Journal of Applied Engineering and Technological Science
Vol 6(2) 2025: 1101-1120

SOLVING SIMULATED IMBALANCED BODY PERFORMANCE DATA USING A-SUWO AND TOMEK LINK ALGORITHM

Febryan Grady[1*], Joel Rizky Wahidiyat[2], Abba Suganda Girsang[3]
Computer Science Department, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University Jakarta, 11480, Indonesia[123]
febryan.grady@binus.ac.id

Received: 26 March 2024, Revised: 28 November 2024, Accepted: 14 January 2025
**Corresponding Author*

ABSTRACT
This research examines the impact of various sampling techniques on the performance of classification models in the context of imbalanced datasets, employing the body performance dataset as a case study. Many studies in this field analyze the effect of sampling techniques on a model performance, however they often begin with imbalance datasets, lacking a balanced baseline for comparison. This research addresses that gap by simulating an imbalanced dataset from an originally balanced dataset, obtaining a target reference point for evaluating the effectiveness of the sampling methods. The dataset is categorized into three versions: (1) a normal distribution, (2) a simulated imbalanced distribution, and (3) a synthesized dataset achieved through various data sampling techniques, including oversampling with Adaptive Semi-Unsupervised Weighted Oversampling (A-SUWO), undersampling with Tomek Link, and hybrid sampling combining both techniques. The primary objective of this research is to identify sampling techniques, when combined with model performance, closely match the performance observed in the original balanced dataset. Based on all experiments using Decision Tree, Random Forest, and K-Nearest Neighbors (KNN) as classifiers, both A-SUWO and Tomek Link led to overfitting due to discernible gap between the training and testing accuracy, averaging 0.21304. Despite overftting and general performance issue, the undersampling with Tomek Link obtained highest test accuracy (0.65023), outperforming A-SUWO (0.62883) and the hybrid approach (0.63568) on average. These findings highlight the importance of appropriate sampling techniques and optimizing model performance in imbalanced datasets.
Keywords: *A-SUWO, Body Performance Data, Data Sampling Techniques, Imbalanced, Tomek Link.*

1. Introduction

Data is a crucial asset and key resource in performing research and development. It is favored for the data to be balanced and accurate as data plays a significant role in making strategic decisions (Hosen et al., 2024; Ionescu & Diaconita, 2023). However, in many real-world scenarios, some data may suffer from skewed or imbalanced distributions (Aguiar et al., 2024; Amin et al., 2016; Hasib et al., 2020). An imbalanced distribution occurs when one class (majority class) contains significantly more samples than the other (minority class) (Thabtah et al., 2020a; Wang & Yao, 2012). Typically, the majority class comprises negative samples, while the minority class includes positive samples (Haibo He & Garcia, 2009). An example of imbalance data distribution is in detecting rare but important diseases, where the number of cases with rare disease is usually much smaller compared to the number of healthy cases. Imbalance data problems can occur in datasets with only two classes, known as binary class, as well as in datasets with more than two classes, known as multi-class (Ali et al., 2019).

The presence of imbalance data distribution poses challenges for machine learning algorithms in performing tasks, such as classification, due to performance bias. Traditionally, the primary objective for classification algorithms is to maximize the overall acccuracy. However, when dealing with imbalanced datasets, this approach may not be suitable, as the majority class tends to dominate the overall distribution of the dataset. This leads to classifers achieving higher accuracy for the majority class while exhibiting lower accuracy for the minority class (Kaur et al., 2019; Thabtah et al., 2020b). Research investigating the impact of class imbalance on classification models discovered that the relationship between the class imbalance ratio and the classifier accuracy follows a convex curve. This implies that as the imbalance ratios increase, the decline in classifier accuracy becomes more pronounced (Thabtah

et al., 2020b). Therefore, it is crucial to address these imbalance issues before performing model training to ensure fair and accurate predictions across all classes.

Previous research suggest that addressing imbalance data can be categorized into three groups: external approaches (data level), internal approaches (algorithmic level), and hybrid approaches (Piyadasa & Gunawardana, 2023). External approaches aim to balance datasets using sampling methods, which may involve undersampling the majority class, oversampling the minority class, or a combination of both sampling, known as hybrid sampling. Internal approaches focus on enhancing the classification algorithm without altering the dataset. This approach can be done through cost-sensitive learning and ensemble methods. Hybrid approaches combine the strength of both external and internal approaches while considering their respective weakness. Researchers mostly used external approaches to address imbalance data due to the classifier independence and generalizability (López et al., 2013). In contrast, internal approaches result in the dataset becoming heavily dependent on the modified classifier.

For external approaches, various methods can be utilized for oversampling, undersampling, and hybrid sampling (Khushi et al., 2021). In oversampling, methods such as Random Oversampling (ROS) aim to increase the minority class samples by randomly duplicating existing ones. However, the ROS approach can lead to overfitting. The Synthetic Minority Oversampling Technique (SMOTE) address this issue by generating new samples through interpolation of existing minority class samples. While SMOTE reduce the risk of overfitting, it can introduce noisy and borderline samples, potentially causing overlap with other classes (Ali et al., 2019). Research by (Nekooeimehr & Lai-Yuen, 2016) highlights this issue and proposes Adaptive Weighted Semi-Unsupervised Weighted Oversampling (A-SUWO) as a solution. A-SUWO employs a semi-unsupervised hierarchical clustering approach and adaptively determines the oversampling size for each sub-cluster based on its classification complexity and cross-validation. Additionally, A-SUWO considers the majority class during both the clustering and oversampling stages.

For undersampling, Random Undersampling (RUS) is a basic undersampling method which randomly removes samples from the majority class based on a pre-defined sampling rate. However, RUS can lead to the loss of crucial information, potentially degrading the classifier performance. To address this issue, researchers have introduced the concept of informative undersampling, where only the least significant majority samples will be removed. An example of this approach is the concept of Tomek-Link pair, introduced by Ivan Tomek. A Tomek-Link pair is a boundary pair within a data distribution that is known to promote noise. Removing this pair can prevent the decision boundary from shifting in the wrong direction (Devi et al., 2020).

In hybrid sampling, one effective method is the combination of SMOTE and Tomek-Link. This combination takes the best from each method. While SMOTE addresses class imbalance by oversampling the minority class through interpolation, its weaknesses in overgeneralization and generation of noisy data can be mitigated by the Tomek-Link undersampling method. Tomek-Link cleans noisy data from the majority class that lies closest to the minority class. By combining SMOTE and Tomek Link, this hybrid method enhances accuracy better than either method alone (Hairani et al., 2023).

Many studies in the context of imbalanced datasets employ various sampling techniques and datasets, with the objective to examine differences in model performance before and after applying sampling methods (Ishaq et al., 2021; Mohammed et al., 2020; Sawangarreerak & Thanathamathee, 2020; Shamsudin et al., 2020). However, the limitation of these studies is that they often begin with originally imbalanced datasets. As a result, while improvements in performance are observed after applying sampling techniques, these studies lack a ”target” baseline for comparison, specifically the performance of the model when trained on a dataset that is originally balanced. The absence of a balanced reference point makes it difficult to evalute the effectiveness of the sampling methods in improving the model performance in comparison to a balanced dataset.

To address this issue and given the ability of A-SUWO and Tomek Link, this research will employ both methods to a body performance dataset sourced from Kaggle. Although the dataset is originally balanced, it will be simulated to be imbalanced to reflect real-world imbalance scenarios. The objective of this research is to identify the most effective sampling

techniques that can closely match the performance of the original balanced dataset when applied to the simulated imbalanced dataset. Three dataset distributions will be analyzed: (1) normal distribution, (2) simulated imbalance distribution, and (3) synthesized distribution using oversampling with A-SUWO, undersampling with Tomek Link, and hybrid sampling with both A-SUWO and Tomek Link. Classifiers, including Decision Tree, Random Forest, and KNN, will be utilized to evalute their performance using accuracy, precision, recall, and F1-score metrics. By systematically assessing these sampling techniques, the research seeks to determine the best approach for addressing the problem of class imbalance across diverse machine learning algorithms.

