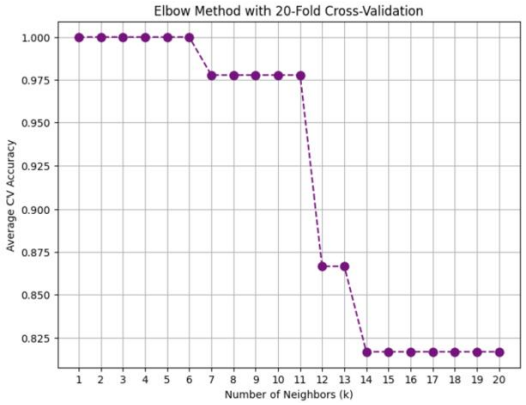


Figure 9. Confusion Matrix

effective promotional or marketing strategies that encourage higher-volume purchases. The imbalance between high and low sales categories is important to consider in the analysis, as it may influence the performance of the predictive model. Therefore, the model must be able to address this imbalance to produce accurate and representative predictions across both categories. Techniques such as over-sampling or undersampling may be employed to balance the data. Further details on card sales are presented in the descriptive statistics shown in Table 3.

The elbow method is used to determine the optimal number of neighbors (k) in the KNN algorithm (Figure 7). The graph shows that $k = 1$ yields the highest accuracy, reaching 1.00, indicating perfect classification at this value [22]. However, as k increases, the accuracy fluctuates and drops significantly after $k = 12$. According to Montesinos López et al., selecting a small k value such as 1 can lead to




```
Average accuracy for each k value:
k =  1 | Average CV Accuracy = 1.0000
k =  2 | Average CV Accuracy = 1.0000
k =  3 | Average CV Accuracy = 1.0000
k =  4 | Average CV Accuracy = 1.0000
k =  5 | Average CV Accuracy = 1.0000
k =  6 | Average CV Accuracy = 1.0000
k =  7 | Average CV Accuracy = 0.9778
k =  8 | Average CV Accuracy = 0.9778
k =  9 | Average CV Accuracy = 0.9778
k = 10 | Average CV Accuracy = 0.9778
k = 11 | Average CV Accuracy = 0.9778
k = 12 | Average CV Accuracy = 0.8667
k = 13 | Average CV Accuracy = 0.8667
k = 14 | Average CV Accuracy = 0.8167
k = 15 | Average CV Accuracy = 0.8167
k = 16 | Average CV Accuracy = 0.8167
k = 17 | Average CV Accuracy = 0.8167
k = 18 | Average CV Accuracy = 0.8167
k = 19 | Average CV Accuracy = 0.8167
k = 20 | Average CV Accuracy = 0.8167

Average of all accuracy (k = 1 to 20): 0.9169
```

Figure 10. Visualization Using 20-Fold Cross-Validation

overfitting, as the model becomes overly sensitive to the training data. In contrast, using a larger *k* can reduce the effect of noise but may introduce higher bias [23]. Nair and Kashyap also noted that choosing *k* greater than 1 can help reduce the influence of outliers, but this must be balanced with maintaining model accuracy [24]. Additionally, Lubis et al. emphasized that while the elbow method is effective in identifying the optimal *k*, it may be limited when applied to datasets with complex distributions and should therefore be supplemented with other validation techniques [22]. Although $k = 1$ provides the highest accuracy in this case, further evaluation is necessary to ensure that the model can generalize well to new data.

The graph displays the decision boundary of the KNN model with $k = 1$, using Principal Component Analysis (PCA) for dimensionality reduction (Figure 8). In the visualization, two classes are represented: red (class 0) and green (class 1). The green region indicates areas classified as class 1, while the red region represents areas classified as class 0. With $k = 1$, the model classifies each point based solely on its nearest neighbor, making its predictions highly sensitive to the surrounding data points. The distribution of data points shows that most belong to class 1, while class 0 is primarily concentrated in the lower-left region of the graph. Because the classification is influenced only by the closest data point, the model is prone to overfitting. PCA helps reduce the dimensionality of the dataset, enabling visualization of data distribution and the decision boundary across two principal components (Principal Component 1 and Principal Component 2). Based on the visualization, the KNN model with $k = 1$ effectively separates the classes, although a few misclassified points remain. To reduce overfitting and improve classification robustness against noise, a larger *k* value is generally recommended.

