LONTAR KOMPUTER VOL. 16, NO. 2 AUGUST 2025 **p-ISSN 2088-1541**
DOI : 10.24843/LKJTI.2025.v16.i2.p05 **e-ISSN 2541-5832**
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021

Evaluation of the performance of the Smote, Smote Enn, and Borderline Smote resampling methods based on the number of outlier data with Z Score

I Gede Aris Gunadi[1], Dewi Oktofa Rahmawati[2], Nurfa Risha[3], Dede Ardiayansah[4]
[1]Physics Study Program / Master's Program in Computer Science, Universitas Pendidikan Ganesha, Singaraja, Bali, Indonesia
[2],[3]Physics Study Program, Universitas Pendidikan Ganesha, Singaraja, Bali, Indonesia
[4]Master's Program in Computer Science, Universitas Pendidikan Ganesha, Singaraja, Bali, Indonesia
[1]igedearisgunadi@undiksha.ac.id

Abstract

Handling class imbalances in datasets poses a significant challenge in classification tasks, particularly when the minority class plays a crucial role in decision-making. One widely adopted solution is oversampling. This study compares the performance of three popular oversampling methods—SMOTE (Synthetic Minority Oversampling Technique), SMOTE-ENN (SMOTE with Edited Nearest Neighbor), and Borderline-SMOTE—based on the number of outliers generated. Outliers are identified using a Z-score-based statistical approach.

The research was conducted by applying the three oversampling methods across several datasets. Evaluation involved measuring the number of outliers after resampling, as well as assessing the impact of these methods on classification performance using accuracy, precision, recall, and F1-score as evaluation metrics. The results indicate that there is ***no substantial difference*** *in the number of outliers produced by SMOTE, SMOTE-ENN, or Borderline-SMOTE. For instance, in the Diabetes dataset, the percentage of outliers before and after resampling using SMOTE, SMOTE-ENN, and Borderline-SMOTE were 7.4%, 6.8%, 6.7%, and 6.3%, respectively. In the Predict Honor dataset, the values were 7.1%, 7.3%, 7.6%, and 7.0%, while in the Wine Quality dataset, they were 8.0%, 7.8%, 6.8%, and 5.8%. In the Smoking Status dataset, the percentages were 7.1%, 7.3%, 7.6%, and 7.0%.*

However, a more detailed examination of each feature in every dataset revealed that the behavior of the three algorithms varies, particularly regarding the number of outliers produced per feature. Despite this variation, the overall difference in total outliers remains insignificant across the methods. The second major finding concerns the performance of the decision tree classification model. It was observed that ***feature correlation has a greater impact on model performance*** *than achieving a perfectly balanced dataset. This suggests that focusing solely on class ratio without considering feature relationships may not lead to optimal results.*

Keywords: *Outlier data, Resample, smote, smote ENN , Borderline Smote*

1. Introduction

One of the key determinants of the performance quality of a machine learning model—particularly in supervised learning—is the balance of the dataset. Imbalanced data can negatively affect the learning process, especially when the model underrepresents the minority class. Data balance plays a vital role in influencing the predictive power of classification models. As stated in [1], achieving balanced data is essential for good classification performance, although perfect balance—defined as an equal number of samples for each class—is not strictly required. A class ratio of approximately 60:40 is still considered acceptable in practice.

However, in many real-world scenarios such as fraud detection, medical diagnosis, and pattern recognition, datasets are often highly imbalanced, with the number of minority class samples being significantly lower than those of the majority class. A common approach to address this issue is resampling. Resampling can be performed in two primary ways: (1) generating additional samples for the minority class to approximate the majority class in size, known as oversampling,

or (2) reducing the number of majority class samples to match the minority class, referred to as undersampling [2], [3].

The effectiveness of these resampling techniques largely depends on data distribution and characteristics—particularly the presence of outliers. Outliers, which are data points that significantly deviate from the general pattern of the dataset, can reduce the representativeness of training data and degrade classification performance. Outliers are typically defined as instances with values far above the upper quartile (Q3) or far below the lower quartile (Q1) of the data distribution [4], [5].

Outliers have a substantial impact on classification model performance, as shown in the study by [6]. That study implemented three algorithms—Linear Discriminant Analysis (LDA), K-Nearest Neighbors (KNN), and Recursive Partitioning and Regression Trees (Rpart)—under three different conditions: (1) using the original dataset, (2) removing outlier data, and (3) randomly deleting instances from the dataset. The Iris dataset was used for experimentation. The results revealed that the lowest error rates were consistently achieved under condition (2), where outlier data was removed. Specifically, for the LDA model, the error rates were 2.02% (original), 1.54% (outlier removal), and 2.30% (random deletion). For KNN, the error rates were 4.05%, 2.30%, and 4.10%, respectively. In the Rpart model, the error rates were 6.69%, 2.90%, and 7.32%, respectively.

In general, outliers can affect machine learning performance in three major ways: (1) Distorting descriptive statistics such as the mean, making them less representative of the dataset; (2) Disrupting model learning, particularly in models sensitive to extreme values such as linear regression; (3) Increasing the risk of overfitting, as some algorithms may become overly influenced by outlier points and fail to generalize to new data [7], [8].

This study focuses on evaluating how SMOTE-based resampling methods behave in the presence of outlier data a condition that has rarely been addressed in prior research. While SMOTE is widely used to balance datasets, its behavior under varying levels of outlier density remains underexplored. Using Z-score-based outlier detection, we measure the extent to which SMOTE, SMOTE-ENN, and Borderline-SMOTE generate or amplify outliers, a comparison that has not been systematically investigated. Although the datasets used in this study are publicly available, they feature varying degrees of imbalance and natural outlier distributions, making them appropriate for controlled experimentation. Unlike most studies that apply SMOTE solely to improve classification accuracy, this research investigates the structural impact of oversampling on data quality, with a specific focus on the emergence of new outliers. To the best of our knowledge, this is the first comparative analysis that evaluates outlier generation across SMOTE variants using Z-score-based metrics.

