Indonesian Journal of Electronics, Electromedical Engineering, and Medical Informatics e-ISSN: 2656-8624
Homepage: https://ijeeemi.org/; Vol. 7, No. 2, pp. 358-369, May 2025

RESEARCH PAPER

OPEN ACCES

Enhancing Imbalanced Data Handling Using MWMOTE and K-Means Clustering

Meida Cahyo Untoro

Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia

ABSTRACT

In machine learning and data mining, dataset quality plays a critical role in model performance. One major challenge is data imbalance, where one class has significantly fewer instances than the other. This imbalance leads to biased models that favor the majority class, negatively impacting the predictive accuracy for minority class instances. This issue is particularly problematic in applications such as medical diagnosis, fraud detection, and other high-stakes fields. This study aims to address data imbalance by employing the MWMOTE (Majority Weighted Minority Oversampling Technique) method, enhanced with K-Means Clustering. The primary goal is to improve the spatial distribution of synthetic samples for the minority class, thus enhancing model performance and generalization. By combining K-Means clustering with MWMOTE, the method generates more well-separated synthetic samples, improving class separation compared to traditional oversampling techniques. The MWMOTE + K-Means approach follows three key steps: 1) K-Means clustering of the minority class to group similar instances, 2) generation of synthetic samples within these clusters to maintain the class structure, and 3) weighting the minority class samples based on proximity to the decision boundary. The optimal number of clusters is determined using the Silhouette Score to ensure high-quality clustering. Experimental results on 10 datasets demonstrate significant performance improvements. The proposed method increases precision by 10%, recall by 40%, and F-measure by 40%, compared to the baseline accuracy of 70%. Although the clustering step introduces a slight increase in computational cost, the improvements in classification metrics validate the effectiveness of the MWMOTE + K-Means approach for handling imbalanced data. In conclusion, the MWMOTE + K-Means method effectively addresses class imbalance by generating well-distributed synthetic samples, leading to improved model performance. Future work could focus on optimizing computational efficiency and comparing this approach with other advanced oversampling techniques.

PAPER HISTORY

Received Feb. 27, 2025
Accepted April 17, 2025
Published May 18, 2025

KEYWORDS

Clustering;
Imbalance;
K-Means;
MWMOTE;
Oversampling;

AUTHOR EMAIL

cahyo.untoro@if.itera.ac.id

1. INTRODUCTION

One of the most significant challenges in machine learning and data mining is the issue of data imbalance, where the distribution of class instances within a dataset is highly uneven [1]. In such datasets, one class typically the majority class dominates [2], while the minority class is underrepresented [3]. This imbalance often leads to biased machine learning models, which favor the majority class [4], resulting in poor performance when tasked with classifying instances from the minority class [5]. Such biased models are particularly concerning in critical domains such as medical diagnosis [6] and fraud detection [7], where accurately identifying rare diseases or detecting fraudulent activities is essential [8]. In these high-stakes applications [9], the failure to classify minority instances (which often represent rare but important events) can have serious consequences [10].

Traditional machine learning models tend to struggle with these imbalanced datasets because they are often optimized for accuracy [11], which can be misleading when the data is imbalanced [12]. A model that correctly predicts the majority class but fails to classify the minority class is still considered accurate according to conventional metrics [13]. This misalignment between accuracy and meaningful performance highlights a crucial challenge in imbalanced data scenarios, which needs to be addressed [14].

To overcome class imbalance, various techniques have been proposed, which can be broadly categorized into two approaches: data-level methods and algorithm-level modifications [15]. Data-level methods, such as oversampling and undersampling, aim to manipulate the dataset to balance class distribution [16]. SMOTE (Synthetic Minority Over-sampling Technique) is one widely used oversampling method that generates synthetic instances in the feature space to help the model learn more representative boundaries for the minority class [16].

Corresponding author: Meida Cahyo Untoro, cahyo.untoro@if.itera.ac.id, Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.69

However, these methods have limitations. For example, oversampling techniques can lead to overfitting, where synthetic samples become too similar to the original data, reducing the model's ability to generalize to unseen instances [17]. Furthermore, generating synthetic instances near the decision boundary often results in poor class separation, particularly when these instances lack diversity in the feature space [18].

In response to these issues, more advanced methods have emerged, such as the MWMOTE (Majority Weighted Minority Oversampling Technique) approach. MWMOTE prioritizes generating synthetic samples based on the weight of minority class instances, particularly focusing on the "hard-to-learn" instances [19]. This weighted approach determines which minority class samples are more challenging to classify, prioritizing them when generating synthetic instances [20]. While MWMOTE improves class balance and classification accuracy, it still faces challenges, particularly regarding the generation of synthetic instances that are too close to the decision boundaries, which can impede class separation and reduce model performance [13].

Despite the progress made with techniques like SMOTE and MWMOTE, a significant research gap remains in improving the spatial distribution of synthetic instances [4]. Many existing methods, including MWMOTE, struggle to generate synthetic samples that are effectively spread across the feature space, resulting in poor separation between classes. When synthetic samples are clustered too closely around the decision boundary, it impairs the model's ability to make accurate predictions, especially on unseen data [21].

Additionally, while clustering methods have been proposed to improve sample distribution, few studies integrate these techniques into the oversampling process itself [22]. This gap offers an opportunity to explore how clustering methods, such as K-Means clustering, can be used to better distribute synthetic instances across the feature space, reducing overlap and improving class separation [23].

This study proposes a hybrid method called MWMOTE + K-Means, which integrates MWMOTE with K-Means clustering to improve the spatial distribution of synthetic instances. The approach starts by applying K-Means clustering to group the minority class instances into clusters. These clusters help identify the regions in the feature space where synthetic instances should be generated. By grouping minority class instances meaningfully, synthetic samples are distributed more evenly, reducing overlap with decision boundaries [24].

The Silhouette Score is used to determine the optimal number of clusters, maximizing intra-cluster similarity and inter-cluster separation [25]. This ensures that the clusters formed by K-Means represent coherent groups of minority class instances. After clustering, MWMOTE is applied to generate synthetic instances within each cluster, prioritizing the most challenging-to-learn instances. This process produces synthetic samples that are both well-distributed and representative of the more difficult instances of the minority class.

The MWMOTE + K-Means approach is implemented in Python using the Jupyter Notebook environment and evaluated across several benchmark datasets. Preliminary results indicate that this hybrid method improves classification precision by up to 10%, recall by 40%, and F-measure by 40%, compared to baseline models.

This study makes several significant contributions to the field of data mining and machine learning, particularly in the context of imbalanced datasets. First, it proposes the novel integration of MWMOTE with K-Means clustering to enhance the spatial distribution of synthetic samples. This integration reduces class overlap and improves class separation, allowing the model to distinguish better between majority and minority classes. By evenly distributing synthetic samples, the method overcomes a key drawback of traditional oversampling techniques, which often generate samples too close to the decision boundary.

Second, the research introduces a hybrid oversampling technique that combines the strengths of MWMOTE and K-Means clustering. This approach provides a more robust solution for imbalanced datasets by using K-Means to cluster minority class instances before applying oversampling. This combined approach improves model training accuracy, making it highly effective in scenarios where class imbalance distorts model performance.

Additionally, the method is evaluated on benchmark datasets, showing substantial improvements in precision, recall, and F-measure. These improvements emphasize the effectiveness of the MWMOTE + K-Means approach in providing more accurate and reliable predictions.

