Matrik: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer
Vol. 22, No. 2, Maret 2023, pp. 227∼238
ISSN: 2476-9843, accredited by Kemenristekdikti, Decree No: 200/M/KPT/2020
DOI: 10.30812/matrik.v22i2.2515

The Effect of Class Imbalance Handling on Datasets Toward Classification Algorithm Performance

Cherfly Kaope , Yoga Pristyanto
Universitas Amikom Yogyakarta, Yogyakarta, Indonesia

Article Info

Article history:

Received November 16, 2022
Revised January 13, 2023
Accepted February 17, 2023

Keywords:

Classification
Imbalanced Class
Machine Learning
Oversampling
Undersampling

ABSTRACT

Class imbalance is a condition where the amount of data in the minority class is smaller than that of the majority class. The impact of the class imbalance in the dataset is the occurrence of minority class misclassification, which can affect classification performance. Various approaches have been taken to deal with the problem of class imbalances, such as the data level approach, algorithmic level approach, and cost-sensitive learning. At the data level, one of the methods used is to apply the sampling method. In this study, the ADASYN, SMOTE, and SMOTE-ENN sampling methods were used to deal with the problem of class imbalance combined with the AdaBoost, K-Nearest Neighbor, and Random Forest classification algorithms. This study aimed to determine the effect of handling class imbalances on the dataset on classification performance. The tests were carried out on five datasets, and based on the classification results, the integration of the ADASYN and Random Forest methods gave better results than other model schemes. The evaluation criteria include accuracy, precision, true positive rate, true negative rate, and g-mean score. The results of the classification of the integration of the ADASYN and Random Forest methods gave 5% to 10% better than other models.

Corresponding Author:

Yoga Pristyanto, +6285156727484,
Faculty of Computer Science,
Universitas Amikom Yogyakarta, Yogyakarta, Indonesia,
Email: yoga.pristyanto@amikom.ac.id

How to Cite:C. Kaope and Y. Pristyanto, ”The Effect of Class Imbalance Handling on Datasets Toward Classification Algorithm Performance”, *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 22, no. 2, pp. 227-238, Mar. 2023.

1. INTRODUCTION

A problem often found in the classification is the imbalance of classes. Class imbalance occurs when the data is not evenly distributed, and the number of minority classes is smaller than that of majority classes [1]. This condition can lead to the classifier mistakenly classifying the minority class and the classifier tending to choose the majority class and ignore the minority class. It can affect the performance of the classification. There are several ways to deal with the problem of class imbalance: the data-level approach, algorithmic level approach, and cost-sensitive learning [2]. One way to deal with class imbalances at the data level is to apply sampling methods [3]. The sampling method is an approach to balance the distribution of minority classes and majority classes. The sampling method is divided into three types: undersampling, oversampling, and a combination of oversampling and undersampling (hybrid sampling) methods. Undersampling removes objects in the majority class randomly with the goal that the number of objects each class has is the same. Oversampling randomly selects objects from minority classes, thus generating new objects. Hybrid sampling is a combination of oversampling and undersampling. This sampling method adds new objects to the minority class and subtracts objects from the majority class to balance the data [4].

Research on handling class imbalances with sampling methods has been widely conducted, resulting in good classification performance. For example, the study conducted by [5] using the ADASYN oversampling method to balance classes on the hypertension dataset shows that the method can help classification models classify hypertension classes and significantly improve classification performance in each classification model compared to without applying oversampling methods. The study by [6] used ADASYN and SMOTE methods to address class imbalances in diabetes mellitus data and was classified using the Support Vector Machine (SVM) algorithm. The study showed increased classification performance after applying the oversampling method, with an accuracy value of 87.3% for the ADASYN + SVM method and 85.4% for the SMOTE + SVM method. In contrast, the accuracy result without oversampling was lower, which was 83%. Another study combined Synthetic Minority Oversampling Technique (SMOTE) oversampling and Edited Nearest Neighbor (ENN) undersampling methods to balance data on Land Use and Land Cover (LULC) classifications and showed SMOTE-ENN improved the performance of Random Forest and Casboost models [7].