Table 1 . Description of Dataset in This Research

No	Dataset name	Description
1	Smoking Dataset	This dataset consists of 26 features, including 25 predictors and 1 target variable. It contains a total of 55,692 instances, with 35,327 instances belonging to the **non-smoking** class and 20,455 instances to the **smoking** class.
2	Predict Honor Dataset	This dataset consists of 42 features, 41 of which are predictor variables. The total number of instances is 21,148. The target variable has two classes: **0** and **1**, with 14,529 and 4,843 instances, respectively.
3	Wine Quality Dataset	This dataset consists of 11 features and a total of 1,599 instances. The target variable contains six classes with the following distribution: Class **5**: 681 instances; Class **6**: 683 instances; Class **7**: 199 instances; Class **4**: 53 instances; Class **8**: 18 instances; Class **3**: 10 instances
4	Diabetes Dataset	This dataset contains 8 features and a total of 768 instances. The target variable consists of two classes: **class 0** with 500 instances and **class 1** with 268 instances.

2. Research Methods

This study was conducted to evaluate the performance and characteristics of the SMOTE, SMOTE-ENN, and Borderline-SMOTE resampling algorithms across various datasets with initially imbalanced class distributions. The comparison focuses on two main aspects: the number of outliers generated by each algorithm and the classification performance achieved after applying

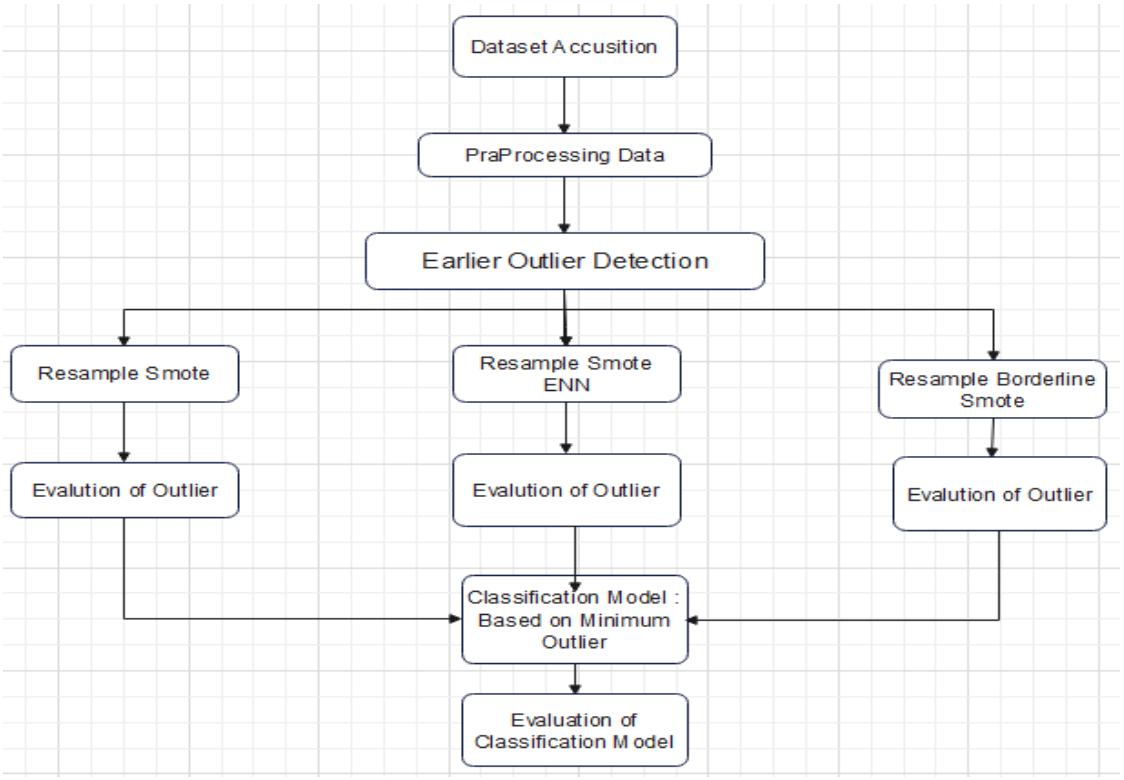


Figure 1. Research Process

oversampling. Four different datasets were used in this research, all obtained from the Kaggle platform. A description of each dataset is provided in Table 1 and the overall research workflow is illustrated in Figure 1.

The datasets used in this study exhibit varying degrees of class imbalance. The Wine Quality and Predict Honor datasets are characterized by severe class imbalance, while the Smoking and Diabetes datasets display moderate imbalance ratios of approximately 63:37 and 65:35, respectively. Although these moderate imbalances are not extreme, they are still known to adversely affect classifier performance particularly in cases where the minority class plays a critical role in decision-making. The inclusion of both moderately and severely imbalanced datasets in this study is intentional. This design enables a more comprehensive evaluation of SMOTE and its variants across a range of real-world conditions. Specifically, we aim to assess whether SMOTE-based resampling remains effective in scenarios with moderate imbalance, as such distributions are commonly encountered in practical applications, including industrial and healthcare domains.