The confusion matrix (Figure 9) further illustrates the classification performance of the model. The results show 7 True Positives (TP), 2 True Negatives (TN), 0 False Positives (FP), and 0 False Negatives (FN). Based on these values, the accuracy, precision, recall, and F1-score all reached 100%. These results show a perfect balance between precision and recall. Overall, the model achieves ideal performance with no classification errors. However, further testing with a larger and more diverse dataset is necessary to ensure the model does not overfit the training data.

B. Handling Overfitting

Based on the Elbow Method visualization using 20-fold cross-validation (Figure 10), the modeling approach does not rely solely on a single *k* value that yields the highest accuracy. Instead, it evaluates the model's performance across all *k* values from 1 to 20. This strategy aims to provide a more comprehensive understanding of the stability and generalizability of the KNN model. By calculating the average accuracy across all *k* values, an overall accuracy of 0.9169 was achieved. This approach offers a fairer and more representative assessment of the model's overall performance and helps reduce the risk of overfitting that may arise when relying only on the best-performing *k* value. Thus, the average accuracy reflects the performance of all tested KNN configurations, resulting in a more robust and reliable predictive outcome. A study by Abriha et al. (2023) showed that depending solely on the *k* value with the highest accuracy during k-fold cross-validation can lead to performance overestimation, particularly in object detection using remote sensing data [25]. To mitigate such bias, evaluating the model across a range of *k* values offers a more stable and representative picture of model performance. This approach, implemented in this study through the Elbow Method and 20-fold cross-validation, produces a more

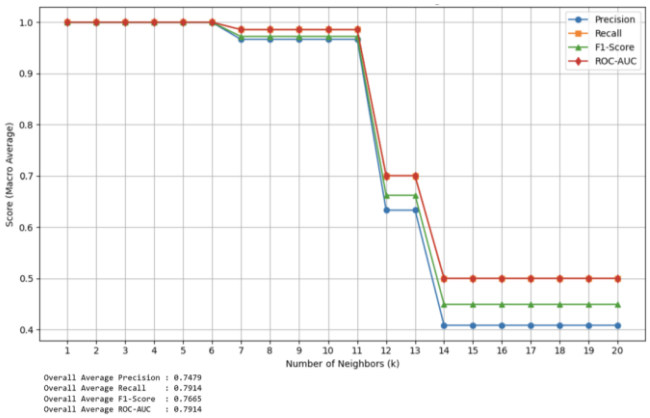


Figure 11. Performance of The KNN Model Based on Four Evaluation Metrics

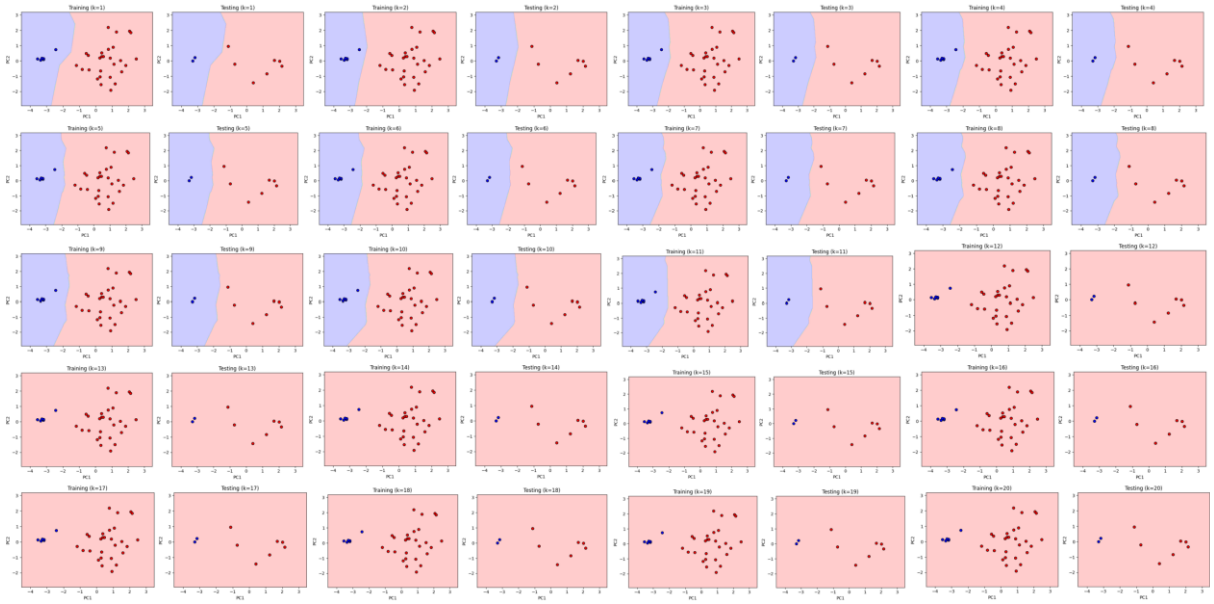


Figure 12. Development of Models with *k* value from 1 to 20

reliable average accuracy and reduces the risk of overfitting, thereby supporting more dependable predictive results.