2. MATERIALS AND METHOD

This section describes the materials and methods used to evaluate the performance of the MWMOTE + K-Means method in addressing the class imbalance problem. The methodology includes the datasets used, data collection process, preprocessing steps, and statistical analysis techniques employed to assess the effectiveness of this hybrid oversampling technique that combines MWMOTE (Majority Weighted Minority Oversampling Technique) with K-Means clustering.

A. Dataset

A dataset is a structured collection of data that has been gathered over time to extract meaningful insights and support decision-making processes [26]. In this study, the dataset is utilized to assess the performance of an oversampling technique that integrates K-Means clustering. The primary objective is to address the class imbalance issue by generating synthetic data for the minority class, thereby improving classification accuracy [27]. The oversampling process involves applying K-Means clustering to redistribute synthetic samples in a way that enhances the representation of minority class

Corresponding author: Meida Cahyo Untoro, cahyo.untoro@if.itera.ac.id, Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.69

instances [28]. For this purpose, 10 datasets are selected from publicly available repositories (**Table 1**), including the Kaggle Dataset Repository and the UCI Machine Learning Repository [29].

B. Data Collection

The datasets utilized in this study were gathered from Kaggle and the UCI Machine Learning Repository, two widely recognized platforms that offer publicly accessible datasets with comprehensive metadata. This metadata includes information such as the number of instances, features, and class distributions, all of which are essential for evaluating machine learning algorithms [30]. Each dataset includes labeled instances with clearly defined majority and minority classes [15]. The imbalance ratio of the classes is explicitly reported for each dataset, providing valuable context for assessing the effectiveness of different oversampling techniques. Some datasets had predefined training and testing splits, while others were manually divided into an 80% training and 20% testing split to ensure consistency in model evaluation [31].

Table 1. Dataset repository

Dataset	attributes	Qty	Majority	Minority	IR
Abalone	8	730	689	41	94:6
Breast	10	105	69	36	66:34
Ecolli	8	335	258	77	77:23
Glass	10	213	162	51	76:24
Page-Blocks	11	5472	5241	231	96:4
Robot	25	5455	4301	1154	79:21
Satimage	37	6434	4398	2036	68:32
Segment	20	2309	1979	330	86:14
Wine	14	177	129	48	73:27
Yeast	9	1483	1180	303	70:20

One of the key datasets used in this study is the Wine Quality dataset, which contains 177 samples divided into two distinct classes: Red Wine and White Wine. This dataset exhibits significant class imbalance, with White Wine making up 72% of the instances, while Red Wine constitutes only 27%. Such an imbalance can lead to biased models that favor the majority class, resulting in poor performance when recognizing minority class instances [32]. To mitigate this issue, the class distribution was carefully analyzed during preprocessing [33]. The oversampling technique was then applied to generate synthetic samples, improving the balance between the classes and allowing for more accurate classification.

This process aims to enhance model performance and provide a more precise evaluation of classification algorithms when working with imbalanced datasets [34]. The varying levels of improvement observed across the datasets in this study can be attributed to differences in dataset characteristics, such as the number of classes, dimensionality, and the degree of class imbalance. Datasets with well-separated class distributions, like Wine and Segment, showed consistent improvements in classification accuracy and F-measure after oversampling [35]. On the other hand, datasets like Abalone and Robot, which have high overlap between classes or high-dimensional features, demonstrated only marginal or even negative improvements from oversampling. This suggests that the effectiveness of clustering-based oversampling methods, such as MWMOTE + K-Means, is sensitive to the inherent structure of the dataset [17].

C. Data Processing Oversampling

The technique of adding minority class data aims to balance the dataset by ensuring that the number of minority class instances is equal to the number of majority class instances. The illustration in Fig. **1** demonstrates the Oversampling method, where the dataset consists of two labels: Label 1 represents the majority class, while Label 0 represents the minority class. To address class imbalance, Label 0 undergoes an Oversampling process, generating synthetic data to match the number of majority class instances labeled as 1.

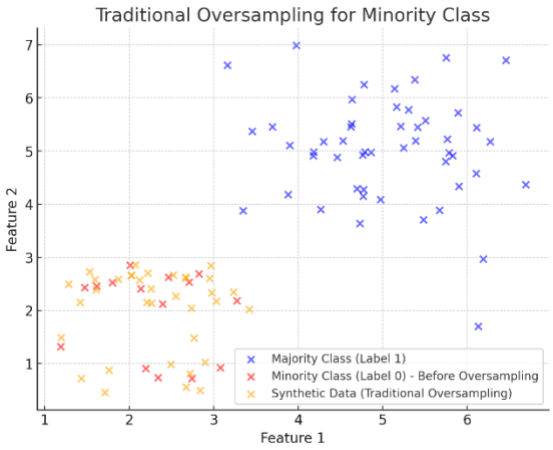

Fig. 1. Traditional Oversampling

To visualize and evaluate the quality of synthetic data generated via MWMOTE + K-Means (Fig 2), dimensionality reduction techniques such as PCA and t-SNE were employed. Visualizations showed that the synthetic instances tended to cluster cohesively within minority class regions, reducing the risk of overlapping with the majority class. For datasets like Wine and Yeast, the visual spread confirmed enhanced class separation. These insights suggest that the integration of K-Means aids in spatially meaningful data generation, thereby preserving class boundaries.

In this study, multiple datasets are tested using three different approaches: MWMOTE Oversampling, K-Means-enhanced MWMOTE, and a model without Oversampling. The synthetic data generation in the Oversampling process is conducted using a K-Means clustering approach, where new data points are

Corresponding author: Meida Cahyo Untoro, cahyo.untoro@if.itera.ac.id, Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.69

generated based on the average values of existing data clusters. This clustering-based technique ensures that the synthetic samples are well-distributed within the minority class, improving the overall balance of the dataset while preserving its underlying structure [17].

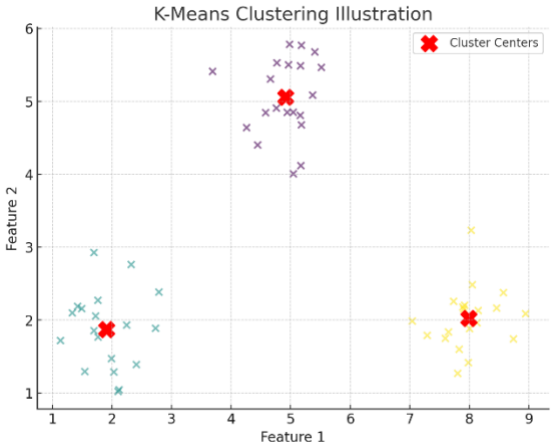

Fig 2. K-Means Clustering

Cluster numbers for K-Means were optimized using the Elbow Method and Silhouette Score. Results indicated that while 2 clusters yielded optimal separation in most datasets, some benefited from 4 or 9 clusters. Silhouette Scores served as a proxy for clustering quality, influencing the generation of more representative synthetic samples. However, results also suggested that cluster count must be tailored per dataset to avoid performance degradation.