2.1 Exploratory Data Analysis (EDA)

This section presents an analysis of the dataset's initial condition, commonly referred to as **Exploratory Data Analysis (EDA)**. The primary focus is to examine key aspects of the data, including missing values, redundancy, and class distribution. As noted in prior studies [9], [10], EDA is essential for identifying common data issues such as missing values, duplicated entries, and outliers. Additionally, EDA provides insights that support feature selection and model development. In this study, two main EDA tasks are conducted, (1) Cleaning the dataset by addressing missing values and redundant data. (2) Feature selection based on correlation analysis. For each dataset, a correlation threshold is defined to determine which features exhibit a moderate level of correlation. This threshold is used to select relevant features for model training. Previous studies, such as [11], have also applied correlation-based feature selection, sometimes using ranking methods to identify the top-*n* most correlated features. However, there

is no universally accepted rule for selecting the optimal *n*, and this study similarly does not rely on a fixed reference for determining the best ranking threshold.

2.2 Evaluation Outlier Pada Dataset

To identify the presence of outliers in the feature and target variables, a **boxplot visualization** was used as an initial diagnostic tool. Figure 2 illustrates the distribution of the data and highlights potential outliers, as described in [12].

In addition to visualization, a second method employed in this study is the **Z-score technique**, which quantitatively measures the presence of outliers. As supported by prior studies [13], [14], the Z-score method is recognized as an effective approach for outlier detection. The Z-score formula used in this analysis is presented in Equation 1.

$$Z\ score = \frac{(X-\mu)}{\sigma} \quad (1)$$

Z = Z-score of the data point
μ = mean of the dataset
σ = standard deviation of the dataset
x = value of the data point being evaluated
In this study, a data point is considered an outlier if its absolute Z-score (**|Z|**) is greater than 3.

2.3 Decision Tree Model

In this study, a single machine learning model was selected: the decision tree classification algorithm based on the Gini index. The Gini index is a commonly used criterion in decision tree construction, where it measures the impurity of a node to determine the best split at each branch. The principles of this classification method are discussed in [15], [16].
Unlike the C4.5 algorithm, which relies on entropy and gain ratio for tree construction, the Gini-based decision tree uses Gini impurity as its splitting criterion. The process of calculating entropy and gain ratio is detailed in [17]. The Gini index formula is presented in Equation 2.

$$Gini = 1 - \Sigma(P_i^2) \quad (2)$$

$\boldsymbol{P}_i$: Proportion of samples in the ith class at a node
$\Sigma(P_i^2)$: The sum of the squares of the class probabilities at that node.

The interpretation of the Gini value is as follows: if all samples within a node belong to a single class (the node is completely homogeneous), the Gini index equals **0**, indicating no impurity. Conversely, if the samples are evenly distributed across all classes (maximum impurity), the Gini index approaches **1**. A detailed explanation of the classification process using the Gini index in decision trees can be found in [15].

2.4. Classifier Performance Evaluation

Classifier performance is commonly evaluated using a confusion matrix approach. This matrix summarizes the classification results into four categories: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). These classification outcomes are illustrated in Figure 2.

	Positive Prediction	Nagative Prediction
Positive Class	True Positive (TP)	False Negative (FN)
Nagative Class	False Positive (FP)	True Negative (TN)

Figure 2. Confusion Matrix

True Positive (TP): The number of positive data correctly predicted as positive
False Positive (FP): The number of negative data incorrectly predicted as positive.
False Negative (FN): The number of positive data incorrectly predicted as negative.
True Negative (TN): The number of negative data correctly predicted as negative.

In the research, five parameters were used, namely:

a. Accuracy: Measures the percentage of correct predictions (both positive and negative). Expressed by Equation 3.

$$Accuracy = \frac{TP+TN}{(TP+TN+FP+FN)} \quad (3)$$

b. Precision: Measures how reliably the model predicts the positive class (how many positive predictions are correct). Expressed by Equation 4.

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

c. Recall (Sensitivity): Also known as sensitivity or True Positive Rate (TPR), it measures how well the model captures positive data. Expressed by Equation 5.

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

d. F1 score :F1-Score is the harmonic average of Precision and Recall, used when it is important to balance the two. Expressed by Equation 6.

$$F1\ Score = \frac{2\ x\ (Precession\ x\ Recall)}{Precession+Recall} \quad (6)$$

3. Result and Discussion

Based on the research workflow outlined in Section 2, the following presents the results and analysis of each stage of the process.

3.1 Results of the Exploratory Data Analysis (EDA)

a. Smoking Dataset.

The detailed results of the Exploratory Data Analysis (EDA) for the Smoking Dataset are presented in Table 2.

Table 2. EDA Dataset Smoking Dataset

Parameter	Description Condition
Empty Data	Not Found
Duplicated Data	Not Found
Balanced Fitur	Class **0** consists of 35,327 instances, while class **1** consists of 20,455 instances, resulting in a class distribution ratio of approximately **63% to 37%**.
Fitur Selection	Based on the correlation analysis with a threshold of 0.3, the selected predictor features are: *gender*, *height*, *weight*, and *hemoglobin*.