2. Literature Review

Traditional machine learning techniques, which often assume that the distribution of instances across classes is about equal, are very different from the field of learning from imbalanced data. However, in practical situations, certain groups are significantly more common than others, leading to an unequal distribution. Class imbalance is a major problem in machine learning, especially in fields like cancer diagnosis, fraud detection, and other applications where the minority class represents uncommon but extremely important cases. A class imbalance can have serious consequences for model performance and decision-making since it frequently results in biased learning models that prioritize the majority class while occasionally completely ignoring the minority class. Furthermore, the most important classes for research purposes are frequently the ones that are underrepresented in imbalanced datasets. In order to overcome the difficulties presented by unbalanced datasets, researchers have conducted extensive studies and developed algorithms to address the challenges posed by imbalanced datasets. Addressing this problem is crucial, as the reliability and applicability of machine learning systems in critical domains are directly impacted by their performance in imbalanced scenarios (Ghosh et al., 2024; Johnson & Khoshgoftaar, 2019).

(Mohammed et al., 2020) conducted experiments using the "Santander Customer Transaction Prediction" dataset, which features imbalanced binary classes aimed to predict specific customer transaction regardless of the transaction amount. The study compared two resampling methods, Random Oversampling (ROS) and Random Undersampling (RUS), to address the imbalance. Various classifiers were utilized for both resampling methods, including Support Vector Machine (SVM), Gaussian Naïve Bayes (NB), Decision Tree (DT), and Random Forest (RF). To measure performance, the study used various evalution metrics such as accuracy, precision, recall, F1-score, and ROC. The findings revealed that oversampling performed better than undersampling. Specifically, the RF classifier with ROS achieved an accuracy of 0.998, precision of 0.999, recall of 0.997, F1-score of 0.998, and ROC score of 0.998.

(Shamsudin et al., 2020) investigated the performance of a classification model by combining the method of oversampling and undersampling methods for detecting the fraud cases in the "credit card fraud detection" dataset. The study utilized Random Undersampling (RUS) and combined with various oversampling techniques, including SMOTE, ADASYN, Borderline SMOTE, SVM-SMOTE, and Random Oversampling (ROS). The Random Forest (RF) algorithm was used to evaluate the dataset, with precision, recall, and F1-measure as the evaluation metrics. The findings revealed that the combination of oversampling and undersampling methods improved the model's performance, with an average of 0.80% in precision, recall, and F1-measure value.

(Sawangarreerak & Thanathamathee, 2020) proposed a combined sampling technique using random oversampling and Tomek Link to improve the performance of imbalanced classification in university student depression data. The classifier used in this study is the Random Forest model. The dataset utilized in the study comprises 1,549 examples, along with an additional 165 examples unrelated to the original dataset, collected from accounting course students over four years. The performance of the model was evaluated using accuracy, precision, recall, and F-measure. The experiment results demonstrated that the proposed method outperformed individual sampling techniques, with an accuracy of 94.17%, an average precision of 91.18%, average recall of 92.08%, and an average F-measure of 91.62%.

(Ishaq et al., 2021) conducted an experiment on a cardiovascular patient survival dataset comprising 299 patients. The aim of the study was to identify significant features and effective data mining techniques to improve the accuracy of cardiovascular patient survival prediction. The study employed nine classifications models, such as Decision Tree (DT), Adaptive Boosting Classifier (AdaBoost), Random Forest (RF), Extra Tree Classifier (ETC), and many more. The models were evaluated using metrics such as accuracy, precision, recall, and F-score. SMOTE was utilized to address the class imbalance problem in the dataset. Additionally, significant features were selected using the RF model. The findings demonstrated that the ETC model outperforms other models, achieving an accuracy of 0.9262 accuracy with SMOTE.

(Mqadi et al., 2021) proposed using a data-point approach to address imbalanced credit card dataset with SMOTE in machine learning classifiers namely SVM, Logistic Regression, Decision Tree, and Random Forest. The classifiers was evaluated using precision, recall, F1-score, and average precision metrics. The results indicated that intially the model struggled to detect fraudulent transactions due to data imbalanced. However, after applying SMOTE, the predictive capabilities improved significantly, with the Random Forest and Decision Tree classifiers achieving the best performance. Specifically, precision score for the positive class improved by 10% for Random Forest and 42% for Decision Tree. Recall increased by 47% for Random Forest and 39% for Decision Tree, while F1-scores improved by 33% for Random Forest and 35% for Decision Tree.

(Apostolopoulos, 2020) utilized SMOTE to generate minority class instances in an imbalanced Coronary Artery Disease (CAD) dataset. Initially, the public Z – Alizadeh Sani dataset, used for non-invasive CAD prediction, was analyzed. Artificial Neural Networks (ANN), Decision Trees, and SVM were employed to classify both the original and augmented datasets, with accuracy as the evaluation metric. The findings demonstrated that Random Forest outperformed the other classifiers, achieving an accuracy improvement from 84.44% to 89.06% after applying SMOTE. However, it was noted that SMOTE augmentation should be deeply examined before being used on medical datasets.

Overall, the literature shows that there is no one-size-fits-all answer, even if great progress has been achieved in tackling the issues of class imbalance. When selecting a resampling or learning approach, concerns including dataset characteristics, domain-specific considerations, and computational cost should all be considered, according to critical research of the methodologies used in the works examined here. The theoretical and practical underpinnings of unbalanced learning research will be strengthened by incorporating new breakthroughs in online learning and hybrid models, as well as by doing a more comprehensive analysis of the ethical issues in high-stakes fields like healthcare.

This research adds to this continuing discussion by investigating how different sampling strategies affect model performance through a novel method by beginning with a balanced dataset and simulating imbalanced circumstances to establish a benchmark for assessment. To determine efficient strategies that reduce overfitting and maximize classification accuracy, the research compares the performance of hybrid approaches, Tomek Link undersampling, and Adaptive Semi-Unsupervised Weighted Oversampling (A-SUWO). The results highlight the significance of choosing and optimizing sampling strategies to match model performance with that of the original balanced dataset, demonstrating that Tomek Link undersampling obtained the highest test accuracy despite overfitting in other methods.

3. Proposed Methods

This study aims to compare outcomes derived from three distinct scenarios: (1) models trained on the initial balanced dataset, (2) models trained on the simulated imbalance dataset, and (3) models trained on the newly synthesized balance dataset created through various sampling techniques. The objective is to identify superior sampling techniques that can closely match the performance of the initial balanced distribution on the imbalanced dataset. The imbalanced dataset is a simulated version of the original balance body performance dataset, emulating a real-world scenario where data imbalance is a common issue. The procedural steps undertaken in this study are Illustrated in Fig 1.

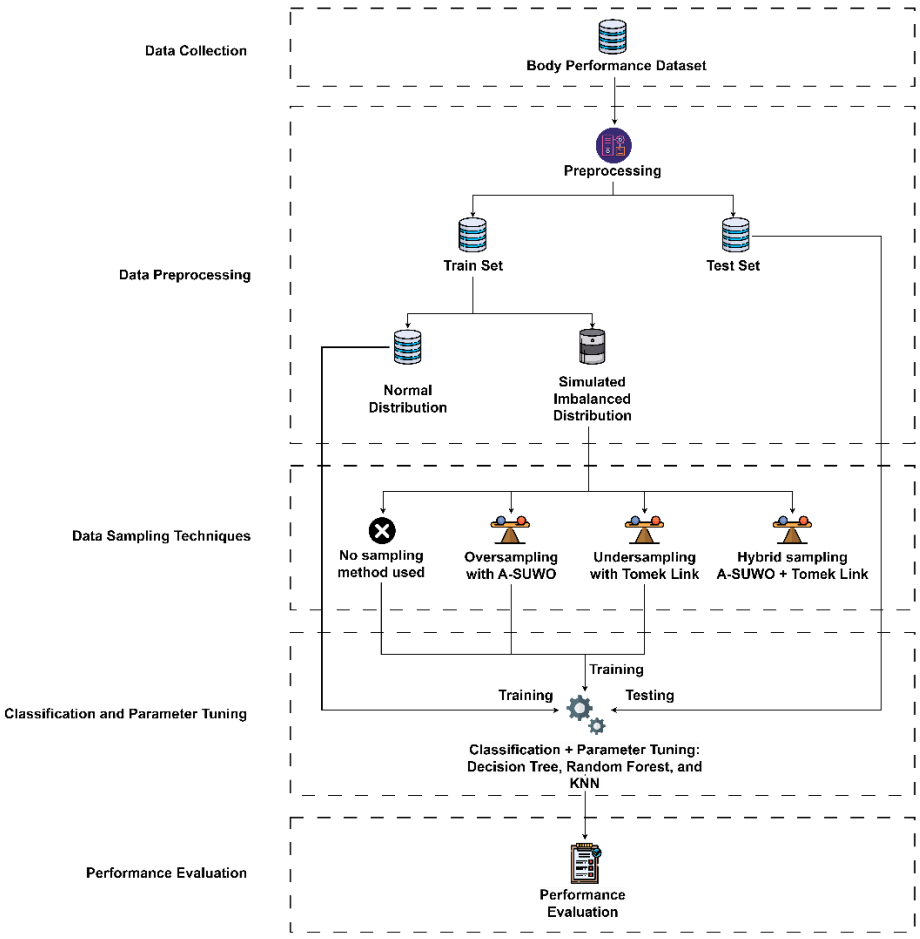

Fig. 1. The Proposed Study Approach

3.1. Data Collection

This study used the body performance dataset that contains performance grades corresponding to different age groups alongside additional performance-related information. The dataset was sourced from Kaggle website, encompasses 13,393 rows of data and 12 distinct attributes. The 'Class' attribute serves as the label for classification purposes. The description of each attribute is shown in Table 1.