Based on Figure 11, the graph illustrates the performance of the KNN model using four evaluation metrics—Precision, Recall, F1-Score, and ROC-AUC—across different values of *k*, ranging from 1 to 20. Overall, all metrics demonstrate strong performance (approaching 1.0) for *k* values between 1 and 11, indicating the model's robust classification capability within this range. However, beginning at *k* = 12, all metrics show a noticeable decline, with performance stabilizing at significantly lower levels beyond *k* = 14. The average values for each metric offer a comprehensive evaluation of the model's performance across the entire range of *k* values. The overall average precision is 0.7479, recall is 0.7914, F1-score is 0.7665, and ROC-AUC is 0.7914. These values suggest that despite the performance drop at higher *k* values, the KNN model generally performs satisfactorily. Among the metrics, recall and ROC-AUC stand out, indicating the model's reliability in identifying positive classes and effectively distinguishing between different classes. Thus, these average scores are important for assessing the model's stability and overall effectiveness across varying *k* values, even when some performance degradation is observed.

Based on Figure 12, provides a visualization of the training and testing phases of the KNN model using 20-fold cross-validation. Each subplot illustrates the data partitioning for individual folds, displaying the corresponding decision boundaries formed by the KNN model. The colored regions represent the model's classification predictions, while the blue and red dots indicate two distinct classes. Across the 20 folds, it can be observed that the decision boundaries in the initial folds are well-defined and closely follow the data distribution, indicating strong classification performance. However, in the later folds—particularly toward the end—the decision regions become more homogeneous (dominated

by a single color), suggesting a reduced ability to differentiate between classes. This pattern aligns with the earlier performance metrics, which show a decline at higher values of *k*. Overall, this visualization reinforces the conclusion that the KNN model performs accurately and consistently in the earlier folds, but its effectiveness decreases when the number of neighbors increases or when data distribution within the folds is less balanced. The use of 20-fold cross-validation offers a comprehensive view of the model's performance variability, underscoring the importance of selecting appropriate parameters—such as the value of *k*—and ensuring balanced data distribution during training and evaluation.

C. Discussion

In this study, the data preprocessing stage—particularly the detection of missing values—showed that the dataset contained no missing entries, thereby eliminating the need for imputation or data removal. This finding aligns with previous research, which suggests that missing values can reduce the quality of predictive models, particularly in machine learning applications [26]. However, despite the absence of missing values, it is important to acknowledge that the dataset may still contain limitations, such as potential measurement errors that were not captured in the analysis. Guida emphasized that measurement errors may go undetected and can impact model results even when no missing values are present [27].

The handling of outliers in this study also produced positive results. Data points identified as outliers were removed to improve model accuracy, consistent with the theory proposed by Andersson et al., who argue that removing outliers can enhance model performance by minimizing distortion caused by extreme values [28]. However, it is also important to consider alternative methods for managing outliers, such as applying data transformation techniques or adopting modeling approaches that are more robust to such anomalies. A more dynamic strategy for detecting and addressing outliers may help ensure that valuable information is not lost during preprocessing.

Additionally, in applying the KNN algorithm, selecting the optimal *k* value plays a crucial role in producing an accurate model while avoiding overfitting. This study found that $k = 1$ yielded very high accuracy but carried the risk of overfitting, as explained by Beckmann et al. [29]. While using a larger *k* can reduce this risk, Zhang et al. noted that increasing the *k* value may lead to higher bias and reduced accuracy if the dataset is not evenly distributed [30]. Therefore, future research could expand the analysis by combining cross-validation techniques with more refined parameter tuning to identify the optimal *k* value and develop a model that generalizes well across larger and more diverse datasets.