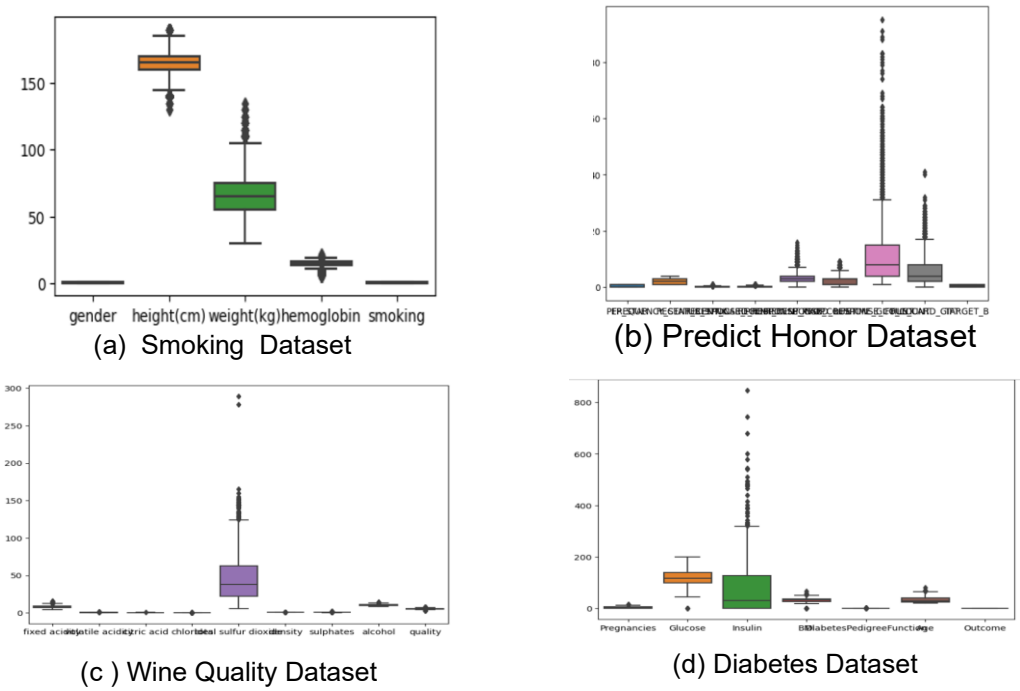
From the EDA on the Smoking Dataset, we observe that there are no missing or duplicated instances, which suggests a relatively clean dataset. However, the class distribution shows a moderate imbalance (63% vs 37%). This justifies the use of resampling methods such as SMOTE to balance the classes before classification. Feature selection based on correlation identified ‘gender’, ‘height’, ‘weight’, and ‘hemoglobin’ as the most informative predictors, which aligns with known medical indicators related to smoking behavior. These features are retained for further analysis

b. Predict Honor Dataset

The detailed results of the Exploratory Data Analysis (EDA) for the Predict Honor Dataset are presented in Table 3.

Table 3. EDA Predict Honor Dataset

Parameter	Description Condition
Empty Data	There were four features with a high proportion of missing data: **‘Target_D’**, with 14,529 missing values , **‘Honor_Age’**, with 4,795 missing values , **‘Income_Group’**, with a significant amount of missing data (exact count not specified) , **‘Wealth_Rating’**, with 8,810 missing values Due to the large amount of missing data, these features were removed from the dataset.
Duplicated Data	Not Found
Balanced Fitur	The target feature consists of two classes: **class 0** with 14,529 instances and **class 1** with 4,843 instances.



(a) Smoking Dataset

(b) Predict Honor Dataset

(c) Wine Quality Dataset

(d) Diabetes Dataset

Figure 3. Visualisation quantity of Outlier Data

Fitur Selection	A total of eight predictor features were selected based on a correlation threshold of 0.1.

The Predict Honor Dataset initially contained substantial missing values in four features, which were removed due to their dominance. After cleaning, a class imbalance remains (approximately 75% vs 25%), indicating a need for oversampling. Eight features with correlation threshold > 0.1 were selected, which are expected to enhance model generalizability while reducing noise

c. Dataset Wine Quality Dataset

The detailed results of the Exploratory Data Analysis (EDA) for the Wine Quality dataset are presented in Table 4.

Table 4. EDA Wine Quality Dataset

Parameter	Description Condition
Empty Data	Not Found
duplicated Data	A total of **240 duplicate instances** were identified and subsequently removed from the dataset.
Balanced Fitur	The target variable consists of the following class distributions: **Class 5**: 577 instances; **Class 6**: 535 instances; **Class 7**: 167 instances; **Class 4**: 53 instances ; **Class 8**: 17 instances ; **Class 3**: 10 instances
Fitur Selection	Based on a correlation threshold of 0.1, a total of eight predictor features were selected.

EDA on the winequality dataset reveals a severe imbalance across multiple classes, especially in the minority classes (class 3 with only 10 instances). This extreme imbalance, combined with duplicated instances (240 rows), makes the dataset highly sensitive to synthetic sampling methods. Feature selection yielded 8 predictors, though visual analysis suggests limited class separability, which may explain lower classification performance later.

d. Diabetes Dataset

The detailed results of the Exploratory Data Analysis (EDA) for the **Diabetes Dataset** are presented in Table 5. The diabetes dataset is relatively balanced (500 vs 268 instances), and free

of missing or duplicated data. Correlation-based feature selection resulted in 6 features used in the classification model. This dataset serves as a suitable test case to evaluate the effect of SMOTE methods under moderately imbalanced but clean conditions.

Table 5. EDA Diabetes Dataset

Parameter	Description Condition
Empty Data	Not Found
Duplicated Data	Not Found
Balanced Fitur	class (0) consists of 500 lines, class (1) consists of 268 lines.
Fitur Selection	The Predictor features selected with a threshold of 0.3 are, 6 features, aimed at Fig 6.

3.2 Analysis of Outlier Data

3.2.1 Initial Condition of Dataset

Before applying resampling techniques, outlier identification was performed using both data visualization and quantitative analysis for each dataset. Figure 3 illustrates the distribution of outliers detected in each dataset.