D. Majority Weighted Minority Oversampling Technique (MWMOTE)

MWMOTE is an advanced oversampling algorithm derived from SMOTE (Synthetic Minority Oversampling Technique). It is specifically designed to handle imbalanced datasets by enhancing the selection and weighting of minority class samples before generating synthetic data. Unlike traditional oversampling methods, MWMOTE ensures that synthetic samples are strategically distributed to avoid the formation of noisy or outlier data, which can negatively impact classification accuracy. By improving the selection and weighting mechanisms, MWMOTE creates better-quality synthetic samples, thereby enhancing the overall performance of classification models when handling imbalanced datasets [36].

MWMOTE follows a structured three-phase approach [13].

1. Phase 1: Separation of Majority and Minority Classes
   This phase involves identifying minority class instances that are in close proximity to majority class instances and removing those that are likely to be noisy or misclassified. The steps in this phase include:
   SminorM: Identifying minority samples that are located within the majority class region. These samples are removed to reduce the impact of noise.
   Sbmajor: Determining the borderline samples of the majority class, which are crucial for later calculations.
2. Phase 2: Weighting and Selection of Minority Class Samples
   This phase involves computing the weights for minority class samples to determine their relevance in synthetic data generation. The steps include:
   Iw: Assigning a weight to each minority class sample based on its proximity to the decision boundary [35].
3. Phase 3: Synthetic Data Generation using K-Means Clustering
   The final phase generates synthetic minority class samples using K-Means clustering to ensure better distribution. The process follows these steps:
   Synthetic data is created along the borderline of the minority class, ensuring the new samples are strategically placed for better classification [36].

This structured approach ensures that the MWMOTE algorithm outperforms traditional oversampling techniques by minimizing overfitting, reducing noise, and improving the distribution of synthetic samples across minority class instances.

The proposed MWMOTE + K-Means framework has the potential to be integrated with advanced classifiers like Random Forest, Support Vector Machines (SVM), and Neural Networks to further enhance model robustness. Ensemble methods can particularly benefit from balanced datasets, while deep learning architectures may leverage clustered synthetic data for improved feature learning in minority class detection [37]. Future studies could investigate hybrid pipelines incorporating these learners with the current oversampling strategy.

E. Method

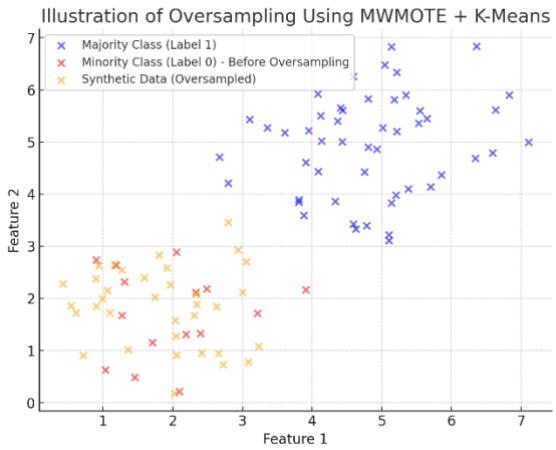

Fig 3. Oversampling

Based on the problem background and research objectives outlined in Chapter 1, this study focuses on addressing imbalanced data using the MWMOTE (Majority Weighted Minority Oversampling Technique) method, enhanced with K-Means Clustering (Fig 3). The research follows a systematic methodology to develop and evaluate an effective solution for class imbalance. The workflow consists of several key stages: problem identification, literature review, model development, model evaluation, and result analysis (Fig 4). Each stage plays a crucial role in ensuring that the proposed

Corresponding author: Meida Cahyo Untoro, cahyo.untoro@if.itera.ac.id, Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.69

approach enhances classification performance while minimizing the risk of overfitting

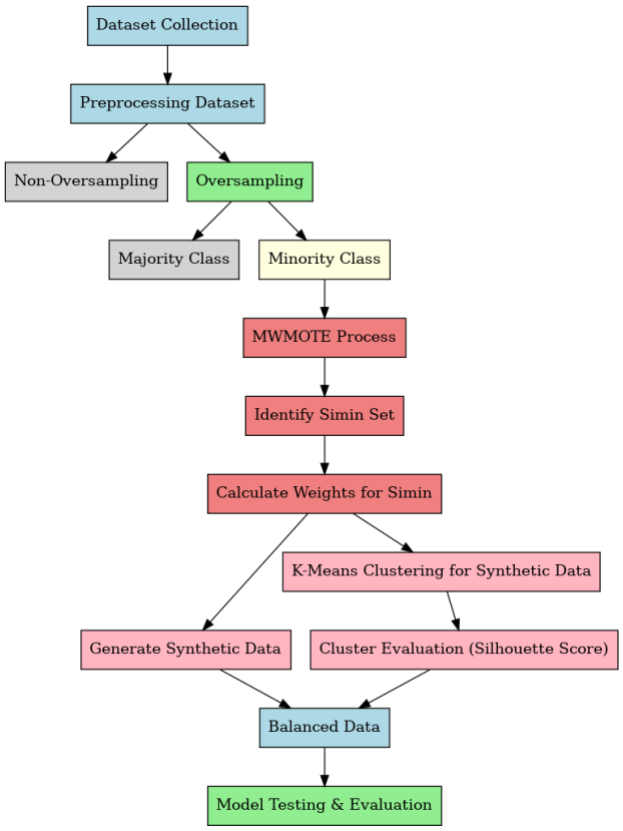

Fig 4. Research Flowchart

The first stage, problem identification, involves analyzing the impact of imbalanced datasets in machine learning applications . Many classification models struggle with minority class instances due to skewed data distribution, leading to biased predictions. To address this, the study explores oversampling techniques with clustering-based enhancements. A literature review is conducted to examine existing approaches, including traditional oversampling methods such as SMOTE, ADASYN, and MWMOTE, as well as advanced clustering-based techniques like CB-MWMOTE and CL-MWMOTE. These insights help formulate an improved model that incorporates K-Means clustering to optimize the distribution of synthetic samples. In the model development phase, the dataset undergoes preprocessing, including data cleaning, normalization, and feature selection, followed by separating majority and minority classes.

The MWMOTE process is applied to identify significant minority samples (Simin set), assign weights, and generate synthetic data. Instead of using standard MWMOTE alone, K-Means clustering is integrated to distribute the synthetic samples more effectively across the dataset. The Silhouette Score method is used to evaluate cluster quality and determine the optimal number of clusters for synthetic sample generation.

Following model development, the evaluation phase assesses the performance of the MWMOTE + K-Means approach. A comparative analysis is conducted between imbalanced data, traditional oversampling methods, and the proposed method using classification metrics such as accuracy, precision, recall, and F1-score. This stage ensures that the model successfully mitigates data imbalance while maintaining generalization capability. The final step, result analysis, interprets the effectiveness of the approach by analyzing its strengths, limitations, and computational trade-offs. The expected outcome is that MWMOTE combined with K-Means clustering will improve class balance while avoiding synthetic sample redundancy and overfitting. Additionally, the study considers the computational cost associated with clustering and explores potential optimizations. By systematically following this workflow, the research aims to demonstrate the effectiveness of clustering-enhanced oversampling as a solution for handling imbalanced datasets in machine learning applications.

The findings of this study pave the way for future exploration in multi-class imbalance scenarios, where the distribution among multiple classes is uneven. Additionally, adapting the MWMOTE + K-Means approach for real-time data streams could address imbalance in applications such as fraud detection or network intrusion systems. Another promising direction involves incorporating adaptive clustering methods to dynamically adjust the number of clusters during the oversampling process based on dataset complexity.