In addition, Imran [8] compared two oversampling methods, namely SMOTE and ROS (Random Over Sampling). The results of this study show that both can improve the performance of the classification algorithm. Whereas Rashu [9] and Thammasiri [10] used one of the undersampling methods, namely RUS (Random Under Sampling), the results of research conducted by both of them showed that the RUS method caused a decrease in the performance of the classification algorithm. On the other hand, the research conducted by Kubat [11] used one of the undersampling methods, namely OSS (Sided Selection). The results showed that applying the OSS method can improve the performance of the classification algorithm. Handling class imbalance with a similar approach was also carried out by Noorhalim [12] and Zhihao [13] using the SMOTE method. Both studies show that applying class imbalance handling to datasets can improve the performance of several classification algorithms. In addition, Sajid Ahmed [14] studied handling class imbalances in datasets. This study used ensemble resampling, while the tested methods included SMOTE-Bagging, RUS-Bagging, ADASYN-Bagging, and RYSIN-Bagging. The results of this study indicate that the four methods used have succeeded in improving the performance of the classification algorithm used.

As we know, most of these studies deal with class imbalance using resampling techniques. On the other hand, the resampling technique has weaknesses, namely the risk of duplicating instances and can cause loss of information or patterns in the dataset. This, of course, impacts the performance of a single classifier used. Besides that, the data level approach could also change the composition contained in the dataset. While the approach at the algorithmic level has a weakness, it is not suitable when applied to datasets with a large class imbalance ratio. This study used two approaches: resampling with ADASYN, SMOTE, and SMOTE-ENN. Meanwhile, the classification algorithm used is a single classifier, namely K-Nearest Neighbors (KNN) and Adaboost and Random Forest as meta-learning. This study aims to determine how much handling class imbalance affects the performance of machine learning models. In addition, this study also aims to compare the performance of several model schemes to handle class imbalances in datasets. There are two contributions to this proposed method. First, the proposed method can be a solution for dealing with imbalanced dataset problems in machine learning. Second, the proposed method can be used as a reference for further research on handling imbalanced dataset problems in machine learning.

2. RESEARCH METHOD

2.1. Dataset

In this study, public datasets from the KEEL-Dataset repository were used. There are five binary class datasets with different imbalanced ratios (IR). The datasets used are Pima, Wisconsin, glass1, glass0, and segment0. The following figure 1 is a description of each dataset, including the number of instances, the number of attributes, and the imbalanced ratio (IR) (Table 1).

Table 1. Datasets Description

Code	Dataset	Number of Instances	Attributes	IR
D1	pima	768	8	1.87
D2	wisconsin	683	9	1.86
D3	glass1	214	9	1.82
D4	glass0	214	9	2.06
D5	segment0	2308	19	6.02

2.2. Research Stages

Imbalanced datasets are divided into training data for machine learning and testing data for testing classification models. After that, the oversampling process is carried out using the ADASYN, SMOTE, and SMOTE-ENN methods to balance the data. Then, the resulting data is used for the classification process using the Random Forest, AdaBoost, and K-Nearest Neighbor algorithms. The final stage is to evaluate each method used to measure the performance of the resulting classification. The stages of the research carried out can be seen in Figure 1.

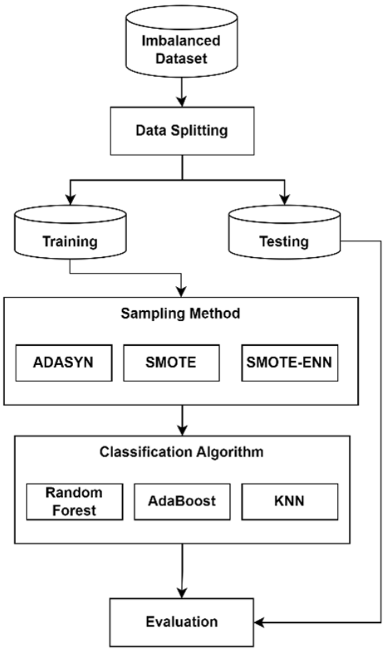

Figure 1. Research stages

2.3. Data Splitting

The initial stage of the unbalanced dataset is divided into two parts. In several studies on imbalanced classes that have been carried out before, a comprehensive scheme for dividing training data and testing data uses the stratified splitting technique. In this study, the data will be divided as follows, 80% of the data will be used as training data for the machine learning model and data to be resampled. Meanwhile, 20% of the data is used for testing machine learning models.