Figure 3 illustrates the presence of outliers in the raw datasets using boxplot visualizations. Notably, features such as 'weight' and 'hemoglobin' in the Smoking Status dataset show extreme values, indicating potential skewness in data distribution. This observation supports the need for careful handling prior to classification. Figure 4 below shows the quantity of outliers for each dataset

Smoking.csv	
Fitur	**Outlier Data**
Gender	0
Height	7
Weight	398
Hemoglobin	525
Smoking	0

(a) . Outlier in Smoking Dataset

diabetes.csv	
Fitur	**Outlier Data**
Pregnancies	4
Glucose	5
Insulin	18
BMI	14
DiabetesPedigree	11
Age	5
Outcome	0

(b) . Outlier in Diabetes Dataset

Predict_honor.csv	
Fitur	**Outlier Data**
Pep Star	0
FrequencyStatus	0
RecentResponseProp	236
RecentCardResponse	121
RecentResponseCount	231
RecentCardResponseCount	239
LifetimeGiftCount	298
FileCardGift	251
Target_B	0

(c) . Outlier in Predict Honor Dataset

Wine_Quality.csv	
Fitur	**Outlier Data**
fixed acidity	9
volatile acidity	9
Citric_acid	1
chlorides	27
Total sulfur	12
density	13
sulphates	21
alcohol	7
quality	10

(d). Outlier in Wine_Quality Dataset

Figure 4 . Quantity outlier Data on each Dataset's Feature Before Resampling

Next, the resampling process was performed using three algorithms: SMOTE, SMOTE-ENN, and Borderline-SMOTE. The following presents the outlier conditions observed in each dataset after the resampling process.

3.2.2 Conditions After Resampling Smoking Dataset

In the Smoking Dataset, three resampling algorithms were implemented: SMOTE, SMOTE-ENN, and Borderline-SMOTE. For SMOTE and Borderline-SMOTE, a resampling ratio of 1 was applied to balance the dataset, meaning the number of instances in each class was equalized based on the majority class. In contrast, SMOTE-ENN was applied using the 'auto' setting. Unlike the other two methods, SMOTE-ENN typically results in a reduction in the total number of instances, as the ENN (Edited Nearest Neighbor) component is designed to remove samples that are considered ambiguous or noisy near class boundaries. The results of outlier detection in the Smoking Dataset after applying each resampling method are presented below. Outliers are analyzed using boxplot visualizations and Z-score analysis, with a Z-score threshold of 3 used to identify outlier instances.

Table 6. Outlier data on Smoking Dataset

	Initial Condition	Smote	Smote ENN	Borderline Smote
Gender	0	0	0	0
Height	7	7	17	7
Weight	398	287	236	541
Hemoglobin	525	671	203	679
Smoking	0	0	0	0
Total Data	55692	70474	26746	70474

Furthermore, the percentage of outlier data per feature was calculated based on the total number of instances resulting from each resampling algorithm. These percentages are illustrated in Figure 5. Subsequently, the resampling process was applied to the Predict Honor dataset. The quantity of outlier data generated by the three resampling algorithms was compared. The results obtained after applying SMOTE, SMOTE-ENN, and Borderline-SMOTE to the Predict Honor dataset are presented in Table 7.

Table 7. Outlier data in Predict Honor Dataset

Fitur	Initial Condition	Smote	Smote ENN	Borderline Smote
Pep_Star	0	0	0	0
Frequency_Status_97nk	0	0	0	0
Recent_Response_Prop	236	344	137	341
Recent_Card_Response_Prop	121	174	69	164
Recent_Response_Count	231	354	181	327
Recent_Card_Response_Count	239	368	174	343
Lifetime_Gift_Count	298	466	208	441
File_Card_Gift	251	414	144	406
Target_B	0	0	0	0
Total Outlier Data	1376	2120	913	2022
Total data	19372	29058	12085	29058

The same resampling treatment was applied to the Wine Quality dataset, and the resulting data are presented in Table 8.

Table 8. Outlier data in Wine Quality Dataset

Fitur	Initial Condition	Smote	Smote ENN	Borderline Smote
fixed acidity	9	17	3	17
volatile acidity	9	48	12	34
citric acid	1	5	5	3
chlorides	27	64	25	62
total sulfur dioxide	12	46	46	43
density	13	23	4	5
sulphates	21	40	18	37
alcohol	7	1	0	1
quality	10	0	0	0
Total Outlier	109	244	113	202
Total data	1359	3112	1666	3462

The same resampling treatment was applied to the **Diabetes dataset**, and the resulting data are presented in Table 9.

Table 9. Outlier data on Diabetes Dataset

Fitur	Initial Condition	Smote	Smote ENN	Borderline Smote
Pregnancies	4	4	1	4
Glucose	5	5	0	7
Insulin	18	25	13	18
BMI	14	16	8	17
DiabetesPedigreeFunction	11	13	8	12
Age	5	5	5	5
Outcome	0	0	0	0
Total Outlier	57	68	35	63
Total Data	768	1000	522	1000

3.3 Model Classification

The dataset was initially imbalanced, as described in Table 1. To address this, three data balancing algorithms were applied. The analysis compared the number of outliers before and after the resampling process. Among the three methods, SMOTE-ENN produced the most favorable results in terms of minimizing outlier data. Following this, the dataset resampled using SMOTE-ENN was used to train a Decision Tree classifier based on the Gini index. The training process used 80% of the data for model training, with the maximum tree depth set to 4.

The classification performance on the resampled Diabetes Dataset was strong, with macro-average scores for precision, recall, F1-score, and accuracy reaching 92%, 91%, 91%, and 91%, respectively. Similarly, on the Smoking Dataset, these metrics reached 91%, 94%, 92%, and 93%, respectively. In contrast, the Wine Quality Dataset yielded the lowest performance, with precision, recall, F1-score, and accuracy of 65%, 57%, 58%, and 72%, respectively.