F. Statistical Analysis

Fig 5. Example of confusion matrix

After the synthetic data generation process and model training, several statistical techniques were applied to evaluate the performance of the MWMOTE + K-Means method. The following methods were used for evaluation:

Classification Metrics (Fig 5): The performance of the models was evaluated using standard classification metrics such as:

1. Accuracy: The proportion of correctly classified instances out of the total instances.
2. Precision: The proportion of true positives among all predicted positives.
3. Recall: The proportion of true positives among all actual positives.

Corresponding author: Meida Cahyo Untoro, cahyo.untoro@if.itera.ac.id, Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.69

4. F-measure: The harmonic mean of precision and recall, providing a balanced evaluation of both metrics.

These metrics were calculated based on the confusion matrix for each model, and the results were compared between the baseline model, MWMOTE, and the MWMOTE + K-Means method.

Computational Complexity: The computational cost of the MWMOTE + K-Means method was also evaluated by measuring the execution time for each dataset. The clustering process (K-Means) and synthetic sample generation were analyzed to determine the time required for each step.

3. RESULTS

A. Oversampling MWMOTE

The MWMOTE (Majority Weighted Minority Oversampling Technique) method consists of three structured phases designed to address imbalanced datasets. This study applies MWMOTE to 10 imbalanced datasets, including the Wine Quality dataset, to enhance classification performance.

The research methodology in this study is composed of sequential stages: dataset selection, preprocessing, MWMOTE weighting, K-Means clustering, synthetic data generation, and model evaluation. Dataset selection prioritized real-world relevance and imbalance characteristics, drawing from UCI and Kaggle repositories. Preprocessing involved normalization, outlier removal, and class stratification. MWMOTE was then applied to identify and weight informative minority instances, followed by K-Means clustering to group similar data points. The number of clusters was determined using the Silhouette Score. Synthetic samples were generated within clusters to preserve locality and structure.

In Phase 1 (Identification of Simin Set), the process starts with SminF, which identifies minority class instances that are present within the majority class using Nearest Neighbors (NN). These instances are removed to ensure a clear separation between the two classes. After this step, a verification process is conducted to confirm the complete removal of minority instances from the majority class region. Following this, Sbmaj is used to identify borderline majority class instances, which are data points located near the decision boundary. Lastly, Simin is determined, representing informative minority class instances, which are crucial for synthetic data generation. If synthetic data points are generated near the decision boundary, they are grouped under Simin to ensure that the new samples contribute effectively to improving class distribution.

In Phase 2 (Weight Calculation for Simin), each minority class instance is assigned a weight based on its proximity to the majority class boundary. The selection of weights is optimized to minimize redundant synthetic data generation while maximizing informative sample selection. In Phase 3 (Synthetic Data Generation using K-Means Clustering), synthetic samples are generated based on cluster formation. Fig 6., Elbow Method is used to predict the optimal number of clusters, and the Silhouette Score is used for validating the best cluster configuration. In this study, two clusters were identified as optimal, with a Silhouette Score of 0.528. Synthetic data points are then distributed within these clusters to ensure better representation of the minority class.

The research methodology in this study is composed of sequential stages: dataset selection, preprocessing, MWMOTE weighting, K-Means clustering, synthetic data generation, and model evaluation. Dataset selection prioritized real-world relevance and imbalance characteristics, drawing from UCI and Kaggle repositories. Preprocessing involved normalization, outlier removal, and class stratification. MWMOTE was then applied to identify and weight informative minority instances, followed by K-Means clustering to group similar data points. The number of clusters was determined using the Silhouette Score. Synthetic samples were generated within clusters to preserve locality and structure

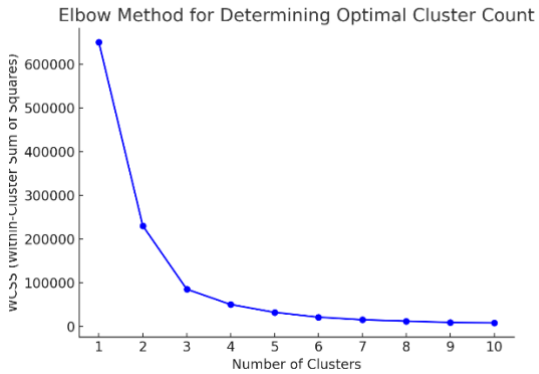

Fig 6. Elbow Method for Determining the Optimal Number of Clusters

B. Evaluation of K-Means Clustering in Oversampling Performance

The comparison of Silhouette Scores for the different numbers of clusters formed and the resulting performance values (Table 2). For the Abalone dataset, with a Silhouette Score of 0.54 for 2 clusters, the performance metrics were: Accuracy of 75.34%, Precision of 99.04%, Recall of 74.82%, and an F-measure of 85%. When the Silhouette Score dropped to 0.49 for 4 clusters, the Accuracy decreased to 70.44%, with Precision of 57.49%, Recall of 66.21%, and an F-measure of 61.54%. A further decrease in the Silhouette Score to 0.37 for 9 clusters yielded similar results to the 4-cluster case, with Accuracy remaining at 70.44%, and Precision, Recall, and F-measure values unchanged. The highest Accuracy of 75.34% was achieved with a Silhouette Score of 0.54 for 2 clusters, followed by a decrease in Accuracy for 4 and 9 clusters, despite different Silhouette Scores.

Corresponding author: Meida Cahyo Untoro, cahyo.untoro@if.itera.ac.id, Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.69

Table 2. Comparison of Performance Results Based on the Number of Clusters

MWMOTE	IR	Silhouette Score	Cluster	Accuracy	Precision	Recall	F-MEASURE
Abalone	0,94 : 0,05	0.54120	2 cluster	**75,34**	99,04	74,82	85,24
		0.49175	4 cluster	70,44	57,49	66,21	61,54
		0.37287	9 cluster	70.44	57,48	66,21	61,54
Breast	0,66 : 0,34	0.70628	2 cluster	**74**	100	66	80
		0.62738	4 cluster	71,2	90	67	78
		0.60485	9 cluster	71,2	90	67	78
Ecolli	0,77 : 0,22	0.43852	2 cluster	81,73	95,45	71,18	81,55
		0.22748	4 cluster	**83,14**	97,91	78,33	87,03
		0.21436	9 cluster	**83,14**	97,92	78,33	87,03
Glass	0,76 : 0,23	0.47922	2 cluster	90,69	92.85	92,85	92,85
		0.41129	4 cluster	90,15	78,77	92,5	85,05
		0.46371	9 cluster	**93,5**	77,5	96,87	86,12
Pg	0,78 : 0,21	0.90445	2 cluster	**94,24**	97,49	96,47	96,98
		0.87483	4 cluster	91,31	92,8	81,05	86,52
		0.79068	9 cluster	91,31	92,8	81,05	86,52
Robot	0,68 : 0,31	0.24299	2 cluster	**91,31**	92,8	81,05	86,52
		0.27505	4 cluster	88,36	86,67	85,01	85,83
		0.27031	9 cluster	88,94	71,12	73,11	72,04
Satimage	0,77 : 0,22	0.53080	2 cluster	**78,93**	71,12	72,99	72,04
		0.40906	4 cluster	73,81	69,17	84,7	76,15
		0.24977	9 cluster	71,72	60,56	84,34	70,5
Segment	0,76 : 0,23	0.51720	2 cluster	71,72	60,56	84,34	70,5
		0.34573	4 cluster	**100**	100	100	100
		0.35122	9 cluster	**100**	100	100	100
Wine	0,95 : 0,04	0.52830	2 cluster	**100**	100	100	100
		0.49631	4 cluster	91,66	100	89,28	94,33
		0.46899	9 cluster	98,52	100	96,77	98,36
Yeast	0,78 : 0,21	0.37128	2 cluster	**98,52**	100	96,77	98,36
		0.23671	4 cluster	96,77	82,5	91,42	83,89
		0.20388	9 cluster	96,34	82,97	86,74	84,81