Table 1 - Description of attributes in the body performance dataset

No	Attribute	Description
1	Age	Person's age in years
2	Gender	Person's gender
3	Height_cm	Person's height in centimeter (cm)
4	Weight_kg	Person's weight in kilogram (kg)
5	Body fat_%	Body fat percentage
6	Diastolic	Diastolic blood pressure
7	Systolic	Systolic blood pressure
8	GripForce	Hand grip strength
9	Sit and bend forward_cm	The length that can be reach when a person sits and bends forward
10	Sit-ups count	Number of sit ups
11	Broad jump_cm	The horizontal jump distance in centimeter (cm)
12	Class	Grade of performance (A, B, C, D). 'A' indicated as best grade

The Body Performance dataset was selected for this research to enable a comparative analysis between scenarios where algorithms are used to address class imbalance and a well-balanced dataset. This dataset is ideal for controlled experimentation, as it initially depicts a balanced distribution, allowing the simulation of imbalanced distributions to mimics real-world imbalanced situations scenarios. This approach maintains the ability to compare outcomes

against the initial balanced distribution, ensuring a consisten and rigorous evaluation of the effectiveness of various balancing methods.

3.2. Data Preprocessing

Before implementing sampling techniques and evaluating the model's performance, preprocessing steps are crucial due to the typical incompleteness and diversity in real-world data, which often involves aggregated, noisy, or missing values. In the body performance dataset, categorical attributes were transformed into numerical ones (e.g. transforming the 'Class' attribute from A, B, C, D to 0, 1, 2, 3), and then dataset was partitioned based on its class labels, enabling a more precise Identification of outliers. By utilizing boxplot on each segmented dataset, numerous outliers were detected across various attributes. Given outliers potential impact on model performance, this study addresses outliers by replacing their values with the median value for each attribute. After handling outliers, the partitioned dataset was merged and undergoes normalization. Specifically, min-max normalization is employed, standardizing the diverse attribute ranges to a specific interval, usually between 0 and 1. The min-max normalization formula can be seen in Eq.(1):

$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

Following the implementation of min-max normalization, the dataset will be split, with 80% assigned to the training set and 20% to the test set. The training set will then be categorized into two different versions: (1) a normal distribution and (2) a simulated imbalanced distribution. The original body performance dataset is already balanced. However, for this study, the dataset will be simulated to be imbalanced. The simulated imbalanced distribution allows researchers to compare model performance on the balanced dataset as a reference to the simulated imbalanced distribution. By simulating the imbalanced distribution of the original dataset, researchers aim to identify the most suitable sampling method that can give performance close to the model's performance on the original balanced dataset.

The simulated imbalance dataset will be created by adjusting the distribution percentage for each 'Class' attribute. These adjustments are determined based on the results of precision and recall obtained from the original dataset. The objective of this approach is to achieve a balance between precision and recall within the dataset, thereby enhancing the overall results. The process for simulating imbalance in the original dataset are as follows.

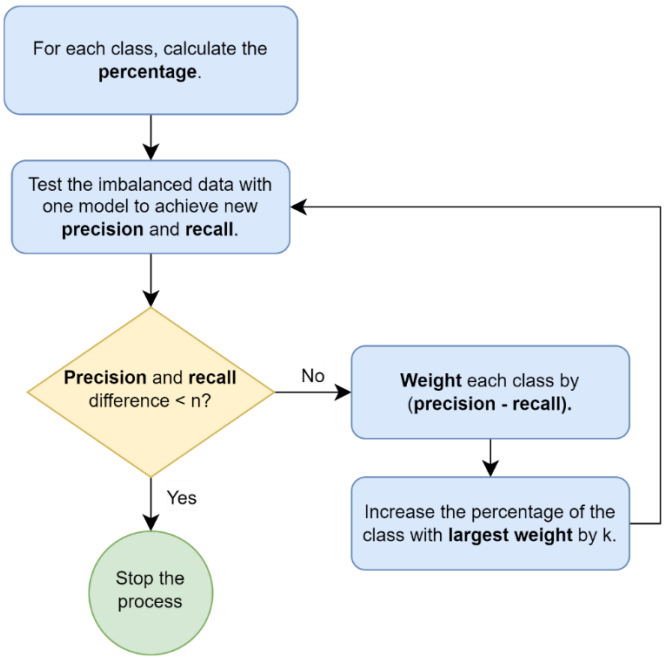

Fig. 2. Process For Simulating The Imbalanced Data Distribution

In this study, both value of *n* and *k* are set to 0.10 or 10%. The chosen value is to ensure a manageable and moderate difference. To calculate the percentage of each class, a process based on the difference between precision and recall will be utilized. The process is outlined as follows.

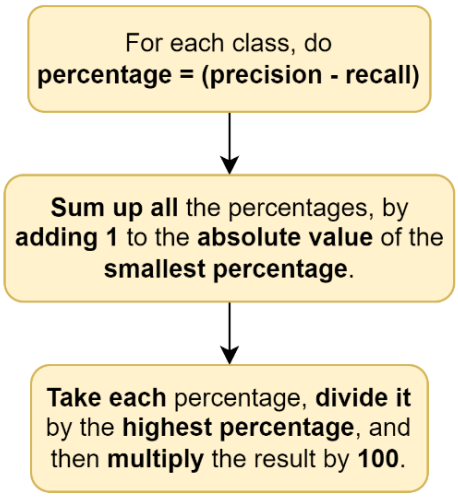

Fig. 3. Process For Calculating The Percentage Of Each Class

Based on the approach above and the result within Section 4, Table 9, the percentage of preserved data of Class within simulated imbalanced class of Body Performance data, would be 32% for Class A, 44% for Class B, 60% for Class C, 100% of Class D. The proportion would be looked as follows.

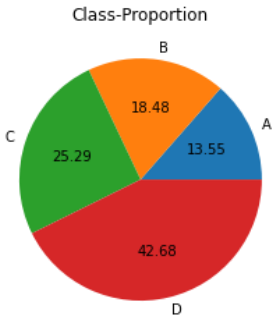

Fig. 4. The Simulated Imbalanced Distribution

3.3. Data Sampling Techniques

3.3.1. Oversampling: Adaptive Semi-Unsupervised Weighted Oversampling (A-SUWO)

A-SUWO, introduced by Nekooeimehr and Lai-yuen (Nekooeimehr & Lai-Yuen, 2016), is an oversampling method design to address imbalanced datasets. This method employs a semi-unsupervised hierarchical clustering approach to cluster minority instances. It will dynamically determine the oversampling size for each sub-cluster by considering its classification complexity and cross-validation. Oversampling the minority instances will be based on their Euclidean Distance to the majority class. The primary goal of A-SUWO is to identify hard-to-learn instances, focusing on those minority instances from each sub-cluster that reside close to the decision boundary. Moreover, A-SUWO ensures the avoidance of generating synthetic minority samples that might overlap with the majority class by considering the majority class during both the clustering and oversampling phases.