2.4. Adaptive Synthetic Sampling (ADASYN)

Adaptive Synthetic Sampling (ADASYN) is one of the oversampling methods. This method synthesizes data adaptively based on the distribution of positive samples [15]. The advantage of ADASYN is that it can focus data duplication on only one specific area [16], where samples are produced more in areas with low minority sample densities than in areas with high densities. This increase in distribution can reduce data imbalances and help improve classification [17].

2.5. Synthetic Minority Over-sampling Technique (SMOTE)

Synthetic Minority Over-sampling Technique (SMOTE) is widely used for data imbalance issues [18]. SMOTE balances the data by adding new data to the minority class from the resulting artificial data so that the amount of data on the minority and majority classes are balanced. Synthetic data are determined based on their closest neighbors. This method generates new data using equation (1) [19].

$$X_{syn} = x_i + rand(0,1) \times (x_{knn} - x_i) \tag{1}$$

Where, X_{syn} are new synthetic samples from SMOTE process, x_i samples that will be synthetic from minority sample, rand $(0, 1)$ is random values from zeros to ones and x_{knn} The number of neighbor samples will be used to synthesize new samples from minority class samples.

2.6. SMOTE-ENN

SMOTE-ENN is a combination of the Synthetic Minority Over-sampling Technique (SMOTE) and undersampling Edited Nearest Neighbors (ENN) methods [20]. SMOTE calculates the distance between random data and k-nearest neighbors taken from minority classes [21]. ENN selects samples randomly and removes samples that do not have k samples in the nearest neighbors, where ENN can minimize the occurrence of noise in the data [22]. Based on [23], the SMOTE-ENN sampling process is as follows.
Step 1. Choosing random data from minority classes.
Step 2. Finds the distance between the random data and the k-nearest neighbor.
Step 3. Multiply the difference by random values 0 and 1. Then add it to the minority class as a synthetic sample.
Step 4. Repeat steps two and three until obtaining the appropriate proportions.
Step 5. Determining k based on the nearest neighbors. If it cannot be determined, then k is assumed to be the third step.
Step 6. Calculates k-nearest neighbors for the observation class from the remaining observation data. Then return to the majority class.
Step 7. When the observation and majority class of k-nearest neighbors are different, the statement and k-nearest neighbors are removed from the dataset.
Step 8. The iterative process continues until the proportion required for each class has been met (steps 2 3).

2.7. Adaptive Boosting (AdaBoost)

AdaBoost is a boosting method designed for classification and can be applied to various classification algorithms [24]. This algorithm pays more attention to samples misclassified by weak classifiers, thus strengthening the classifier [25].

2.8. K-Nearest Neighbor (KNN)

K-Nearest Neighbor is a popular algorithm used in classification. The algorithm is simple, easy to implement, and produces good results across multiple domains [26]. KNN determines data points based on the distance of the data to its neighbors [27]. The KNN algorithm uses Euclidean Distance to measure the distance of the dataeuclidean Distance equation (2) [28].

$$d(x_i, x_j) = \sqrt{\Sigma_{r=1}^{n}(a_r(x_i) - a_r(x_j))^2} \tag{2}$$

Where $d(x_i, x_j)$ is Euclidean Distance, x_i records to i, x_j to j, and a_r data to r.

2.9. Random Forest

Random forest is an extension of tree-based bagging as a basic learning model [29]. Random forest classification selects a random subset from training data [30]. This algorithm is used to generate accurate predictions [31].

2.10. Evaluation

This study used a confusion matrix to measure classification performance. The confusion matrix represents the results of classifying predicted and actual values shown in Table 2 [32].

Table 2. Confusion Matrix

		Positive Predictions	Negative Predictions
Actual	Positive	TP	FN
	Negative	FP	TN

True Positive (TP) is the number of positive classes that are classified as true as positive, False Positive (FP) is the number of negative classes that are incorrectly classified as positive, True Negative (TN) is the number of negative classes that are correctly classified as negative, and False Negative (FN) is the number of falsely classified positive classes as negative. Therefore, based on the confusion matrix, the evaluation parameters of classification performance accuracy, precision, true positive rate (TPR), true negative rate (TNR), geometric mean (G-Mean) can be calculated by the equation (3), (4), (5), (6), and (7) [33, 34].