For the Breast dataset, the Silhouette Score of 0.70 for 2 clusters resulted in an Accuracy of 74%, Precision of 100%, Recall of 66%, and F-measure of 80%. When the Silhouette Score decreased to 0.62 for 4 clusters, the Accuracy dropped to 71.2%, while Precision remained at 90%, Recall at 67%, and F-measure at 78%. With a

Corresponding author: Meida Cahyo Untoro, cahyo.untoro@if.itera.ac.id, Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.69

Silhouette Score of 0.37 for 9 clusters, the results remained the same as for 4 clusters, with Accuracy at 71.2% and Precision, Recall, and F-measure unchanged. The highest Accuracy of 74% was achieved with the Silhouette Score of 0.70 for 2 clusters, showing a decline in Accuracy for both 4 and 9 clusters, despite different Silhouette Scores.

In the case of the Ecoli dataset, a Silhouette Score of 0.43 for 2 clusters led to an Accuracy of 81.73%, Precision of 95.45%, Recall of 71.18%, and F-measure of 87.03%. When the Silhouette Score dropped to 0.22 for 4 clusters, the Accuracy improved to 90%, with Precision at 97.97%, Recall at 78.33%, and F-measure at 87.03%. The Silhouette Score of 0.37 for 9 clusters resulted in the same Accuracy of 90%, with Precision, Recall, and F-measure unchanged. The highest Accuracy of 90% was achieved with Silhouette Scores of 0.32 and 0.21 for 4 and 9 clusters, showing a decrease in Accuracy for the 2-cluster case with a Silhouette Score of 0.70.

For the Glass dataset, the Silhouette Score of 0.47 for 2 clusters resulted in an Accuracy of 90.69%, Precision of 92.45%, Recall of 71.18%, and F-measure of 81.55%. A Silhouette Score of 0.49 for 4 clusters led to a Accuracy of 70.44%, Precision of 57.49%, Recall of 66.21%, and F-measure of 61.54%. The Silhouette Score of 0.37 for 9 clusters produced the same performance values as the 4-cluster case. The highest Accuracy of 90.69% was achieved with the Silhouette Score of 0.47 for 2 clusters, while Accuracy decreased for 4 and 9 clusters.

The Page-Block dataset showed a Silhouette Score of 0.54 for 2 clusters, leading to an Accuracy of 75.34%, Precision of 99.04%, Recall of 74.82%, and F-measure of 85%. With a Silhouette Score of 0.49 for 4 clusters, Accuracy decreased to 70.44%, with Precision of 57.49%, Recall of 66.21%, and F-measure of 61.54%. Similarly, the Silhouette Score of 0.37 for 9 clusters resulted in the same performance metrics as 4 clusters, with Accuracy at 70.44%. As with other datasets, the highest Accuracy of 75.34% was achieved with Silhouette Score of 0.54 for 2 clusters, and a decline in Accuracy for 4 and 9 clusters was observed.

For the Robot dataset, with a Silhouette Score of 0.54 for 2 clusters, the performance metrics were: Accuracy of 75.34%, Precision of 99.04%, Recall of 74.82%, and F-measure of 85%. When the Silhouette Score dropped to 0.49 for 4 clusters, the Accuracy decreased to 70.44%, with Precision of 57.49%, Recall of 66.21%, and F-measure of 61.54%. A Silhouette Score of 0.37 for 9 clusters produced the same performance as 4 clusters, with Accuracy remaining at 70.44%. As with previous datasets, the highest Accuracy of 75.34% was obtained for 2 clusters with a Silhouette Score of 0.54, and there was a decrease in Accuracy for 4 and 9 clusters.

The Satimage dataset yielded similar results, with a Silhouette Score of 0.53 for 2 clusters achieving Accuracy of 75.34%, Precision of 99.04%, Recall of 74.82%, and F-measure of 85%. The Silhouette Score of 0.49 for 4 clusters resulted in Accuracy of 70.44%, Precision of 57.49%, Recall of 66.21%, and F-measure of 61.54%. With a Silhouette Score of 0.37 for 9 clusters, the performance metrics remained unchanged, and the Accuracy was again 70.44%.

Finally, for the Wine dataset, the Silhouette Score for 2, 4, and 9 clusters was all 0.54, with Accuracy consistently at 91.66%, Precision at 100%, Recall at 89.28%, and F-measure at 94.33%. This dataset demonstrated high Accuracy across all clusters, despite varying Silhouette Scores.

In summary, for most datasets, the highest Accuracy was achieved with 2 clusters and the highest Silhouette Scores, whereas performance generally declined as the number of clusters increased, especially when the Silhouette Score decreased. This trend highlights the sensitivity of the MWMOTE + K-Means method to the clustering configuration.

4. DISCUSSION

A. Model Evaluation

After balancing the dataset using oversampling techniques such as MWMOTE and MWMOTE combined with K-Means clustering, followed by classification using the Naïve Bayes algorithm, the next step involves evaluating the model's performance. The evaluation aims to assess the effectiveness of the oversampling approach and determine how well the classification model performs on different datasets.

To achieve this, the study utilizes a confusion matrix, comparing the original dataset with the predicted classifications from the trained model. This evaluation method provides insights into classification accuracy, precision, recall, and F-measure, allowing for a detailed analysis of model effectiveness. The following section presents a comparison of performance metrics across different datasets, demonstrating the impact of MWMOTE-based oversampling techniques on classification outcomes. This study opens opportunities for integrating MWMOTE + K-Means with ensemble classifiers such as Random Forest, or deep learning models like CNNs for image-based data. Such integrations could potentially leverage the improved data balance to further boost classification performance.

The comparison between accuracy obtained from balanced (oversampled) and imbalanced (non-oversampled) data reveals varying effects across different datasets (Table 3). Some datasets, such as Segment, showed no change in accuracy, regardless of whether oversampling was applied or not. However, several datasets, including Breast, Ecoli, Glass, Page-Block, Satimage, Wine, and Yeast, exhibited an increase in accuracy when using MWMOTE + K-Means oversampling, compared to the original imbalanced dataset. In contrast, Robot and Abalone datasets experienced a decline in accuracy after applying MWMOTE + K-Means oversampling, suggesting that in some cases, synthetic data generation may negatively impact classification performance. The overall accuracy improvement rate for MWMOTE + K-Means across 10

Corresponding author: Meida Cahyo Untoro, cahyo.untoro@if.itera.ac.id, Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.69

datasets is 70%, which is calculated based on 7 datasets showing an increase in accuracy out of the total 10 datasets used in the study. Conversely, the accuracy decline rate due to MWMOTE + K-Means oversampling is 20%, as 2 datasets exhibited reduced accuracy after oversampling. These results indicate that while MWMOTE + K-Means is generally effective in enhancing classification accuracy, its impact may vary depending on the dataset structure, highlighting the importance of evaluating dataset-specific characteristics before applying clustering-based oversampling techniques.