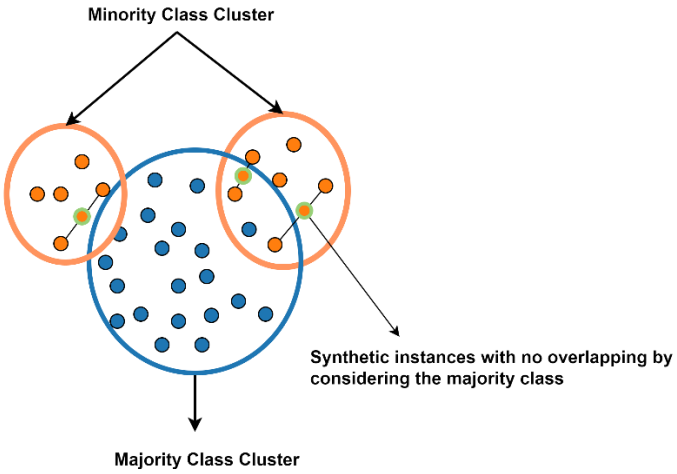

Fig. 5. Adaptive Semi-Unsupervised Weighted Oversampling (A-SUWO)

The use of the Adaptive Semi-Unsupervised Weighted Oversampling (A-SUWO) technique to rectify class imbalance in a dataset is illustrated in Figure 5. In this illustration, the orange circles represent examples of the majority class, and the blue dots represent cases of the minority class. Sub-clusters within the majority class are shown by the smaller orange circles, while the cluster of minority class samples is highlighted by the huge blue circle. The synthetic samples produced by the A-SUWO technique, which aims to create samples close to the decision border where minority class instances are underrepresented, are depicted by the green dots inside these sub-clusters. In order to prevent the introduction of noisy or redundant instances, A-SUWO dynamically modifies the quantity of synthetic samples for each sub-cluster, giving priority to regions with higher classification complexity. It also makes sure that the generated synthetic samples do not overlap with the majority class. By carefully placing the synthetic samples, the classifier is better able to distinguish between the classes, which solves the problems caused by unbalanced datasets and increases model accuracy.

A-SUWO is superior to conventional oversampling methods in a number of ways. Through the use of a semi-unsupervised hierarchical clustering technique, A-SUWO dynamically modifies the degree of oversampling according to the classification complexity of each sub-cluster, thereby adapting to the dataset's complexities. In order to address the problem of class imbalance, where simple random oversampling could result in overfitting or the creation of redundant synthetic samples, this makes sure that the most difficult minority instances—those close to the decision boundary—are oversampled more successfully. Furthermore, A-SUWO avoids the creation of synthetic samples that might overlap with the majority class, a typical issue with many oversampling techniques, by accounting for the majority class during both the clustering and oversampling phases. This method improves the generalization of the model, making A-SUWO an effective solution for imbalanced data. The method’s ability to balance the dataset in a targeted and thoughtful way, while avoiding common pitfalls like noisy or biased synthetic data, makes it particularly well-suited for real-world applications where imbalanced data is prevalent (Sun et al., 2020).

To implement A-SUWO, the smote-variants package was utilized. This package provides a python implementation of 85 different oversampling techniques that can be used for developing solutions in the field of imbalanced learning (Kovács, 2019). By utilizing the smote-variants package, researchers can experiment with various oversampling techniques, unlike previous oversampling techniques which offered only a limited number of open-source options. Based on the smote-variants package documentation, Table 2 provides the specific parameter values used to implement A-SUWO in this study.

Table 2 - Parameters for A-SUWO implementation

Method	Parameters
A-SUWO	oversampler: A-SUWO, oversampler_params: (random_state: 42, n_neighbors: 3)

3.3.2. Undersampling: Tomek Link

Developed by Ivan Tomek (Liu et al., 2018), Tomek Link is an undersampling method that works by removing instances from the majority class that are close to the minority class. Tomek Link utilized the nearest neighbor rule to find samples that should be removed. Considered as an improved Condensed Nearest Neighbor (CNN), Tomek Link detects pairs of data points (x_1, x_2) where x_1 belongs to minority class and x_2 represents the majority class. By calculating the Euclidean distance $d(x_1, x_2)$, a pair of (x_1, x_2) is considered as Tomek Link if no other instance x_3 exists where $d(x_1, x_3) < d(x_1, x_2)$ or $d(x_2, x_3) < d(x_1, x_2)$ (E. F. ; Swana et al., 2022). The visual representation of Tomek Link is shown in Fig 3.

By eliminating instances of the majority class that are close to the minority class's decision boundary, Tomek Link is a useful undersampling method that tackles the problem of class imbalance. The Condensed Nearest Neighbor (CNN) algorithm is enhanced by this technique, which focuses on "borderline" majority class samples that are near minority class occurrences. Tomek Link's main advantage is that it may tighten the decision boundary without removing important data because it only eliminates majority class instances that are most likely to cause noise and obstruct model training. Tomek Link makes sure that only situations where the majority class sample is too close to the minority class are eliminated by identifying pairs of instances where one belongs to the minority class and the other to the majority class, then calculating their Euclidean distance. By successfully minimizing the overlap between the two classes, this procedure produces a model that is more accurate and has better generalization. Furthermore, Tomek Link minimizes the possibility of data loss, a typical problem with other undersampling techniques, while maintaining the overall distribution of the majority class by concentrating on the examples that are closest to the decision boundary (E. F. Swana et al., 2022).

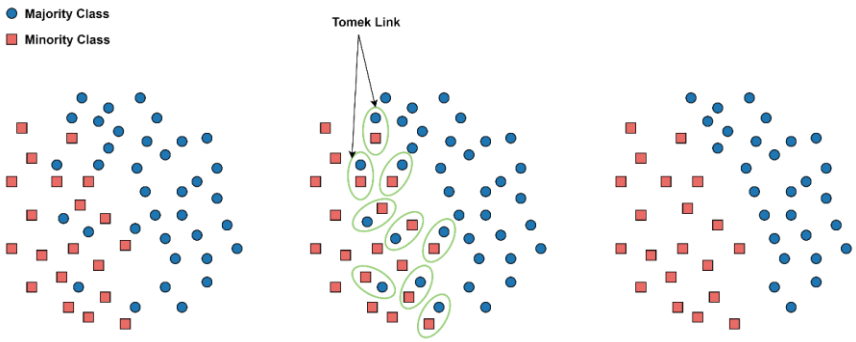

Fig. 6. Tomek Link

With red squares signifying the dominant class and blue circles representing the minority class, Fig 6 illustrates how to use the Tomek Link undersampling technique to address class imbalance in a dataset. Although there are few examples of the majority class near the decision boundary, the majority class is generally situated distant from the minority class in the first panel (left). Tomek Links, or pairs of points where one belongs to the minority class and the other to the majority class based on their Euclidean distance, are highlighted in the second panel (middle) as key majority class examples. These occurrences are eliminated because they are either unnecessary or noisy. The Tomek Link technique refines the decision border between the two classes by eliminating instances of the indicated majority class, as seen in the final panel (right). By decreasing overlap, minimizing noise, and concentrating the learning process on the more instructive examples, this procedure improves classifier performance and eventually produces a more accurate model.

To implement Tomek Link, the imbalanced-learn package is utilized. This package offers a number of re-sampling techniques commonly used to address class imbalance in datasets (Lemaître et al., 2016). Based on the imbalanced-learn package documentation, Table 3 presents the specific parameter value used to implement Tomek Link in this study.

Table 3 - Parameters for Tomek Link implementation

Method	Parameters
Tomek Link	sampling_strategy: not minority

3.4. Classification and Parameter Tuning

3.4.1. Classification

In this study, after preprocessing the dataset and using sampling methods to address imbalances, three models are proposed to handle both normal and imbalanced distribution scenario in the body performance dataset, which are Decision Tree, Random Forest, and KNN.

3.4.1.1. Decision Tree

Decision Tree is a machine learning algorithm that organizes data in a hierarchical, tree-like structure such that nodes represent as features, branches symbolize decision paths, and leaf nodes show outcome classes. This algorithm creates a model like tree that uses input variables to create multiple predictions on each leaf node (Rajaguru & Sannasi Chakravarthy, 2019). The Decision Tree structure is visually represented in Fig 7.

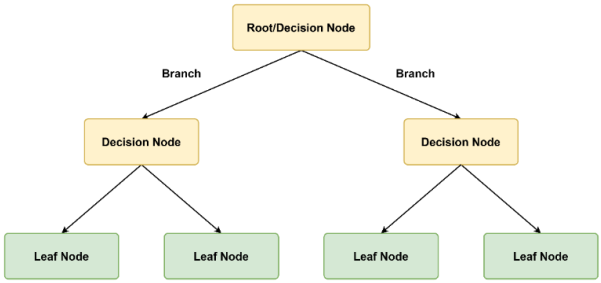

Fig. 7. Decision Tree

The Decision Tree approach was used for this investigation due to its low computational complexity and suitability for evaluating complicated datasets (Priyanka & Kumar, 2020). Decision trees are very effective for exploratory research since they are one of the fastest methods for creating or detecting new features. Furthermore, since the approach is robust to missing values and unaffected by outliers, it necessitates relatively fewer steps for data pretreatment and cleaning. Due to these characteristics, Decision Trees are a great option for assessing the dataset since they strike a compromise between performance, interpretability, and simplicity, guaranteeing accurate findings even when dealing with imperfect or raw data (Bansal et al., 2022).