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

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

$$True\ Positive\ Rate = \frac{(TP)}{(TP+FN)} \tag{5}$$

$$True\ Negative\ Rate = \frac{(TN)}{(TN+FP)} \tag{6}$$

$$G-mean = \sqrt{Sensitivity * Specificity} \tag{7}$$

3. RESULT AND ANALYSIS

The initial stage of the unbalanced dataset is divided into two parts. In several studies on imbalanced classes that have been carried out before, a comprehensive scheme for dividing training data and testing data uses the stratified splitting technique. In this study, the data will be divided as follows, 80% of the data will be used as training data for the machine learning model and data to be resampled. Meanwhile, 20% of the data is used for testing machine learning models. Table 3 shows the data for the training process and the data for validation or testing.

Table 3. Stratified Splitting Scheme

Dataset	Training	Testing
D1	614	154
D2	455	114
D3	171	43
D4	171	43
D5	1846	462

3.1. Resampling Process

After the training and testing data are determined, a resampling process is carried out on the training data using ADASYN, SMOTE, and SMOTE-ENN. The distribution of positive and negative classes in the training data before and after applying the sampling method can be seen in Table 4 and Table 5.

Table 4. Class Distribution of Training Set Before Resampling Process

Dataset	Positive	Negative
D1	221	393
D2	165	290
D3	57	114
D4	61	110
D5	259	1587

Dataset 1 has 221 positive class samples and 393 negative samples. Dataset 2 has 165 positive class samples and 290 negative samples. Dataset 3 has 57 positive samples and 114 negative samples. Dataset 4 has 61 positive samples and 110 negative samples. Dataset 5 has 259 positive samples and 1587 negative samples. The number of samples in both classes (positive and negative) indicates a class imbalance before the sampling method is applied. The number of samples in the positive class is smaller than those in the negative class.

Table 5. Class Distribution of Training Set After Resampling Process

Dataset	Resampling	Positive	Negative
D1	ADASYN	393	393
	SMOTE	393	393
	SMOTE-ENN	210	164
D2	ADASYN	286	290
	SMOTE	290	290
	SMOTE-ENN	249	258
D3	ADASYN	124	114
	SMOTE	114	114
	SMOTE-ENN	83	79
D4	ADASYN	116	110
	SMOTE	110	110
	SMOTE-ENN	85	67
D5	ADASYN	1591	1587
	SMOTE	1587	1587
	SMOTE-ENN	1587	1549

The resampling results of each technique in Table 5 show that the training set conditions after resampling using SMOTE result in the number of instances of the two classes being the same. This is because SMOTE, apart from performing data synthesis, also performs data duplication. While using ADASYN and SMOTE-ENN, there tends to be little difference in the number of instances between the two classes.

3.2. Classification Performance

The following process is classified using AdaBoost, K-Nearest Neighbor, and Random Forest. Classification performance is evaluated with a confusion matrix, where the metrics used are accuracy, precision, recall, true negative rate, and g-mean score. A comparison of classification performance between the original data and after resampling can be seen in Table 6, Tabel 7, Table 8, Table 9, and Table 10.

Table 6. Accuracy Values Each Model

Dataset	Resampling Techniques	Random Forest	AdaBoost	K-Nearest Neighbor
D1	Original Data	0.786	0.779	0.721
	ADASYN	0.786	0.76	0.63
	SMOTE	0.799	0.766	0.675
	SMOTE-ENN	0.753	0.76	0.734
D2	Original Data	0.965	0.956	0.912
	ADASYN	0.947	0.947	0.895
	SMOTE	0.93	0.939	0.912
	SMOTE-ENN	0.93	0.939	0.939
D3	Original Data	0.791	0.767	0.767
	ADASYN	0.837	0.744	0.837
	SMOTE	0.791	0.791	0.791
	SMOTE-ENN	0.767	0.791	0.814
D4	Original Data	0.86	0.93	0.791
	ADASYN	0.93	0.86	0.767
	SMOTE	0.93	0.907	0.791
	SMOTE-ENN	0.837	0.651	0.674
D5	Original Data	0.991	0.998	0.987
	ADASYN	0.998	0.994	0.989
	SMOTE	0.989	0.994	0.983
	SMOTE-ENN	0.998	0.998	0.981

The accuracy in the classification results on the original data showed quite good values. However, the classification results cannot be trusted because the dataset's condition is unbalanced. Table 6 accuracy values on the ADASYN+RF combination resulted in the best performance compared to other methods with accuracy values of 0.786, 0.947, 0.837, 0.93, and 0.998.