In addition, the evaluation of the KNN model using 20-fold cross-validation and the Elbow Method further underscores the importance of considering model stability across different parameter configurations. Rather than relying solely on the *k* value that yields the highest accuracy, this study averaged performance across *k* values from 1 to 20, resulting in an overall accuracy of 0.9169. This approach provides a more representative measure of the model's general performance and reduces the risk of overfitting, as supported by Abriha et al. [25]. Moreover, the analysis of multiple evaluation metrics—including precision, recall, F1-score, and ROC-AUC—showed that the model performed optimally when *k* values ranged from 1 to 11, with performance declining beyond $k = 12$. Visualizations of decision boundaries across folds revealed that while the model classified data effectively in the early folds, its ability to distinguish between classes diminished in later ones. These findings highlight the importance of combining cross-validation with comprehensive metric analysis to develop a robust and generalizable KNN model.

The findings of this study also offer valuable business implications, particularly for telecommunications companies such as Telkomsel. By leveraging the average accuracy obtained from KNN models across various *k* values, businesses can make more reliable and data-driven decisions. In the context of inventory planning, accurate sales predictions enable Telkomsel to optimize stock levels for 4G LTE starter packs, minimizing excess inventory while ensuring availability in high-demand areas. From a marketing perspective, understanding purchasing trends allows for targeted campaigns tailored to specific customer segments, thereby increasing engagement and conversion rates. Additionally, retail decision-making can be enhanced by identifying regions or outlets with varying demand levels, allowing for more strategic product placement and promotional activities. Ultimately, the model provides a foundation for more efficient operational strategies, cost reduction, and improved customer satisfaction.

One important aspect to consider in this study is the inherent limitations of the KNN algorithm, which may affect the interpretation of the results. As a distance-based classifier, KNN is highly sensitive to the scale and distribution of input features, making it vulnerable to noise, irrelevant attributes, and high-

dimensional data. Its performance also tends to degrade with large datasets due to the increased computational load during prediction. Additionally, KNN lacks a built-in mechanism for handling class imbalance, which may lead to biased predictions favoring the majority class. In this study, various preprocessing techniques—including normalization, feature selection, and data balancing (e.g., SMOTE)—were applied to address these issues. However, the limitations of the KNN algorithm remain relevant and may affect the model's ability to generalize to unseen data or across different business contexts.

Future research could extend this study by investigating alternative machine learning algorithms that offer improved performance and robustness compared to KNN, such as Random Forest, Support Vector Machines (SVM), or Gradient Boosting methods. These algorithms are generally more effective in handling high-dimensional and imbalanced datasets and often provide better scalability and interpretability. Moreover, incorporating external variables—such as market trends, promotional campaigns, seasonal fluctuations, and competitor actions—could enhance the model's predictive capability by offering a more comprehensive perspective on the factors influencing sales. Such integration would support the development of a more dynamic and adaptive predictive model, better suited to real-world business environments.

IV. CONCLUSION

Based on the results of this study, the KNN model with $k = 1$ achieved exceptionally high accuracy (100%), indicating perfect classification within the dataset used. However, while this level of accuracy is ideal, using such a small k value may lead to overfitting, making the model overly sensitive to the training data and potentially less effective when applied to new data. Therefore, further testing with a range of k values is necessary to strike a balance between accuracy and the model's ability to generalize to more complex or unseen data. For future card sales predictions, enhancing the model to reduce the risk of overfitting is recommended. This can be achieved by using larger k values and incorporating cross-validation techniques. The application of the Elbow Method with 20-fold cross-validation in this study demonstrated that averaging accuracy across all k values provides a more reliable estimate of overall model performance. This approach effectively minimized overfitting and yielded more stable prediction outcomes compared to relying solely on the best-performing k value. Additionally, testing the model on a larger and more diverse dataset is essential to ensure robustness and predictive accuracy under varying market conditions. Implementing data balancing techniques to address class imbalance in sales categories, along with incorporating external variables—such as promotional activities or market trends—could further enhance the quality and applicability of future sales predictions.

REFERENCES

[1] A. Ardiansyah, "Pengaruh Kemudahan dan Keamanan Data Pribadi Terhadap Minat Menggunakan Dompet Digital (E-Wallet) Linkaja (Studi Kasus Pada Mahasiswa Fakultas Syariah dan Ekonomi Islam Tahun 2017-2019)," IAIN Syekh Nurjati Cirebon, 2021. Accessed: Feb. 11, 2025. [Online]. Available: http://repository.syekhnurjati.ac.id/5202/

[2] S. Hutajulu, W. Dhewanto, and E. A. Prasetio, "Two scenarios for 5G deployment in Indonesia," *Technol Forecast Soc Change*, vol. 160, p. 120221, 2020, doi: 10.1016/j.techfore.2020.120221.