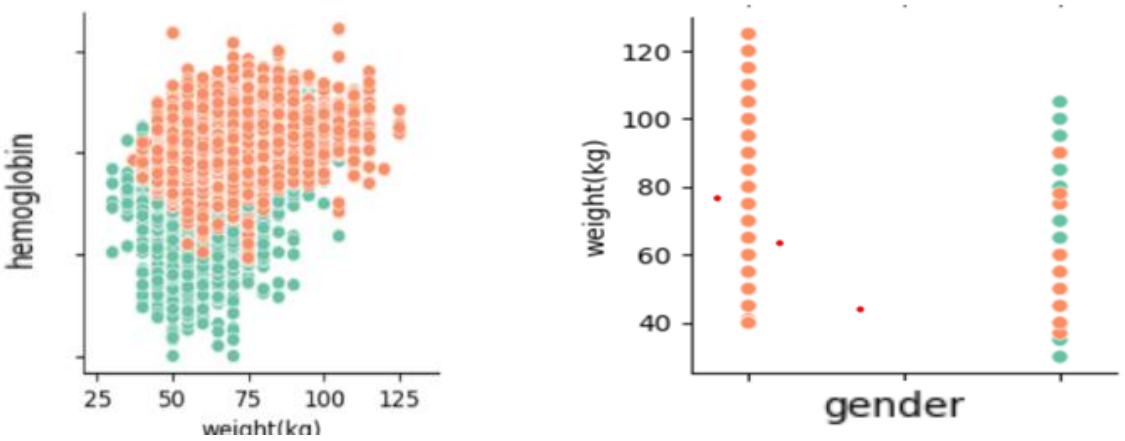
Table 10. Performance of the Decision Tree Gini Index in 4 Datasets Resample Test

Dataset	Accuracy	Precesion	Recall	F1-Score
Smoking Dataset	0.93	0.91	0.94	0.92
Predict Honor Dataset	0.74	0.74	0.74	0.74
Wine Quality Dataset	0.72	0.65	0.57	0.58
Diabetes Dataset	0.91	0.92	0.91	0.91

In the Smoking Dataset and Diabetes Datasets, a correlation threshold of 0.3 was used for feature selection, while in the Wine Quality Dataset and Predict Honor Datasets, a lower threshold of 0.1 was applied. This approach is supported by the findings in [1], which suggest that the impact of effective feature selection is more significant than achieving perfect class balance.

To evaluate the effect of tree complexity on classification performance, additional experiments were conducted using the Predict Honor Dataset, with decision trees of varying maximum depths: 4, 7, 9, and 11. The results indicate that increasing tree depth improves accuracy up to a certain point. Beyond that, performance gains become marginal or may even decline due to overfitting. At a maximum depth of 4, the model achieved solid performance with an accuracy of approximately 74% and an F1-score of 74%. Increasing the depth to 7 and 9 resulted in slight improvements, suggesting better capture of complex decision boundaries.

The correlation among features plays a crucial role in determining the classification model's performance. Tools such as pair plot analysis can aid in detecting feature correlations, identifying outliers, visualizing data distributions, and guiding feature selection. Based on the pair plot visualization, several feature pairs show clear class separation. For example, the combination of hemoglobin and weight demonstrates a relatively distinct separation between classes.



(a). Correlation Hemoglobin with Weight (b).Correlation weight with gender

Figure 4 . Correlation Feature based on Pairplot

The visualization in Figure 4(a) illustrates the correlation between the hemoglobin and weight features for class separation between class 0 (non-smoking) and class 1 (smoking). Class 0 is represented in red, while class 1 is shown in blue. The separation between the two classes is relatively clear, indicating that this feature pair contributes meaningfully to classification.

In contrast, Figure 4 (b) presents the relationship between the weight and gender features, which shows a weaker class separation. For individuals with gender value 1, both low and high weights still result in overlapping instances between the two classes, making it more challenging to distinguish between smokers and non-smokers. Nonetheless, some degree of separation is still observable in the plot. Across all datasets used in this study, pairplot visualizations were employed to evaluate the quality of feature correlations and their ability to separate classes. Based on the analysis of these visualizations, it can be concluded that the selected features in each dataset exhibit sufficient class separation, supporting their suitability for the classification process. However, the strength of feature correlations and separability naturally varies among datasets. This difference in feature separability is reflected in the classification performance. For instance, the Wine Quality dataset showed lower performance compared to the Smoking dataset. As seen in the pairplot visualizations, the Wine Quality dataset exhibits more overlap between classes, making it more difficult for the model to distinguish between them. Consequently, this results in lower classification metrics.

3.4 Discussion

Several key points emerged in the initial discussion regarding the common assumption that SMOTE-ENN or Borderline-SMOTE would outperform standard SMOTE. However, this assumption does not always hold true across all datasets. For example, Figure 5 below illustrates the results from the Smoking Dataset, which highlights some unexpected outcomes in terms of outlier generation and classification performance.