Table 7. Precision Values Each Model

Dataset	Resampling Techniques	Random Forest	AdaBoost	K-Nearest Neighbor
D1	Original Data	0.659	0.633	0.538
	ADASYN	0.621	0.574	0.431
	SMOTE	0.654	0.585	0.476
	SMOTE-ENN	0.569	0.583	0.545
D2	Original Data	0.939	0.957	0.878
	ADASYN	0.918	0.902	0.83
	SMOTE	0.898	0.9	0.849
	SMOTE-ENN	0.868	0.9	0.917
D3	Original Data	0.857	0.846	0.846
	ADASYN	0.837	0.75	0.8
	SMOTE	0.791	0.917	0.813
	SMOTE-ENN	0.767	0.813	0.867
D4	Original Data	0.667	0.8	0.5
	ADASYN	0.8	0.615	0.467
	SMOTE	0.8	0.727	0.5
	SMOTE-ENN	0.563	0.333	0.368
D5	Original Data	1.0	1.0	0.971
	ADASYN	1.0	1.0	0.945
	SMOTE	0.971	1.0	0.931
	SMOTE-ENN	1.0	1.0	0.918

Table 7 shows the precision values in the ADASYN, SMOTE, and SMOTE-EN sampling methods. In the Random Forest and AdaBoost algorithms, the precision value is seen to have decreased. However, in SMOTE-ENN + KNN, the precision value is quite good compared to the original data and combined KNN with other sampling methods.

Table 8. True Positive Rate Values Each Model

Dataset	Resampling Techniques	Random Forest	AdaBoost	K-Nearest Neighbor
D1	Original Data	0.617	0.66	0.596
	ADASYN	0.766	0.83	0.66
	SMOTE	0.723	0.809	0.638
	SMOTE-ENN	0.787	0.745	0.766
D2	Original Data	0.979	0.936	0.915
	ADASYN	0.957	0.979	0.936
	SMOTE	0.936	0.957	0.957
	SMOTE-ENN	0.979	0.957	0.936
D3	Original Data	0.632	0.579	0.579
	ADASYN	0.789	0.632	0.842
	SMOTE	0.632	0.579	0.684
	SMOTE-ENN	0.684	0.684	0.684
D4	Original Data	0.667	0.889	0.667
	ADASYN	0.889	0.889	0.778
	SMOTE	0.889	0.889	0.778
	SMOTE-ENN	1.0	0.667	0.778
D5	Original Data	0.943	0.986	0.943
	ADASYN	0.986	0.957	0.986
	SMOTE	0.957	0.957	0.957
	SMOTE-ENN	0.986	0.986	0.957

True positive rate results prove that classifiers predict minority classes better. Table 8 shows the true positive rate results on each method used. In dataset 1, the true positive rate is best indicated by the ADASYN+AB method. In dataset 2, true positive rates are best generated by ADASYN+AB and SMOTE-ENN+RF. In dataset 3, the true positive rate value is best generated by ADASYN+KNN. In dataset 4, the true positive rate results are best shown by the SMOTE-ENN+RF combine, and in dataset 5, the true positive rate results are best shown by ADASYN+AB, ADASYN+KNN, SMOTE-ENN+RF, and SMOTE-ENN+AB. Based on the true positive rate results, it can be seen that the overall true positive rate value on the original data is lower than the true positive rate value in the ADASYN, SMOTE, and SMOTE-ENN methods as well as the ADASYN and SMOTE-ENN methods showing better true positive rate results compared to the SMOTE method.

Table 9. True Negative Rate Values Each Model

Dataset	Resampling Techniques	Random Forest	AdaBoost	K-Nearest Neighbor
D1	Original Data	0.86	0.832	0.776
	ADASYN	0.794	0.729	0.617
	SMOTE	0.832	0.748	0.692
	SMOTE-ENN	0.738	0.766	0.72
D2	Original Data	0.955	0.97	0.91
	ADASYN	0.94	0.925	0.866
	SMOTE	0.925	0.925	0.881
	SMOTE-ENN	0.896	0.925	0.94
D3	Original Data	0.917	0.917	0.917
	ADASYN	0.789	0.833	0.833
	SMOTE	0.917	0.958	0.875
	SMOTE-ENN	0.833	0.875	0.917
D4	Original Data	0.912	0.941	0.824
	ADASYN	0.941	0.853	0.765
	SMOTE	0.941	0.912	0.794
	SMOTE-ENN	0.794	0.647	0.647
D5	Original Data	1.0	1.0	0.995
	ADASYN	1.0	1.0	0.99
	SMOTE	0.995	1.0	0.987
	SMOTE-ENN	1.0	1.0	0.985