[3] D. Chrisinta and J. E. Simarmata, "Eksplorasi Teknik Web Scraping pada Data Mining: Pendekatan Pencarian Data Berbasis Python," *Faktor Exacta*, vol. 17, no. 1, pp. 1979–276, May 2024, doi: 10.30998/FAKTOREXACTA.V17I1.22393.

[4] A. Isnain, … J. S.-I. (Indonesian J., and undefined 2021, "Implementation of K-Nearest Neighbor (K-NN) algorithm for public sentiment analysis of online learning," *journal.ugm.ac.idAR Isnain, J Supriyanto, MP KharismaIJCCS (Indonesian Journal of Computing and Cybernetics Systems), 2021•journal.ugm.ac.id*, vol. 15, no. 2, pp. 121–130, 2021, doi: 10.22146/ijccs.65176.

[5] I. Triguero, D. García-Gil, J. Maillo, J. Luengo, S. García, and F. Herrera, "Transforming big data into smart data: An insight on the use of the k-nearest neighbors algorithm to obtain quality data," *Wiley Interdiscip Rev Data Min Knowl Discov*, vol. 9, no. 2, Mar. 2019, doi: 10.1002/WIDM.1289.

[6] P. Cunningham and S. J. Delany, "K-Nearest Neighbour Classifiers-A Tutorial," *ACM Comput Surv*, vol. 54, no. 6, Jul. 2021, doi: 10.1145/3459665.

[7] S. Ayyad, A. Saleh, L. L.- Biosystems, and undefined 2019, "Gene expression cancer classification using modified K-Nearest Neighbors technique," *Elsevier*, Accessed: Mar. 08, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0303264718302685

[8] H. Said, N. Matondang, H. I.-Techno. Com, and undefined 2022, "Penerapan Algoritma K-Nearest Neighbor Untuk Memprediksi Kualitas Air yang Dapat Dikonsumsi," *academia.eduH Said, NH Matondang, HN IrmandaTechno. Com, 2022•academia.edu*, Accessed: Mar. 08, 2025. [Online]. Available: https://www.academia.edu/download/89314820/2927.pdf

[9] I. Nikmatun, I. W.-J. Simetris, and undefined 2019, "Implementasi data mining untuk klasifikasi masa studi mahasiswa menggunakan algoritma K-Nearest Neighbor," *academia.eduIA Nikmatun, I WaspadaJurnal Simetris, 2019•academia.edu*, Accessed: Mar. 08, 2025. [Online]. Available: https://www.academia.edu/download/103513773/304201835.pdf

[10] A. Choirun and A. Andri, "Penerapan Algoritma K-Nearest Neighbor Untuk Prediksi Penjualan Obat Pada Apotek Kimia Farma Atmo Palembang," Universitas Bina Darma, 2020.

[11] Y. R. Amalia, "Penerapan data Mining untuk Prediksi Penjualan Produk Elektronik Terlaris Menggunakan Metode K-Nearest Neighbor," Universitas Islam Negeri Raden Fatah, 2018. Accessed: Feb. 19, 2025. [Online]. Available:

[12] A. A. WPR, F. Rozi, and F. Sukmana, "Prediksi Penjualan Produk Unilever Menggunakan Metode K-Nearest Neighbor," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 6, no. 1, pp. 155–160, Jun. 2021, doi: 10.29100/JIPI.V6I1.1910.
[13] D. Chrisinta and J. E. Simarmata, "Comparative Study of Support Vector Machine and Naive Bayes for Sentiment Analysis on Lecturer Performance," *Journal of Research in Mathematics Trends and Technology*, vol. 5, no. 1, pp. 1–7, 2023, doi: 10.32734/jormtt.v5i1.
[14] J. E. Simarmata, G. W. Weber, and D. Chrisinta, "Performance Evaluation of Classification Methods on Big Data: Decision Trees, Naive Bayes, K-Nearest Neighbors, and Support Vector Machines," *Jurnal Matematika, Statistika dan Komputasi*, vol. 20, no. 3, pp. 623–638, 2024, doi: 10.20956/j.v20i3.32970.
[15] A. Neonub, Y. R. L. Oktaviana, and D. Chrisinta, "Implementasi Algoritma Naive Bayes Pada Data Ulasan Mahasiswa Tentang Sarana dan Prasarana Kampus," *Prosiding Seminar Nasional Sains dan Teknologi "SainTek,"* vol. 1, no. 2, pp. 206–212, 2024, Accessed: Feb. 11, 2025.
[16] D. Chrisinta and J. E. Simarmata, "Analisis Sentimen Penilaian Masyarakat Terhadap Pejabat Publik Menggunakan Algoritma Naïve Bayes Classifier," *Komputika: Jurnal Sistem Komputer*, vol. 12, no. 1, pp. 93–101, 2023, doi: 10.34010/komputika.v12i1.9638.
[17] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona, "A survey on missing data in machine learning," *J Big Data*, vol. 8, no. 1, pp. 1–37, Dec. 2021, doi: 10.1186/S40537-021-00516-9.
[18] D. Bertsimas, C. Pawlowski, and Y. D. Zhuo, "From predictive methods to missing data imputation: an optimization approach," *Journal of Machine Learning Research*, vol. 18, no. 196, pp. 1–39, 2018, Accessed: Feb. 11, 2025.
[19] L. Theodorakopoulos, A. Theodoropoulou, and Y. Stamatiou, "A state-of-the-art review in big data management engineering: Real-life case studies, challenges, and future research directions," *Eng*, vol. 5, no. 3, pp. 1266–1297, 2024, doi: 10.3390/eng5030068.
[20] G. Jesus, A. Casimiro, and A. Oliveira, "Using Machine Learning for Dependable Outlier Detection in Environmental Monitoring Systems," *ACM Transactions on Cyber-Physical Systems*, vol. 5, no. 3, pp. 1–30, Jul. 2021, doi: 10.1145/3445812.
[21] H. Aguinis, R. K. Gottfredson, and H. Joo, "Best-practice recommendations for defining, identifying, and handling outliers," *Organ Res Methods*, vol. 16, no. 2, pp. 270–301, Apr. 2013, doi: 10.1177/1094428112470848.
[22] A. Lubis, Y. Irawan, J. Junadhi, and S. Defit, "Leveraging K-Nearest Neighbors with SMOTE and boosting techniques for data imbalance and accuracy improvement," *Journal of Applied Data Sciences*, vol. 5, no. 4, pp. 1625–1638, 2024, doi: 10.47738/jads.v5i4.343.
[23] O. A. Montesinos López, A. Montesinos López, and J. Crossa, *Overfitting, model tuning, and evaluation of prediction performance*. Springer International Publishing, 2022. doi: 10.1007/978-3-030-89010-0_4.
[24] P. Nair and I. Kashyap, "Hybrid pre-processing technique for handling imbalanced data and detecting outliers for KNN classifier," in *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, 2019, pp. 460–464. doi: 10.1109/COMITCon.2019.8862250.
[25] D. Abriha, P. Srivastava, and S. Szabó, "Smaller is Better? Unduly Nice Accuracy Assessments in Roof Detection Using Remote Sensing Data With Machine Learning And K-Fold Cross-Validation," *Heliyon*, vol. 9, no. 3, pp. 1–17, 2023, doi: 10.1016/j.heliyon.2023.e14045.
[26] J. Josse, J. M. Chen, N. Prost, G. Varoquaux, and E. Scornet, "On the consistency of supervised learning with missing values," *Statistical Papers*, vol. 65, no. 9, pp. 5447–5479, Dec. 2024, doi: 10.1007/s00362-024-01550-4.
[27] R. D. Guida *et al.*, "Non-targeted UHPLC-MS metabolomic data processing methods: a comparative investigation of normalisation, missing value imputation, transformation and scaling," *Metabolomics*, vol. 12, no. 5, pp. 1–14, May 2016, doi: 10.1007/s11306-016-1030-9.
[28] J. L. R. Andersson, M. S. Graham, and E. Zsoldos, "Incorporating outlier detection and replacement into a non-parametric framework for movement and distortion correction of diffusion MR images," *Neuroimage*, vol. 141, pp. 556–572, 2016, doi: 10.1016/j.neuroimage.2016.06.058.
[29] M. Beckmann, N. F. F. Ebecken, and B. S. P. De Lima, "A KNN undersampling approach for data balancing," *Journal of Intelligent Learning Systems and Applications*, vol. 7, no. 4, pp. 104–116, 2015, doi: 10.4236/jilsa.2015.74010.
[30] S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, "Efficient kNN classification with different numbers of nearest neighbors," in *IEEE transactions on neural networks and learning systems*, 2017, pp. 1774–1785. doi: 10.1109/TNNLS.2017.2673241.