Table 3. Accuracy Comparison Across Datasets

Dataset	Normal	MWMOTE	MWMOTE + K-Means	Change (%)
Abalone	**92,46**	75,34	75,34	-17,12
Breast	61,9	61,9	**74**	12,1
Ecolli	65,67	74,62	**81,73**	16,06
Glass	90,69	**93,02**	**93,02**	2,33
Page-Blocks	94,15	**94,24**	**94,24**	0,09
Robot	**100**	78,36	78,36	-21,64
Satimage	77,15	**78,16**	**78,16**	1,01
Segment	**100**	**100**	**100**	0
Wine	91,66	**91,67**	**91,67**	0,01

B. Comparison of Precision, Recall, and F-Measure Across Datasets

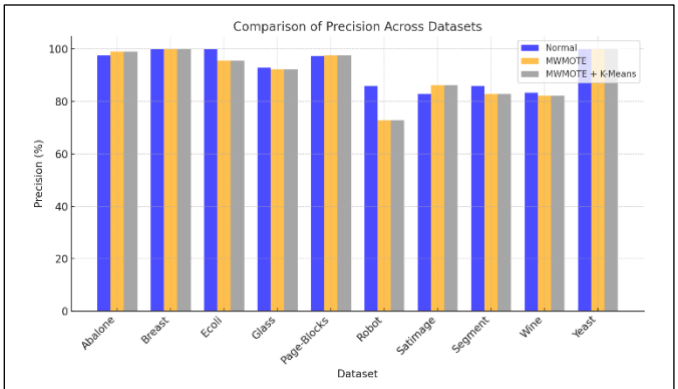

Fig 7. Precision Comparison Across

Fig 7, the precision values for different datasets were compared across three conditions : the original dataset (Normal), the dataset processed with MWMOTE, and the dataset enhanced using MWMOTE + K-Means. The results indicate that MWMOTE and MWMOTE + K-Means generally improve precision in most datasets, with some variations based on data characteristics. The Segment, Wine, and Yeast datasets maintain a precision of 100% across all conditions, while Robot and Satimage show a decline in precision after oversampling.

The recall values demonstrate how well the model identifies positive instances across different datasets (Fig 8). The comparison shows that MWMOTE + K-Means leads to an improvement in recall for most datasets, particularly in Abalone, Ecoli, and Glass. However, Breast and Robot datasets experience minimal changes, with recall values remaining close to the original dataset. The Segment dataset maintains a recall of 100%, suggesting that its classification performance is unaffected by oversampling techniques.

Fig 8. Recall Comparison Across Datasets

Fig 9, F-Measure represents the harmonic mean of precision and recall, providing a balanced evaluation of classification performance. The results indicate that MWMOTE + K-Means effectively enhances F-Measure values in several datasets, particularly in Ecoli, Glass, and Yeast. However, some datasets, such as Robot and Satimage, show minor decreases in F-Measure after applying oversampling, indicating that the effectiveness of oversampling depends on dataset characteristics. Overall, the findings suggest that while MWMOTE + K-Means improves classification performance in most cases, some datasets may require further optimization to avoid performance drops.

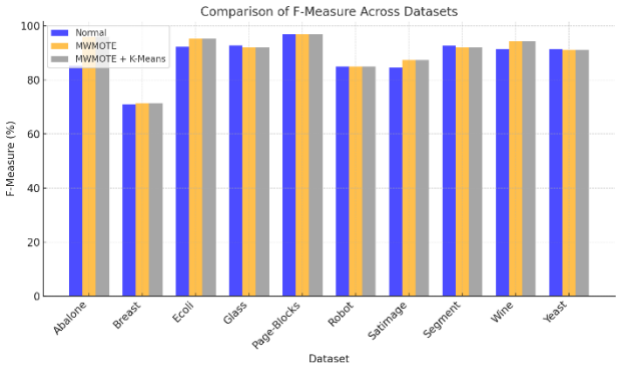

Fig 9. F-Measure Comparison Across Datasets

C. Evaluation of Computational Time in Running the Program

The evaluation of computational time is conducted to analyze the performance of running the program across different datasets. The execution time is measured in seconds, and each test is repeated three times to ensure

Corresponding author: Meida Cahyo Untoro, cahyo.untoro@if.itera.ac.id, Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.69

reliability and eliminate potential inconsistencies caused by hardware performance fluctuations. The computation begins from the evaluation of the model using the Naïve Bayes classifier and extends to calculating the performance metrics for each dataset. The variations in computational time across different datasets are influenced by several factors, including the number of instances, the number of attributes, and the data ratio between majority and minority classes. By averaging the computation time over three runs for each dataset, the study ensures a fair and accurate assessment of the impact of oversampling techniques on computational efficiency.

The results indicate differences in execution time when comparing MWMOTE + K-Means with the Normal dataset (Fig 10). The Abalone dataset showed a time difference of 0.028 seconds, while the Breast dataset recorded a difference of 0.021 seconds. Similarly, for the Ecoli dataset, the computation time difference was 0.016 seconds, whereas the Glass dataset exhibited a more significant difference of 0.052 seconds. The Page-Block dataset showed a difference of 0.027 seconds, and for Satimage, the time difference was 0.023 seconds. Additionally, the Segment dataset had the highest computation time difference at 0.058 seconds, followed by Wine with 0.018 seconds, and Yeast with 0.029 seconds. These variations suggest that while MWMOTE + K-Means adds a minor computational overhead, the impact is manageable and dependent on dataset complexity. Overall, the results emphasize that although oversampling increases computation time slightly, it remains within an acceptable range, ensuring that the improved model performance outweighs the computational cost.

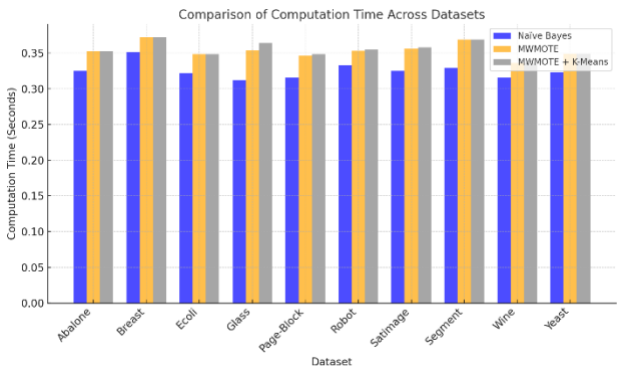

Fig 10. Computational Time Comparison Across Different Datasets

D. Research Analysis

The study on Handling Imbalanced Data Using the K-Means Clustering Approach in MWMOTE was conducted using 10 UCI datasets, producing varied classification performance results. The MWMOTE + K-Means approach showed an accuracy increase of 20% compared to MWMOTE alone and a 70% increase over the original (non-oversampled) dataset across different datasets. Several datasets, including Breast, Ecoli, Glass, Page-Block, Satimage, Segment, Wine, and Yeast, demonstrated higher accuracy using MWMOTE + K-Means compared to the original dataset. However, datasets such as Abalone and Robot experienced a decline in accuracy after oversampling, indicating that clustering-based oversampling may not always enhance model performance. Additionally, precision improved in 30% of the datasets when applying MWMOTE + K-Means, with Breast, Glass, Page-Block, Wine, and Yeast showing the most significant gains. Recall and F-Measure also increased in 40% of the datasets, further supporting the effectiveness of MWMOTE + K-Means in specific cases. However, for datasets such as Abalone, Ecoli, Page-Block, Satimage, and Yeast, classification performance decreased, highlighting that the impact of oversampling varies based on dataset characteristics.