3.4.1.2. Random Forest

Random Forest, developed by Breiman in 2001, is a supervised classification algorithm that works by creating multiple *n* decision trees from random subsets of the training data and predictor variables. Each tree's result is combined to make the final prediction, resulting in substantially higher accuracy compared to a single decision tree model (Speiser et al., 2019). Additionally, increasing the number of trees enhances accuracy and reduces overfitting risk, unlike other methods (Madeeh & Abdullah, 2021). The visual representation of Random Forest is depicted in Fig 8.

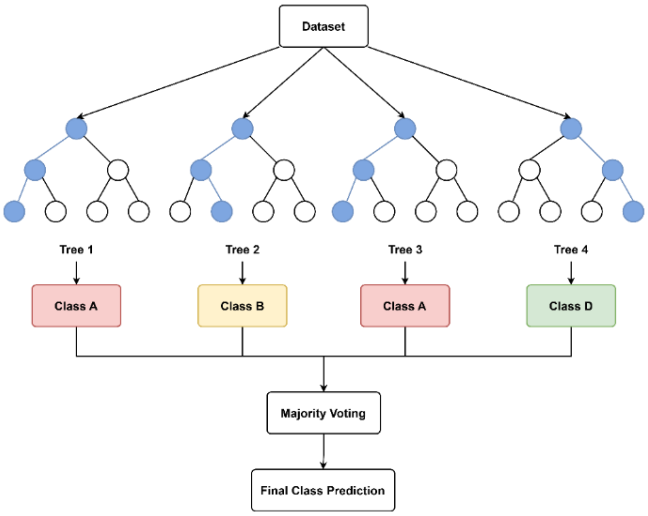

Fig. 8. Random Forest

Random Forest is a widely used ensemble method known for its consistent performance across various machine learning tasks, which is why this study selected it. As an ensemble technique, Random Forest enhances accuracy and reduces overfitting by aggregating the predictions from multiple decision trees. Due to its simplicity, interpretability, and ability to handle large datasets with diverse features, it is considered one of the leading algorithms and often outperforms some deep learning models in specific applications (Grinsztajn et al., 2022).

3.4.1.3. K-Nearest Neighbor (KNN)

KNN is a supervised machine learning algorithm used mostly for classification tasks. The model has a variable parameter, *k*, which quantifies the nearest points or neighbors considered. KNN identifies k nearest neighbors by calculating distances between points using standard Euclidean distance *d(x, y)*. After evaluating these *k* nearest neighbors, a majority voting rule is used to determine the class occurrence. The class with the most number of observations with its neighbor is the final classification (Madeeh & Abdullah, 2021; Uddin et al., 2022a).

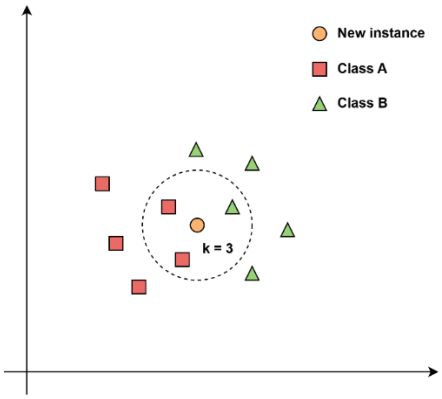

Fig 9. K-Nearest Neighbor (KNN)

As depicted in Fig 9, employing a value of k = 3 determines the number of nearest neighbors taken into consideration for the new instance. In this case, it identifies 3 nearest neighbors, where 2 belong to class A and 1 belongs to class B. The KNN algorithm will then utilize the majority voting rule to classify the new instance as class A.

This study employed the K-Nearest Neighbors (KNN) algorithm due to its simplicity and widespread use in machine learning. KNN is one of the easiest algorithms, utilizing instance-based learning to make predictions based on the similarity to neighboring data points. Its simplicity and ease of understanding make it a popular choice for various machine learning applications (Uddin et al., 2022b). Moreover, KNN is highly adaptable and performs well in situations where there is a non-linear relationship between features and target labels (Halder et

al., 2024). With these advantages, KNN is a valuable addition to this study for analyzing the dataset and comparing the performance of different algorithms.

3.4.2. Parameter Tuning

Table 4 - List of parameters of each model

Model	Parameters
Decision Tree	criterion, max_depth, min_samples_split, min_samples_leaf
Random Forest	n_estimators, criterion, max_depth, min_samples_split, min_samples_leaf
KNN	n_neighbors, weights, metric

Table 4 provides a list of parameters for each model, where the optimal values will be found by using Grid Search. In this study, various scenarios will be explored, including normal distribution, imbalanced distribution, oversampling with A-SUWO, undersampling with Tomek Link, and hybrid sampling with A-SUWO and Tomek Link. Detailed information regarding these parameter values for each scenario can be found in Section 4.

3.5. Performance Evaluation

The evaluation of the model's performance will utilize the fundamental concept of a confusion matrix. The confusion matrix is a tabular representation employed to illustrate how well a classification method performs on a dataset where the actual values are known. It effectively displays the classification outcomes for true and false instances (Hairani et al., 2023). The table representing the confusion matrix table is presented in Table 5.

Table 5 - Confusion matrix table

	Actual (+)	Actual (-)
Prediction (+)	True Positives (TP)	False Positives (FP)
Prediction (-)	False Negatives (FN)	True Negatives (TN)

The accuracy metric is a commonly used measure that assesses the overall correctness of a model's prediction. However, when addressing the imbalance problem in datasets, relying solely on accuracy for evaluating the model's performance is inadequate due to the model might achieve a higher accuracy for the majority class but lower accuracy for the minority class. Therefore, to provide a more comprehensive evaluation, precision, recall, and f1-score will also be utilized as evaluation metrics in this study.

Precision evaluates the model's ability to precisely predict positive classes from the total positive class predictions. Meanwhile, recall measures the model's accuracy in correctly identifying positive class from the total actual positive class. The f1-score represents the harmonic mean of precision and recall, providing a balanced assessment of both metrics. The formula for each metric can be seen in Eq.(2), Eq.(3), Eq.(4), Eq.(5):

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

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

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

$$F1\ score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{5}$$

4. Results and Discussions

Normal Distribution

In the initial phase of experimentation, the original dataset, which exhibited a normal distribution, was utilized to assess the performance of the proposed models. A hyperparameter optimization process was conducted using Grid Search Cross Validation (CV) to identify the optimal combination of hyperparameters for each model. This procecss ensures the best-suited

hyperparameters are selected for each model. Table 6 provides a comprehensive summary of the parameter values selected for each model, as determined through the Grid Search CV procedure. Additionally, it includes the corresponding training accuracy achieved for each parameter configuration.

Table 6 - Parameters and Train Accuracy For The Original Dataset With Normal Distribution

Classifier	Parameters	Train Accuracy
Decision Tree	criterion: gini, max_depth: 12, min_samples_leaf: 10, min_samples_split: 2	0.67911
Random Forest	criterion: entropy, max_depth: 40, min_samples_leaf: 1, min_samples_split: 2, n_estimators: 700	0.73502
KNN	metric: l1, n_neighbors: 28, weights: distance	0.64420

The testing evaluation results for the normal distribution dataset, as shown in Table 7, indicate that Random Forest achieves the highest test accuracy of 0.74804, outperforming both the Decision Tree and KNN classifiers. This superior performance can be attributed to Random Forest's ensemble learning approach, which aggregates the predictions of several decision trees trained on arbitrary subsets of the data. This approach mitigates the risk of overfitting and enhance the model's ability to generalize effectively to new and unseen data. Additionally, Random Forest performs well across all classes, as evidenced by its consistently high precision and recall scores in Table 7, which are a result of its capacity to assess feature relevance and manage intricate feature interactions. In contrast, the decision tree model exhibits lower performance, with a test accuracy of 0.69279, due to its tendency to overfit the data. A single Decision Tree model can quickly grow overly intricate and sensitive to even slight changes in the data, leading to suboptimal performance, especially in classes where the model's decisions might not generalize well. Similarly, KNN demonstrates the lowest test accuracy of 0.64725, primarily due to its dependence on the distance between data points. This dependence can be greatly impacted by the scale and dispersion of the data.