The true negative rate value indicates the classifier's ability to predict negative classes. Table 9 shows the true negative rate value, where the true negative rate value shows a high result in the original data. However, as seen in Table 8, the resulting recall value is lower than ADASYN, SMOTE, and SMOTE-ENN. This can happen because, in the original data that experience a class imbalance, the classifier will tend to classify the majority class (negative) and ignore the minority class (positive) so that the true negative rate value in the original data can be higher than the results when the data is in a balanced state or after the implementation of the ADASYN, SMOTE, and SMOTE-ENN sampling methods.

Table 10. Geometric Mean Values Each Model

Dataset	Resampling Techniques	Random Forest	AdaBoost	K-Nearest Neighbor
D1	Original Data	0.728	0.741	0.68
	ADASYN	0.78	0.778	0.638
	SMOTE	0.776	0.777	0.664
	SMOTE-ENN	0.762	0.755	0.742
D2	Original Data	0.967	0.953	0.913
	ADASYN	0.949	0.952	0.9
	SMOTE	0.931	0.941	0.918
	SMOTE-ENN	0.936	0.941	0.938
D3	Original Data	0.761	0.728	0.728
	ADASYN	0.831	0.725	0.838
	SMOTE	0.761	0.745	0.774
	SMOTE-ENN	0.755	0.774	0.792
D4	Original Data	0.78	0.915	0.741
	ADASYN	0.915	0.871	0.771
	SMOTE	0.915	0.9	0.786
	SMOTE-ENN	0.891	0.657	0.709
D5	Original Data	0.971	0.993	0.969
	ADASYN	0.993	0.978	0.988
	SMOTE	0.976	0.978	0.972
	SMOTE-ENN	0.933	0.993	0.971

Table 10 shows the g-mean values. The g-mean results are more realistic than general accuracy, which will still give high results despite minority class misclassifications [35]. Of all the datasets tested, ADASYN+RF excelled in three datasets, namely datasets 1, 4, and 5. ADASYN+KNN at dataset 3 and SMOTE+RF at dataset 4. ADASYN+RF produced a better g-mean value than the original data and other sampling methods.

Based on the experimental results that have been carried out from the five indicators used, namely accuracy, precision, true positive rate, true negative rate, and geometric mean, it shows that handling class imbalances in datasets greatly influences the performance of machine learning models. Integrating ADASYN and Random Forest predominantly gave better results than without sampling or combining classification algorithms and other sampling methods. However, in some datasets, other methods showed better results. For example, the results from the ADSYN Resampling and Random Forests models are more than the others because there are no duplicate sample values from the oversampling process. Whereas in SMOTE, there are still some sample duplications, so the results are not as optimal as in ADASYN. Therefore, ADSYN Resampling and Random Forests generally produce better performance than other models such as SMOTE [12, 13] and SMOTE-ENN [7].

4. CONCLUSION

Based on the classification results, implementing sampling methods in each classification model shows an improvement in classification performance. The classification performance on the algorithm without sampling looks quite good, but this is invalid because the classifier only predicts the majority class and the presence of minority class misclassifications. Therefore, the classification model gives different results based on the sampling method applied. Overall, the method that produces the best performance is the combination of ADASYN and Random Forest which is shown by the accuracy, precision, true positive rate, true negative rate, and g-mean. The results from ADSYN Resampling and the Random Forests model are more than the other models because there are no duplicated sample values from the oversampling process. Whereas in SMOTE, there are still several sample duplications, so the results are not as optimal as in ADASYN. Therefore, ADSYN Resampling and Random Forests generally produce better performance than other models such as SMOTE and SMOTE-ENN. The results of this experiment can also be used as a reference for further research on handling imbalanced dataset problems in machine learning. For further research, you can use datasets with a more significant number of samples or multiclass datasets. In addition, the sampling method can be combined with other classification algorithms.

5. ACKNOWLEDGEMENTS

Thanks to the Department of Research and Community Service, Universitas Amikom Yogyakarta, for funding and supporting this research.