The number of clusters used to generate synthetic data for the minority class significantly influenced classification performance. However, determining the optimal cluster count using the Silhouette Score did not always guarantee improved model performance. For example, in the Ecoli dataset, the model achieved consistent accuracy (81.73%) across 2, 4, and 9 clusters, with identical precision (90.00%), recall (71.18%), and F-Measure (81.55%), indicating that the number of clusters did not impact performance in this case. Additionally, computation time increased across all datasets when using MWMOTE + K-Means, with delays ranging from 0.016 to 0.058 seconds, depending on dataset size, attribute count, and class imbalance ratio. The increase in computation time was attributed to the clustering process required for weighting and generating synthetic data, making MWMOTE more computationally efficient than MWMOTE + K-Means. These findings suggest that while MWMOTE + K-Means can improve classification performance in certain cases, it also introduces additional computational complexity, requiring careful consideration of dataset characteristics before applying clustering-based oversampling techniques.

Despite improvements in model performance, MWMOTE + K-Means introduces additional computational overhead due to clustering and sample generation stages. Moreover, its performance may diminish on very high-dimensional datasets or where minority class examples are extremely sparse. Future optimization is needed to scale the method for real-time systems and large-scale datasets.

5. CONCLUSION

The MWMOTE + K-Means Oversampling method has been shown to effectively address data imbalance issues across 10 UCI repository datasets, including Abalone, Breast, Ecoli, Glass, Page-Block, Robot, Satimage, Segment, Wine, and Yeast. Based on the analysis in Chapter 4, 70% of the datasets showed improved accuracy using MWMOTE + K-Means compared to the original dataset, with Breast, Ecoli, Glass, Page-Block, Satimage, Wine, and Yeast demonstrating significant gains. Additionally, 20% of the datasets exhibited higher

Corresponding author: Meida Cahyo Untoro, cahyo.untoro@if.itera.ac.id, Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.69

accuracy when compared to MWMOTE alone. Improvements in precision (10%) were observed in Abalone, Page-Block, and Wine, while 30% of datasets showed better precision over MWMOTE alone. Recall values increased in 40% of datasets, while F-Measure improved in 40% of cases, particularly in Breast, Ecoli, Segment, and Wine. Despite these performance gains, MWMOTE + K-Means increased computational time in 8 out of 10 datasets, primarily due to centroid and cluster formation in synthetic data generation. Hardware differences also contributed to processing speed variations. While MWMOTE + K-Means enhances classification performance, its higher computational cost requires careful consideration when choosing an oversampling method for imbalanced data handling. Future work could explore dynamic clustering or adaptive sampling mechanisms to support real-time deployment and scalability for multi-class classification scenarios.

REFERENCES

[1] S. Matharaarachchi, M. Domaratzki, and S. Muthukumarana, "Machine Learning with Applications Enhancing SMOTE for imbalanced data with abnormal minority instances," vol. 18, no. September, 2024.

[2] S. Afrose, W. Song, C. B. Nemeroff, C. Lu, and D. (Daphne) Yao, "Subpopulation-specific machine learning prognosis for underrepresented patients with double prioritized bias correction," *Commun. Med.*, vol. 2, no. 1, pp. 1–14, 2022, doi: 10.1038/s43856-022-00165-w.

[3] A. M. Halim, M. Dwifebri, and F. Nhita, "Handling Imbalanced Data Sets Using SMOTE and ADASYN to Improve Classification Performance of Ecoli Data Sets," *Build. Informatics, Technol. Sci.*, vol. 5, no. 1, pp. 246–253, 2023, doi: 10.47065/bits.v5i1.3647.

[4] D. Dablain, B. Krawczyk, and N. Chawla, "Towards a holistic view of bias in machine learning: bridging algorithmic fairness and imbalanced learning," *Discov. Data*, vol. 2, no. 1, 2024, doi: 10.1007/s44248-024-00007-1.

[5] D. Sets, "Bias and Class Imbalance in Oncologic Data — Towards Inclusive and Transferrable AI in Large Scale Oncology Data Sets," 2022.

[6] M. C. Untoro and J. L. Buliali, "Penanganan imbalance class data laboratorium kesehatan dengan Majority Weighted Minority Oversampling Technique," *Regist. J. Ilm. Teknol. Sist. Inf.*, vol. 4, no. 1, p. 23, 2018, doi: 10.26594/register.v4i1.1184.

[7] S. Mishra, "Handling Imbalanced Data : SMOTE vs . Random Undersampling," *Int. Res. J. Eng. Technol.*, vol. 4, no. 8, pp. 317–320, 2017, [Online]. Available: https://irjet.net/archives/V4/i8/IRJET-V4I857.pdf

[8] O. Alam, N. Khan, and A. Ullah, "Unlocking Rare Diseases Genetics : Insights from Genome-Wide Association Studies and Single Nucleotide Polymorphisms," pp. 1–28, 2024.

[9] A. I. N. U. S. Healthcare, O. Emi-johnson, K. Nkrumah, A. Folasole, and T. K. Amusa, "International Journal of Engineering Technology Research & Management OPTIMIZING MACHINE LEARNING FOR IMBALANCED CLASSIFICATION : International Journal of Engineering Technology Research & Management," no. 11, pp. 89–106, 2023.

[10] R. Baker, C. Mills, and J. Choi, "The Difficulty of Achieving High Precision with Low Base Rates for High-Stakes Intervention," *15th Int. Conf. Learn. Anal. Knowledge, LAK 2025*, pp. 790–796, 2025, doi: 10.1145/3706468.3706477.

[11] D. A. Wood, S. Mardanirad, and H. Zakeri, "Effective prediction of lost circulation from multiple drilling variables: a class imbalance problem for machine and deep learning algorithms," *J. Pet. Explor. Prod. Technol.*, vol. 12, no. 1, pp. 83–98, 2022, doi: 10.1007/s13202-021-01411-y.

[12] P. Kumar, R. Bhatnagar, K. Gaur, and A. Bhatnagar, "Classification of Imbalanced Data:Review of Methods and Applications," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1099, no. 1, p. 012077, 2021, doi: 10.1088/1757-899x/1099/1/012077.

[13] M. C. Untoro, "MWMOTE optimization for imbalanced data using complete linkage," *J. Teknol. dan Sist. Komput.*, vol. 9, no. 2, pp. 77–82, 2021, doi: 10.14710/jtsiskom.2021.13748.

[14] A. Azhari, E. Buulolo, and N. Sialalhi, "Sistem Rekomendasi Dosen Pendamping Skripsi Berbasis Text Rank menggunakan Metode Cosine Similarity," *Pelita Inform. …*, vol. 10, pp. 119–122, 2022, [Online]. Available: https://www.ejurnal.stmik-budidarma.ac.id/index.php/pelita/article/view/3772%0Ahttps://www.ejurnal.stmik-budidarma.ac.id/index.php/pelita/article/download/3772/2499

[15] L. Wang, M. Han, X. Li, N. Zhang, and H. Cheng, "Review of Classification Methods on Unbalanced Data Sets," *IEEE Access*, vol. 9, pp. 64606–64628, 2021, doi: 10.1109/ACCESS.2021.3074243.