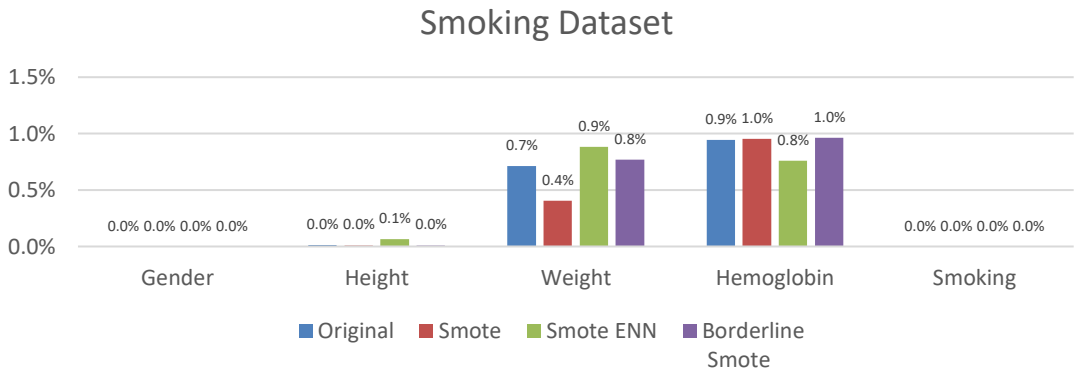


Figure 5. Percentage of Outlier Data in Each Feature on Smoking Status Dataset

However, this was not confirmed in the present study. In several features, the number of outliers generated by SMOTE-ENN and Borderline-SMOTE was actually higher than that produced by SMOTE. There are several possible explanations for why SMOTE-ENN may result in more outliers compared to standard SMOTE. As is known, SMOTE-ENN combines two techniques: SMOTE, an oversampling method that synthesizes new instances for the minority class, and Edited Nearest Neighbors (ENN), which removes samples deemed "inconsistent" based on the majority of their nearest neighbors. Importantly, ENN can remove samples from both majority and minority classes. Several mechanisms may contribute to the increased number of outliers in SMOTE-ENN: (1) Excessive removal of data points in certain regions may leave synthetic samples generated by SMOTE without sufficient support from neighboring instances of the same class, causing them to be interpreted as outliers. (2) Aggressive deletion near class boundaries may eliminate valid data points, resulting in an unnatural distribution that increases the likelihood of outlier formation. Based on the findings of this study, it can be argued that the differences in outlier generation between SMOTE and SMOTE-ENN are not consistently large or significant. The extent of difference largely depends on the nature of the dataset. In datasets with highly overlapping or complex class boundaries, as illustrated in Figure 10(a), the risk of outlier formation increases, even with more sophisticated resampling techniques. However, in datasets with clearer class separation such as in Figure 10(b) SMOTE-based algorithms are more likely to generate high-quality synthetic samples, thus improving model performance.

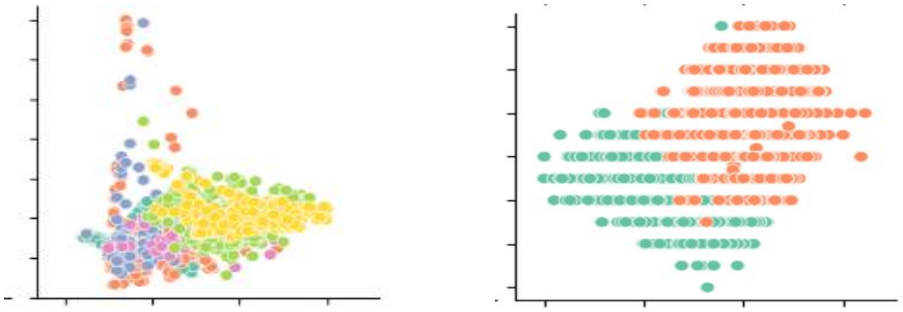


Figure 6 (a) Complex Overlapping data distribution (b) Separated well data distribution

This study also found that Borderline-SMOTE had a less significant impact compared to standard SMOTE. As is known, Borderline-SMOTE is an oversampling technique that focuses on generating synthetic samples near the decision boundary, where the minority class is most at risk of misclassification. However, when the data distribution exhibits significant class overlap, the algorithm may produce inaccurate or misleading synthetic samples, ultimately reducing the effectiveness of the resampling process. In such scenarios, the boundary between classes becomes ambiguous, making it difficult to generate meaningful data without introducing noise or outliers. Based on the findings of this study, it is recommended that a thorough preliminary

analysis of the data distribution be conducted prior to applying Borderline-SMOTE. Understanding the degree of class overlap and the structure of the feature space can help determine whether this resampling method is appropriate or if alternative techniques may yield better performance.

4. Conclusions

Based on the experimental results, several important conclusions can be drawn. First, the study shows that feature selection plays a more dominant role in improving classification performance compared to data balancing through resampling. In this context, it was also found that achieving a perfectly balanced class distribution (ratio 1:1) is not necessarily optimal for model performance. Second, although theoretically SMOTE-ENN and Borderline-SMOTE are expected to reduce the number of outliers due to their refined sampling mechanisms, the findings did not fully support this assumption. In several datasets, the standard SMOTE algorithm actually produced fewer outliers than its more complex variants, indicating that these advanced methods may behave unpredictably depending on the data structure.

Smote ENN can produce more outliers than smote, which is caused by the combination of oversampling and aggressive instance deletion. This process can disrupt the original data distribution, particularly around class boundaries. Such behavior is a side effect of highly aggressive balancing strategies and may be mitigated through careful parameter tuning, data preprocessing, and distribution analysis prior to resampling.

Finally, when applying SMOTE-ENN, caution is advised, especially in datasets where the majority and minority classes overlap significantly. In such cases, SMOTE may generate synthetic samples in overlapping regions, and the ENN step may remove nearby majority instances, unintentionally isolating the synthetic points. This effect causes the synthetic data to behave like outliers, which may harm the model's generalization. Therefore, understanding the interaction between class distribution and the chosen resampling strategy is critical for successful implementation.