6. DECLARATIONS

AUTHOR CONTIBUTION
Cherfly Kaope is a model maker and data collector. Yoga Pristyanto analyzes and writes articles.

FUNDING STATEMENT
Researchers would like to express their deepest gratitude to the Department of Research and Community Service, Universitas Amikom Yogyakarta, for funding and supporting this research.

COMPETING INTEREST
The future research direction is to develop further models for class imbalance problems with various fields of knowledge that intersect with the application of artificial intelligence.

REFERENCES

[1] W. Ustyannie and S. Suprapto, "Oversampling Method To Handling Imbalanced Datasets Problem in Binary Logistic Regression Algorithm," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 14, no. 1, p. 1, 2020.

[2] H. Ali, M. N. M. Salleh, R. Saedudin, K. Hussain, and M. F. Mushtaq, "Imbalance class problems in data mining: A review," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, no. 3, pp. 1552–1563, 2019.

[3] N. S. Ramadhanti, W. A. Kusuma, and A. Annisa, "Optimasi Data Tidak Seimbang pada Interaksi Drug Target dengan Sampling dan Ensemble Support Vector Machine," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 6, pp. 1221–1230, dec 2020.

[4] I. Lin, O. Loyola-González, R. Monroy, and M. A. Medina-Pérez, "A Review of Fuzzy and Pattern-Based Approaches for Class Imbalance Problems," *Applied Sciences*, vol. 11, no. 14, pp. 1–23, jul 2021.

[5] N. Chamidah, M. M. Santoni, and N. Matondang, "Pengaruh Oversampling pada Klasifikasi Hipertensi dengan Algoritma Naïve Bayes, Decision Tree, dan Artificial Neural Network (ANN)," *JURNAL RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 1, no. 3, pp. 635–641, 2017.

[6] N. G. Ramadhan, "Comparative Analysis of ADASYN-SVM and SMOTE-SVM Methods on the Detection of Type 2 Diabetes Mellitus," *Scientific Journal of Informatics*, vol. 8, no. 2, pp. 276–282, 2021.

[7] H. L. Ngo, H. D. Nguyen, P. Loubiere, T. V. Tran, G. erban, M. Zelenakova, P. Brecan, and D. Laffly, "The Composition of Time-Series Images and Using The Technique SMOTE ENN for Balancing Datasets in Land Use/Cover Mapping," *Acta Montanistica Slovaca*, vol. 27, no. 2, pp. 342–359, 2022.

[8] M. Imran, M. Afroze, S. K. Sanampudi, A. Abdul, and M. Qyser, "Data Mining of Imbalanced Dataset in Educational Data Using Weka Tool," *International Journal of Engineering Science and Computing*, vol. 6, no. 6, pp. 7666–7669, 2016.

[9] R. I. Rashu, N. Haq, and R. M. Rahman, "Data Mining Approaches to Predict Final Grade by Overcoming Class Imbalance Problem," in *2014 17th International Conference on Computer and Information Technology, ICCIT 2014*, 2014, pp. 14–19.

[10] D. Thammasiri, D. Delen, P. Meesad, and N. Kasap, "A Critical Assessment of Imbalanced Class Distribution Problem: The Case of Predicting Freshmen Student Attrition," *Expert Systems with Applications*, vol. 41, no. 2, pp. 321–330, 2014.

[11] M. Kubat and S. Matwin, "Addressing the Curse of Imbalanced Training Sets: One Sided Selection," in *International Conference on Machine Learning*, vol. 97, 1997, pp. 179–186.

[12] N. Noorhalim, A. Ali, and S. M. Shamsuddin, "Handling Imbalanced Ratio for Class Imbalance Problem Using SMOTE," in *Proceedings of the Third International Conference on Computing, Mathematics and Statistics (iCMS2017)*, 2017, pp. 19–29.

[13] Z. Peng, F. Yan, and X. Li, "Comparison of The Different Sampling Techniques for Imbalanced Classification Problems in Machine Learning," in *Proceedings - 2019 11th International Conference on Measuring Technology and Mechatronics Automation, ICMTMA 2019*, 2019, pp. 431–434.

[14] S. Ahmed, A. Mahbub, F. Rayhan, R. Jani, S. Shatabda, and D. M. Farid, "Hybrid Methods for Class Imbalance Learning Employing Bagging with Sampling Techniques," in *2nd International Conference on Computational Systems and Information Technology for Sustainable Solutions, CSITSS 2017*. IEEE, 2018, pp. 1–5.