[16] T. Wongvorachan, S. He, and O. Bulut, "A Comparison of Undersampling, Oversampling, and SMOTE Methods for Dealing with Imbalanced Classification in Educational Data Mining," *Inf.*, vol. 14, no. 1, 2023, doi: 10.3390/info14010054.

[17] Y. Yang, H. A. Khorshidi, and U. Aickelin, "A review on over-sampling techniques in classification of multi-class imbalanced datasets: insights for medical problems," *Front. Digit. Heal.*, vol. 6, no. July, 2024, doi: 10.3389/fdgth.2024.1430245.

[18] F. Poucin, A. Kraus, and M. Simon, "Boosting Instance Segmentation with Synthetic Data: A study to overcome the limits of real world data sets," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2021-Octob, pp. 945–953, 2021, doi: 10.1109/ICCVW54120.2021.00110.

[19] "MWMOTE-FRIS-INFFC : An Improved Majority Weighted Minority Oversampling Technique for

Corresponding author: Meida Cahyo Untoro, cahyo.untoro@if.itera.ac.id, Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.69

Solving Noisy and Imbalanced Classification Datasets," pp. 1–15, 2024.

[20] A. Islam, S. B. Belhaouari, A. U. Rehman, and H. Bensmail, "KNNOR: An oversampling technique for imbalanced datasets[Formula presented]," *Appl. Soft Comput.*, vol. 115, p. 108288, 2022, doi: 10.1016/j.asoc.2021.108288.

[21] C. Esposito, G. A. Landrum, N. Schneider, N. Stiefl, and S. Riniker, "GHOST: Adjusting the Decision Threshold to Handle Imbalanced Data in Machine Learning," *J. Chem. Inf. Model.*, vol. 61, no. 6, pp. 2623–2640, 2021, doi: 10.1021/acs.jcim.1c00160.

[22] A. X. Wang, S. S. Chukova, and B. P. Nguyen, "Synthetic minority oversampling using edited displacement-based k-nearest neighbors," *Appl. Soft Comput.*, vol. 148, no. May, p. 110895, 2023, doi: 10.1016/j.asoc.2023.110895.

[23] M. Ahmed, R. Seraj, and S. M. S. Islam, "The k-means algorithm: A comprehensive survey and performance evaluation," *Electron.*, vol. 9, no. 8, pp. 1–12, 2020, doi: 10.3390/electronics9081295.

[24] P. Soltanzadeh and M. Hashemzadeh, "RCSMOTE: Range-Controlled synthetic minority over-sampling technique for handling the class imbalance problem," *Inf. Sci. (Ny).*, vol. 542, pp. 92–111, 2021, doi: 10.1016/j.ins.2020.07.014.

[25] D. A. I. C. Dewi and D. A. K. Pramita, "Analisis Perbandingan Metode Elbow dan Silhouette pada Algoritma Clustering K-Medoids dalam Pengelompokan Produksi Kerajinan Bali," *Matrix J. Manaj. Teknol. dan Inform.*, vol. 9, no. 3, pp. 102–109, 2019, doi: 10.31940/matrix.v9i3.1662.

[26] G. P. Selvarajan, "Harnessing AI-Driven Data Mining for Predictive Insights : A Framework for Enhancing Decision- Making in Dynamic Data Environments," no. February 2021, 2024.

[27] M. C. Untoro and M. A. N. M. Yusuf, "Evaluate of Random Undersampling Method and Majority Weighted Minority Oversampling Technique in Resolve Imabalanced Dataset," *IT J. Res. Dev.*, vol. 8, no. 1, pp. 1–13, 2023, doi: 10.25299/itjrd.2023.12412.

[28] J. Fonseca, G. Douzas, and F. Bacao, "Improving imbalanced land cover classification with k-means smote: Detecting and oversampling distinctive minority spectral signatures," *Inf.*, vol. 12, no. 7, 2021, doi: 10.3390/info12070266.

[29] M. C. Mihaescu and P. S. Popescu, "Review on publicly available datasets for educational data mining," *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 11, no. 3, pp. 1–16, 2021, doi: 10.1002/widm.1403.

[30] G. Mustafa, M. Usman, M. T. Afzal, A. Shahid, and A. Koubaa, "A comprehensive evaluation of metadata-based features to classify research paper's topics," *IEEE Access*, vol. 9, pp. 133500–133509, 2021, doi: 10.1109/ACCESS.2021.3115148.

[31] H. Bichri, A. Chergui, and M. Hain, "Investigating the Impact of Train / Test Split Ratio on the Performance of Pre-Trained Models with Custom Datasets," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 2, pp. 331–339, 2024, doi: 10.14569/IJACSA.2024.0150235.

[32] S. Bagui and K. Li, "Resampling imbalanced data for network intrusion detection datasets," *J. Big Data*, vol. 8, no. 1, 2021, doi: 10.1186/s40537-020-00390-x.

[33] T. A. Alghamdi and N. Javaid, "A Survey of Preprocessing Methods Used for Analysis of Big Data Originated from Smart Grids," *IEEE Access*, vol. 10, pp. 29149–29171, 2022, doi: 10.1109/ACCESS.2022.3157941.

[34] J. T. Hancock, T. M. Khoshgoftaar, and J. M. Johnson, "Evaluating classifier performance with highly imbalanced Big Data," *J. Big Data*, vol. 10, no. 1, 2023, doi: 10.1186/s40537-023-00724-5.

[35] C. Liao and M. Dong, "Acwgan: an Auxiliary Classifier Wasserstein Gan-Based Oversampling Approach for Multi-Class Imbalanced Learning," *Int. J. Innov. Comput. Inf. Control*, vol. 18, no. 3, pp. 703–721, 2022, doi: 10.24507/ijicic.18.03.703.

[36] M. Mohamad, A. Selamat, I. M. Subroto, and O. Krejcar, "Improving the classification performance on imbalanced data sets via new hybrid parameterisation model," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 33, no. 7, pp. 787–797, 2021, doi: 10.1016/j.jksuci.2019.04.009.

[37] A. S. Dina, A. B. Siddique, and D. Manivannan, "Effect of Balancing Data Using Synthetic Data on the Performance of Machine Learning Classifiers for Intrusion Detection in Computer Networks," *IEEE Access*, vol. 10, no. August, pp. 96731–96747, 2022, doi: 10.1109/ACCESS.2022.3205337.

AUTHOR BIOGRAPHY

Meida Cahyo Untoro, received the B.S. degree in Teknik Komputer from STMIK STIKOM Bali in 2012, M.Kom. degrees in Teknik Informatika from the Institut Teknologi Sepuluh Nopember Surabaya, Indonesia in 2018, 20219 he has been an Assistant Professor with the Teknik Informatika, Institut Teknologi Sumatera, Indonesia. Since 2025, he is an IEEE member.

Corresponding author: Meida Cahyo Untoro, cahyo.untoro@if.itera.ac.id, Department of Informatics Engineering, Fakultas Teknologi Industri, Institut Teknologi Sumatera, Lampung, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.69