References

[1] I. G. A. Gunadi and D. O. Rachmawati, "A Comparative Study on the Impact of Feature Selection and Dataset Resampling on the Performance of the K-Nearest Neighbors (KNN) Classification Algorithm," *J. Nas. Pendidik. Tek. Inform.*, vol. 13, no. 2, pp. 419–427, 2024, doi: 10.23887/janapati.v13i2.82174.

[2] L. Qadrini, H. Hikmah, and M. Megasari, "Oversampling, Undersampling, Smote SVM dan Random Forest pada Klasifikasi Penerima Bidikmisi Sejawa Timur Tahun 2017," *J. Comput. Syst. Informatics*, vol. 3, no. 4, pp. 386–391, 2022, doi: 10.47065/josyc.v3i4.2154.

[3] R. Mohammed, J. Rawashdeh, and M. Abdullah, "Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results," *2020 11th Int. Conf. Inf. Commun. Syst. ICICS 2020*, no. May, pp. 243–248, 2020, doi: 10.1109/ICICS49469.2020.239556.

[4] D. Cousineau and S. Chartier, "Outliers detection and treatment: a review.," *Int. J. Psychol. Res.*, vol. 3, no. 1, pp. 58–67, 2010, doi: 10.21500/20112084.844.

[5] I. W. Dharmana, I. G. A. Gunadi, and L. J. E. Dewi, "Deteksi Transaksi *Fraud* Kartu Kredit Menggunankan *Oversampling* ADASYN dan Seleksi Fitur SVM-RFECV," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 11, no. 1, pp. 125–134, 2024, doi: 10.25126/jtiik.20241117640.

[6] E. Acuña and C. Rodriguez, "On detection of outliers and their effect in supervised classification," *Dep. Math. Puerto Rico,Mayaguez*, no. June, 2005.

[7] P. R. Sihombing, S. Suryadiningrat, D. A. Sunarjo, and Y. P. A. C. Yuda, "Identifikasi Data Outlier (Pencilan) dan Kenormalan Data Pada Data Univariat serta Alternatif Penyelesaiannya," *J. Ekon. Dan Stat. Indones.*, vol. 2, no. 3, pp. 307–316, 2023, doi: 10.11594/jesi.02.03.07.

[8] S. Paludi, "Identifikasi dan Pengaruh Keberadaan Data Pencilan (Outlier) (Studi Kasus Jumlah Kunjungan Wisman dan Pengunjung Asing ke Indonesia Melalui Pintu Masuk Makasar Antara Bulan Januari 2007 s.d. Juli 2008)," *Panor. Nusant.*, no. January 2009, pp. 56–62, 2009, [Online]. Available: https://stein.ac.id/e-journal/pn_6/PN_6.pdf

[9] A. Wibowo, "Analisa Dan Visualisasi Data Penjualan Menggunakan Exploratory Data Analysis Pada PT. Telkominfra," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9,

no. 3, pp. 2292–2304, 2022, doi: 10.35957/jatisi.v9i3.2737.

[10] M. Radhi, A. Amalia, D. R. H. Sitompul, S. H. Sinurat, and E. Indra, "Analisis Big Data Dengan Metode Exploratory Data Analysis (Eda) Dan Metode Visualisasi Menggunakan Jupyter Notebook," *J. Sist. Inf. dan Ilmu Komput. Prima(JUSIKOM PRIMA)*, vol. 4, no. 2, pp. 23–27, 2022, doi: 10.34012/jurnalsisteminformasidanilmukomputer.v4i2.2475.

[11] N. T. Romadloni and Hilman F Pardede, "Seleksi Fitur Berbasis Pearson Correlation Untuk Optimasi Opinion Mining Review Pelanggan," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 3, no. 3, pp. 505–510, 2019, doi: 10.29207/resti.v3i3.1189.

[12] M. Y. Darsyah, "Penggunaan Stem and Leaf dan Boxplot untuk Analisis Data," *J. Pendidik. Mat.*, vol. 1, no. 1, pp. 55–67, 2014, [Online]. Available: http://103.97.100.145/index.php/JPMat/article/view/1045/1093

[13] P. V. Anusha, C. Anuradha, P. S. R. Chandra Murty, and C. S. Kiran, "Detecting outliers in high dimensional data sets using Z-score methodology," *Int. J. Innov. Technol. Explor. Eng.*, vol. 9, no. 1, pp. 48–53, 2019, doi: 10.35940/ijitee.A3910.119119.

[14] K. Malik, H. Sadawarti, and K. G. S, "Comparative Analysis of Outlier Detection Techniques," *Int. J. Comput. Appl.*, vol. 97, no. 8, pp. 12–21, 2014, doi: 10.5120/17026-7318.

[15] T. Daniya, M. Geetha, and K. S. Kumar, "Classification and regression trees with gini index," *Adv. Math. Sci. J.*, vol. 9, no. 10, pp. 8237–8247, 2020, doi: 10.37418/amsj.9.10.53.

[16] S. Tangirala, "Evaluating the impact of GINI index and information gain on classification using decision tree classifier algorithm," *Int. J. Adv. Comput. Sci. Appl.*, no. 2, pp. 612–619, 2020, doi: 10.14569/ijacsa.2020.0110277.

[17] I. M. Arya, A. Dwija, I. M. Gede, and I. G. Aris, "JTIM : Jurnal Teknologi Informasi dan Multimedia https://journal.sekawan-org.id/index.php/jtim/ Perbandingan Algoritma Naive Bayes Berbasis Feature Selection Gain Ratio dengan Naive Bayes Kovensional dalam Prediksi Komplikasi Hipertensi I Made Arya Adinat," vol. 6, no. 1, pp. 37–49, 2024, [Online]. Available: https://doi.org/10.35746/jtim.v6i1.488