Table 7 - Testing Evaluation of Various Models For The Original Dataset With Normal Distribution

Classifier	Class	Precision	Recall	F1-Score	Test Accuracy
Decision Tree	0	0.68	0.82	0.74	0.69279
	1	0.58	0.57	0.58	
	2	0.68	0.62	0.65	
	3	0.87	0.77	0.81	
Random Forest	0	0.71	0.86	0.78	0.74804
	1	0.64	0.62	0.63	
	2	0.78	0.68	0.73	
	3	0.88	0.84	0.86	
KNN	0	0.63	0.83	0.72	0.64725
	1	0.52	0.51	0.52	
	2	0.61	0.58	0.59	
	3	0.91	0.67	0.77	

Simulated Imbalance Distribution

In the second experiment, the simulated imbalance dataset is employed to evaluate the performance of the proposed models. Grid Search CV is utilized to identify the optimal parameter values used for each model. The corresponding training accuracy achieved for each parameter configuration is presented in Table 8.

Table 8 - Parameters and Train Accuracy For The Simulated Imbalanced Distribution

Classifier	Parameters	Train Accuracy
Decision Tree	criterion: gini, max_depth: 10,	0.66624

	min_samples_leaf: 12, min_samples_split: 2	
Random Forest	criterion: entropy, max_depth: 30, min_samples_leaf: 1, min_samples_split: 2, n_estimators: 350	0.73448
KNN	metric: l1, n_neighbors: 62, weights: distance	0.66703

After testing all proposed models in this study, as depicted in Table 9, it is evident that the imbalance nature of the data negatively affects the performance of each model, leading to deteriorated accuracy across all metrics. However, by creating a custom preserved data percentage for each 'Class' column, as explained in Section 3.2, the precision and recall metrics, particularly for the KNN model, have been stabilized. The next step involves the implementation of oversampling and undersampling methods to address data imbalance and further enhance the overall model accuracy.

Table 9 - Testing Evaluation of Various Models For The Simulated Imbalanced Distribution

Classifier	Class	Precision	Recall	F1-Score	Test Accuracy
Decision Tree	0	0.68	0.71	0.69	0.66405
	1	0.55	0.49	0.52	
	2	0.59	0.69	0.63	
	3	0.87	0.77	0.82	
Random Forest	0	0.74	0.78	0.76	0.72266
	1	0.63	0.52	0.57	
	2	0.65	0.78	0.71	
	3	0.88	0.82	0.85	
KNN	0	0.67	0.75	0.71	0.63307
	1	0.52	0.44	0.48	
	2	0.53	0.73	0.62	
	3	0.93	0.61	0.74	

Oversampling with A-SUWO

In response to the decline in model performance resulting from the simulated imbalanced distribution of the original dataset, several sampling techniques were utilized. The first technique employed was oversampling using the Adaptive Semi-Unsupervised Weighted Oversampling (A-SUWO) method. Table 10 presents the optimal parameter values determined through Grid Search CV for each model in the A-SUWO oversampling scenario, along with the corresponding training accuracy achieved for each configuration. The training accuracy results demonstrate that A-SUWO significantly improved model performance compared to the imbalanced training data, with an average accuracy increase of approximately 11%. This highlights the effectiveness of employing A-SUWO to improve the model's learning process.

Table 10 - Parameters and Train Accuracy For The Oversampling Case Using A-SUWO

Classifier	Parameters	Train Accuracy
Decision Tree	criterion: entropy, max_depth: 18, min_samples_leaf: 1, min_samples_split: 2	0.76729
Random Forest	criterion: entropy, max_depth: 30, min_samples_leaf: 1, min_samples_split: 2, n_estimators: 900	0.85204
KNN	metric: l1, n_neighbors: 4, weights: distance	0.77227

Despite the training data showing an improvement of over 11% on average, the testing data, as shown in Table 11, shows a significant decline with A-SUWO, dropping by an average of 4%. The gap between training and testing accuracies within A-SUWO is about 17%. The significant gap suggests that A-SUWO may contribute to overfitting, due to the notable difference in accuracy between the training and testing datasets. Following the performance of A-SUWO as an oversampling technique, the next technique to be examined involves the implementation of the undersampling technique using Tomek Link on the simulated imbalance dataset.

Table 11 - Testing Evaluation of Various Models For The Oversampling Case Using A-SUWO

Classifier	Class	Precision	Recall	F1-Score	Test Accuracy
Decision Tree	0	0.68	0.59	0.63	0.59686
	1	0.49	0.45	0.47	
	2	0.54	0.54	0.54	
	3	0.67	0.83	0.74	
Random Forest	0	0.73	0.74	0.74	0.70996
	1	0.60	0.55	0.57	
	2	0.70	0.69	0.70	
	3	0.79	0.87	0.83	
KNN	0	0.63	0.61	0.62	0.57969
	1	0.44	0.48	0.46	
	2	0.51	0.50	0.50	
	3	0.79	0.74	0.76	

Undersampling with Tomek Link

Table 12 presents the parameter values utilized for all proposed models for the undersampling scenario employing Tomek Link, along with the corresponding training accuracy achieved for each configuration. The results indicate an average improvement in accuracy of over 4% when using Tomek Link, compared to the accuracy obtained with the imbalanced training data. However, this improvement is lower than the average training accuracy achieved using the A-SUWO technique.

Table 12 - Parameters and Train Accuracy For The Undersampling Case Using Tomek Link

Classifier	Parameters	Train Accuracy
Decision Tree	criterion: entropy, max_depth: 12, min_samples_leaf: 18, min_samples_split: 2	0.70994
Random Forest	criterion: entropy, max_depth: 30, min_samples_leaf: 1, min_samples_split: 2, n_estimators: 900	0.76941
KNN	metric: l1, n_neighbors: 26, weights: distance	0.71303

After evaluating all proposed models on testing data, as presented in Table 13, a decline in performance is observed when using the synthesized dataset with Tomek Link, dropping by an average of around 2%. The gap between training and testing data with Tomek Link is about 8%. This implies that Tomek Link may be contributing to overfitting, given the notable difference between training and testing data accuracies.

Table 13 - Testing Evaluation of Various Models For The Undersampling Case Using Tomek Link

Classifier	Class	Precision	Recall	F1-Score	Test Accuracy
Decision Tree	0	0.64	0.74	0.68	0.63269
	1	0.52	0.38	0.44	
	2	0.60	0.60	0.60	
	3	0.73	0.83	0.78	
Random Forest	0	0.69	0.79	0.74	0.68607

	1	0.62	0.41	0.49	
	2	0.67	0.67	0.67	
	3	0.73	0.89	0.80	
KNN	0	0.64	0.76	0.70	
	1	0.52	0.34	0.41	0.63195
	2	0.57	0.62	0.60	
	3	0.76	0.82	0.79	

Comparing the testing accuracy results between Tomek Link and A-SUWO, as shown in Table 13 and Table 11, shows that most models, except Random Forest, have demonstrated improved testing accuracies. The Decision Tree model increased by 3%, KNN by 5%, while Random Forest decreased by 2%. Additionally, after comparing the training accuracy results between Tomek Link and A-SUWO, as depicted in Table 12 and Table 10, there is an increase in training accuracy across models, except for the Random Forest model. This suggests that the decline in Random Forest performance might be caused by overfitting. The more the model overfits, the more it defies the testing performance.

Hybrid Sampling with A-SUWO and Tomek Link

Table 14 provides a comprehensive overview of the training accuracy across all models and the list of parameters values utilized for the final sampling technique using combination of both A-SUWO and Tomek Link.

Table 14 - Parameters and Train Accuracy For The Hybrid Sampling Case Using A-SUWO and Tomek Link

Classifier	Parameters	Train Accuracy
Decision Tree	criterion: entropy, max_depth: 20, min_samples_leaf: 1, min_samples_split: 2	0.80980
Random Forest	criterion: entropy, max_depth: None, min_samples_leaf: 1, min_samples_split: 2, n_estimators: 900	0.88648
KNN	metric: l1, n_neighbors: 3, weights: distance	0.84179

Comparing the training accuracy results between hybrid sampling with A-SUWO and Tomek Link and the training accuracy achieved with the simulated imbalance data, shows an average improvement of over 15%, making it the highest average training accuracy compared with other sampling methods used in this study. The training results demonstrate the effectiveness of combining A-SUWO and Tomek Link sampling methods in improving the model's ability to learn from the dataset.

Table 15 - Testing evaluation of various models for the hybrid sampling case using A-SUWO and Tomek Link

Classifier	Class	Precision	Recall	F1-Score	Test Accuracy
Decision Tree	0	0.66	0.65	0.65	
	1	0.53	0.42	0.47	0.62112
	2	0.58	0.61	0.59	
	3	0.69	0.82	0.75	
Random Forest	0	0.71	0.78	0.75	
	1	0.61	0.51	0.56	0.70959
	2	0.70	0.69	0.69	
	3	0.80	0.87	0.83	
KNN	0	0.58	0.65	0.61	
	1	0.43	0.40	0.41	0.57633
	2	0.52	0.53	0.52	
	3	0.78	0.74	0.76	

Through rigorous testing of the proposed models, as depicted in Table 15, there was a consistent decline in testing accuracy compared to the testing accuracy from the simulated,

averaging over 3%. Additionally, the gap between the training and testing accuracy is approximately 21%, a strong indication of overfitting in the results. This implies that while employing A-SUWO and Tomek Link as sampling techniques can enhance the model's learning capabilities of the data, it also introduces the risk of overfitting.

The findings show that class imbalance is a significant challenge to machine learning algorithms, as evidenced by the decline in performance of all classifiers when used with the generated imbalanced data. The risk of overfitting was apparent, with testing accuracy sharply declining, even though oversampling with A-SUWO and undersampling with Tomek Link demonstrated possible gains in training accuracy. This implies that although these sampling techniques can improve models' learning from the unbalanced data, they may also make models overly specific to the training set, which would impair their capacity to generalize to new data. Overfitting can result in subpar decision-making in real-world situations when predicted accuracy on fresh data is essential. Therefore, in order to reduce overfitting and enhance generalizability, the results emphasize the necessity of carefully balancing training and testing accuracy as well as the investigation of hybrid approaches that mix oversampling and undersampling techniques with other strategies. According to the findings of this study, data balancing methods such as A-SUWO and Tomek Link can improve model performance, but they should only be used sufficiently, especially in situations when there is a lot of imbalances, to prevent diminishing returns in real-world applications.

Discussion

Upon comparing the testing accuracy results between hybrid sampling using A-SUWO and Tomek Link and oversampling using A-SUWO alone, as shown in Table 15 and Table 11, it is evident that the testing accuracy of the Decision Tree has increased by 2%. In contrast, the training accuracy of the other proposed models has experienced a slight decrease. This suggests that there is no significant improvement between the use of A-SUWO alone and the hybrid sampling approach using A-SUWO and Tomek Link. Further examination of the testing accuracy results between hybrid sampling using A-SUWO and Tomek Link and undersampling using Tomek Link alone, as shown in Table 15 and Table 13, the testing accuracy of the KNN model has declined by 6%, the Decision Tree by 2%, while the Random Forest has shown an increase by 2%. This indicates that the implementation of the hybrid sampling approach using A-SUWO and Tomek Link only leads to an overall degradation in testing accuracy. By analyzing the three applied methods, namely A-SUWO, Tomek Link, and both combined, it can be inferred that all these approaches contribute to overfitting and an overal deterioration in performance. This is evident from the consistent decline in general testing accuracy across all sampling outcomes in comparison to the simulated imbalance scenario. Consequently, it becomes apparent that neither A-SUWO nor Tomek Link is well-suited for the simulated imbalanced dataset.

The study's findings are in line with earlier investigations into the effects of resampling methods when dealing with unbalanced datasets. (Mohammed et al., 2020) for instance, discovered that Random Oversampling (ROS) enhanced Random Forest performance, which is comparable to the gains observed with A-SUWO in this investigation. The enhanced training accuracy seen in this work is consistent with (Shamsudin et al., 2020) demonstration that combining oversampling and undersampling techniques could enhance precision, recall, and F1-score.

The results of (Mqadi et al., 2021), who documented performance gains with undersampling techniques, are consistent with the undersampling method with Tomek Link used in this study. However, (Sawangarreerak & Thanathamathee, 2020) point out that undersampling might not always be the best option, as evidenced by the minor decline in performance with Random Forest utilizing Tomek Link. Overall, the results of this study support several findings in the literature and emphasize how crucial it is to balance sample strategies to prevent overfitting and enhance generalization.

5. Conclusion

In this study, after simulating the distribution of the original dataset to be imbalance, it is evident that the presence of imbalance data can affect the performance of classification models. This can be seen by the discernible decline in metric results across all models. For example, a specific metric such as the accuracy value of the Random Forest model decreases from 0.74804 on the original dataset to 0.69690 on the simulated imbalanced dataset, highlighting the importance of sampling techniques to address this issue. Various sampling techniques were applied, including oversampling with A-SUWO, undersampling with Tomek Link, and hybrid sampling using both A-SUWO and Tomek Link. Based on the proposed sampling techniques, it was found that employing A-SUWO and Tomek Link on the body performance data might lead to overfitting, as evidenced by the significant gap between training and testing accuracy. As an example, in the hybrid sampling case, the accuracy metric of the Random Forest model was 0.88648, while its testing accuracy was 0.70959. This overfitting also led to general performance degradation, as seen by the drop in testing performance in the simulated imbalance and across all three methods.

Despite the existence of overfitting when using A-SUWO and Tomek Link, a comparison of all experimented results indicated that the undersampling with Tomek Link produce better metric results than other methods. This is evident from the average testing score of 0.65023 for Tomek Link, compared to 0.62883 for A-SUWO and 0.63568 for the hybrid sampling approach. The study suggests that extending the implementation of A-SUWO and Tomek Link to other imbalanced datasets with varying imbalance ratios and data sizes would provide a more comprehensive evaluation of both methods. Additionally, exploring the combination of other oversampling and undersampling techniques could help identify optimal combinations suitable for diverse datasets.

Open Data & Contributorship

The dataset used in this research can be retrieved in https://bit.ly/bodyPerformanceData. Abba Suganda Girsang is responsible for supervising the research project, offering guidance, and feedback on both the experiments and preparation of the manuscript. Febryan Grady and Joel Rizky Wahidiyat are responsible for conducting data collection, data preprocessing, experiments, and writing the manuscript.

References

Aguiar, G., Krawczyk, B., & Cano, A. (2024). A survey on learning from imbalanced data streams: taxonomy, challenges, empirical study, and reproducible experimental framework. *Machine Learning*, *113*(7), 4165–4243. https://doi.org/10.1007/S10994-023-06353-6/FIGURES/45

Ali, H., Mohd Salleh, M. N., Saedudin, R., Hussain, K., & Mushtaq, M. F. (2019). Imbalance class problems in data mining: a review. *Indonesian Journal of Electrical Engineering and Computer Science*, *14*(3), 1552. https://doi.org/10.11591/ijeecs.v14.i3.pp1552-1563

Amin, A., Anwar, S., Adnan, A., Nawaz, M., Howard, N., Qadir, J., Hawalah, A., & Hussain, A. (2016). Comparing Oversampling Techniques to Handle the Class Imbalance Problem: A Customer Churn Prediction Case Study. *IEEE Access*, *4*, 7940–7957. https://doi.org/10.1109/ACCESS.2016.2619719

Apostolopoulos, I. D. (2020). Investigating the Synthetic Minority class Oversampling Technique (SMOTE) on an imbalanced cardiovascular disease (CVD) dataset. *International Journal of Engineering Applied Sciences and Technology*, *04*(09), 431–434. https://doi.org/10.33564/ijeast.2020.v04i09.058

Bansal, M., Goyal, A., & Choudhary, A. (2022). A comparative analysis of K-Nearest Neighbor, Genetic, Support Vector Machine, Decision Tree, and Long Short Term Memory algorithms in machine learning. *Decision Analytics Journal*, *3*, 100071. https://doi.org/10.1016/J.DAJOUR.2022.100071

Devi, D., Biswas, S. K., & Purkayastha, B. (2020). A Review on Solution to Class Imbalance Problem: Undersampling Approaches. *2020 International Conference on Computational*

Performance Evaluation (ComPE), 626–631. https://doi.org/10.1109/ComPE49325.2020.9200087

Ghosh, K., Bellinger, C., Corizzo, R., Branco, P., Krawczyk, B., & Japkowicz, N. (2024). The class imbalance problem in deep learning. *Machine Learning, 113*(7), 4845–4901. https://doi.org/10.1007/S10994-022-06268-8/FIGURES/27

Grinsztajn, L., Oyallon, E., & Varoquaux, G. (2022). Why do tree-based models still outperform deep learning on typical tabular data?. *Advances in Neural Information Processing Systems, 35*, 507–520.

Haibo He, & Garcia, E. A. (2009). Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering, 21*(9), 1263–1284. https://doi.org/10.1109/TKDE.2008.239

Hairani, H., Anggrawan, A., & Priyanto, D. (2023). Improvement Performance of the Random Forest Method on Unbalanced Diabetes Data Classification Using Smote-Tomek Link. *JOIV : International Journal on Informatics Visualization, 7*(1), 258. https://doi.org/10.30630/joiv.7.1.1069

Halder, R. K., Uddin, M. N., Uddin, M. A., Aryal, S., & Khraisat, A. (2024). Enhancing K-nearest neighbor algorithm: a comprehensive review and performance analysis of modifications. *Journal of Big Data 2024 11*(1), 1–55. https://doi.org/10.1186/S40537-024-00973-Y

Hasib, K. Md., Iqbal, Md. S., Shah, F. M., Al Mahmud, J., Popel, M. H., Showrov, Md. I. H., Ahmed, S., & Rahman, O. (2020). A Survey of Methods for Managing the Classification and Solution of Data Imbalance Problem. *Journal of Computer Science, 16*(11), 1546–1557. https://doi.org/10.3844/jcssp.2020.1546.1557

Hosen, M. S., Islam, R., Naeem, Z., Folorunso, E. O., Chu, T. S., Mamun, M. A. Al, & Orunbon, N. O. (2024). Data-Driven Decision Making: Advanced Database Systems for Business Intelligence. *Nanotechnology Perceptions, 20*(S3), 687-704–687–704. https://doi.org/10.62441/NANO-NTP.V20IS3.51

Ionescu, S. A., & Diaconita, V. (2023). Transforming Financial Decision-Making: The Interplay of AI, Cloud Computing and Advanced Data Management Technologies. *International Journal Of Computers Communications & Control, 18*(6), 1–19. https://doi.org/10.15837/IJCCC.2023.6.5735

Ishaq, A., Sadiq, S., Umer, M., Ullah, S., Mirjalili, S., Rupapara, V., & Nappi, M. (2021). Improving the Prediction of Heart Failure Patients' Survival Using SMOTE and Effective Data Mining Techniques. *IEEE Access, 9*, 39707–39716. https://doi.org/10.1109/ACCESS.2021.3064084

Johnson, J. M., & Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. *Journal of Big Data, 6*(1), 1–54. https://doi.org/10.1186/S40537-019-0192-5/TABLES/18

Kaur, H., Pannu, H. S., & Malhi, A. K. (2019). A Systematic Review on Imbalanced Data Challenges in Machine Learning. *ACM Computing Surveys (CSUR), 52*(4). https://doi.org/10.1145/3343440

Khushi, M., Shaukat, K., Alam, T. M., Hameed, I. A., Uddin, S., Luo, S., Yang, X., & Reyes, M. C. (2021). A Comparative Performance Analysis of Data Resampling Methods on Imbalance Medical Data. *IEEE Access, 9*, 109960–109975. https://doi.org/10.1109/ACCESS.2021.3102399

Kovács, G. (2019). Smote-variants: A python implementation of 85 minority oversampling techniques. *Neurocomputing, 366*, 352–354. https://doi.org/10.1016/j.neucom.2019.06.100

Lemaître, G., Nogueira, F., & Aridas, C. K. (2016). Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *Journal of Machine Learning Research, 18*, 1–5. https://arxiv.org/abs/1609.06570v1

Liu, C., Wu, J., Mirador, L., Song, Y., & Hou, W. (2018). Classifying DNA methylation imbalance data in cancer risk prediction using SMOTE and Tomek link methods. *Communications in Computer and Information Science, 902*, 1–9. https://doi.org/10.1007/978-981-13-2206-8_1/COVER

López, V., Fernández, A., García, S., Palade, V., & Herrera, F. (2013). An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences*, *250*, 113–141. https://doi.org/10.1016/j.ins.2013.07.007

Madeeh, O. D., & Abdullah, H. S. (2021). An Efficient Prediction Model based on Machine Learning Techniques for Prediction of the Stock Market. *Journal of Physics: Conference Series*, *1804*(1). https://doi.org/10.1088/1742-6596/1804/1/012008

Mohammed, R., Rawashdeh, J., & Abdullah, M. (2020). Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results. *2020 11th International Conference on Information and Communication Systems, ICICS 2020*, 243–248. https://doi.org/10.1109/ICICS49469.2020.239556

Mqadi, N., Naicker, N., & Adeliyi, T. (2021). A SMOTe based Oversampling Data-Point Approach to Solving the Credit Card Data Imbalance Problem in Financial Fraud Detection. *International Journal of Computing and Digital Systems*, *10*(1), 277–286. https://doi.org/10.12785/ijcds/100128

Nekooeimehr, I., & Lai-Yuen, S. K. (2016). Adaptive semi-unsupervised weighted oversampling (A-SUWO) for imbalanced datasets. *Expert Systems with Applications*, *46*, 405–416. https://doi.org/10.1016/J.ESWA.2015.10.031

Piyadasa, T. D., & Gunawardana, K. (2023). A Review on Oversampling Techniques for Solving the Data Imbalance Problem in Classification. *International Journal on Advances in ICT for Emerging Regions (ICTer)*, *16*(1), 22–31. https://doi.org/10.4038/icter.v16i1.7260

Priyanka, & Kumar, D. (2020). Decision tree classifier: A detailed survey. *International Journal of Information and Decision Sciences*, *12*(3), 246–269. https://doi.org/10.1504/IJIDS.2020.108141

Rajaguru, H., & Sannasi Chakravarthy, S. R. (2019). Analysis of Decision Tree and K-Nearest Neighbor Algorithm in the Classification of Breast Cancer. *Asian Pacific Journal of Cancer Prevention : APJCP*, *20*(12), 3777. https://doi.org/10.31557/APJCP.2019.20.12.3777

Sawangarreerak, S., & Thanathamathee, P. (2020). Random forest with sampling techniques for handling imbalanced prediction of university student depression. *Information*, *11*(11), 519. https://doi.org/10.3390/INFO11110519

Shamsudin, H., Yusof, U. K., Jayalakshmi, A., & Akmal Khalid, M. N. (2020). Combining oversampling and undersampling techniques for imbalanced classification: A comparative study using credit card fraudulent transaction dataset. *IEEE International Conference on Control and Automation, ICCA*, 2020-October, 803–808. https://doi.org/10.1109/ICCA51439.2020.9264517

Speiser, J. L., Miller, M. E., Tooze, J., & Ip, E. (2019). A comparison of random forest variable selection methods for classification prediction modeling. *Expert Systems with Applications*, *134*, 93–101. https://doi.org/10.1016/J.ESWA.2019.05.028

Sun, L., Zhou, Y., Wang, Y., Zhu, C., & Zhang, W. (2020). The Effective Methods for Intrusion Detection With Limited Network Attack Data: Multi-Task Learning and Oversampling. *IEEE Access*, *8*, 185384–185398. https://doi.org/10.1109/ACCESS.2020.3029100

Swana, E. F., Doorsamy, W., & Bokoro, P. (2022). Tomek Link and SMOTE Approaches for Machine Fault Classification with an Imbalanced Dataset. *Sensors*, *22*(9), 3246. https://doi.org/10.3390/s22093246

Thabtah, F., Hammoud, S., Kamalov, F., & Gonsalves, A. (2020a). Data imbalance in classification: Experimental evaluation. *Information Sciences*, *513*, 429–441. https://doi.org/10.1016/J.INS.2019.11.004

Uddin, S., Haque, I., Lu, H., Moni, M. A., & Gide, E. (2022a). Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction. *Scientific Reports*, *12*(1), 1–11. https://doi.org/10.1038/s41598-022-10358-x

Wang, S., & Yao, X. (2012). Multiclass imbalance problems: Analysis and potential solutions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, *42*(4), 1119–1130. https://doi.org/10.1109/TSMCB.2012.2187280