[15] H. Ding, L. Chen, L. Dong, Z. Fu, and X. Cui, "Imbalanced Data Classification: A KNN and Generative Adversarial Networks-Based Hybrid Approach for Intrusion Detection," *Future Generation Computer Systems*, vol. 131, no. June, pp. 240–254, jun 2022.

[16] P. Wibowo and C. Fatichah, "An In-Depth Performance Analysis of The Oversampling Techniques for High-Class Imbalanced Dataset," *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 7, no. 1, pp. 63–71, 2021.

[17] R. Gupta, R. Bhargava, and M. Jayabalan, "Diagnosis of Breast Cancer on Imbalanced Dataset Using Various Sampling Techniques and Machine Learning Models," in *2021 14th International Conference on Developments in eSystems Engineering (DeSE)*, dec 2021, pp. 162–167.

[18] R. R. Rao and K. Makkithaya, "Learning from A Class Imbalanced Public Health Dataset: A Cost-Based Comparison of Classifier Performance," *International Journal of Electrical and Computer Engineering*, vol. 7, no. 4, pp. 2215–2222, 2017.

[19] N. Santoso, W. Wibowo, and H. Hikmawati, "Integration of Synthetic Minority Oversampling Technique for Imbalanced Class," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 13, no. 1, p. 102, jan 2019.

[20] A. Indrawati, H. Subagyo, A. Sihombing, W. Wagiyah, and S. Afandi, "Analyzing The Impact of Resampling Mehod for Imbalanced Data Text in Indonesian Scientific Articles Categorization," *BACA: JURNAL DOKUMENTASI DAN INFORMASI*, vol. 41, no. 2, pp. 133–141, dec 2020.

[21] S. Akter, D. Das, R. U. Haque, M. I. Quadery Tonmoy, M. R. Hasan, S. Mahjabeen, and M. Ahmed, "AD-CovNet: An Exploratory Analysis Using A Hybrid Deep Learning Model to Handle Data Imbalance, Predict Fatality, and Risk Factors in Alzheimer's Patients with COVID-19," *Computers in Biology and Medicine*, vol. 146, no. July, pp. 1–19, jul 2022.

[22] T. Sasada, Z. Liu, T. Baba, K. Hatano, and Y. Kimura, "A Resampling Method for Imbalanced Datasets Considering Noise and Overlap," in *Procedia Computer Science*, vol. 176. Elsevier B.V., 2020, pp. 420–429.

[23] A. L. Karn, C. A. T. Romero, S. Sengan, A. Mehbodniya, J. L. Webber, D. A. Pustokhin, and F.-D. Wende, "Fuzzy and SVM Based Classification Model to Classify Spectral Objects in Sloan Digital Sky," *IEEE Access*, vol. 10, pp. 101 276–101 291, 2022.

[24] A. Subasi, M. Balfaqih, Z. Balfagih, and K. Alfawwaz, "A Comparative Evaluation of Ensemble Classifiers for Malicious Webpage Detection," in *Procedia Computer Science*. Elsevier B.V., 2021, pp. 272–279.

[25] Y. Wang and L. Feng, "Improved Adaboost Algorithm for Classification Based on Noise Confidence Degree and Weighted Feature Selection," *IEEE Access*, vol. 8, pp. 153 011–153 026, 2020.

[26] Y. M. Wazery, E. Saber, E. H. Houssein, A. A. Ali, and E. Amer, "An Efficient Slime Mould Algorithm Combined with K-Nearest Neighbor for Medical Classification Tasks," *IEEE Access*, vol. 9, pp. 113 666–113 682, 2021.

[27] F. M. M. Shamrat, S. Chakraborty, M. M. Imran, J. N. Muna, M. M. Billah, P. Das, and M. O. Rahman, "Sentiment Analysis on Twitter Tweets about COVID-19 Vaccines Using NLP and Supervised KNN Classification Algorithm," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 23, no. 1, pp. 463–470, 2021.

[28] A. R. Isnain, J. Supriyanto, and M. P. Kharisma, "Implementation of K-Nearest Neighbor (K-NN) Algorithm For Public Sentiment Analysis of Online Learning," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 15, no. 2, pp. 121–130, apr 2021.

[29] B. Solihah, A. Azhari, and A. Musdholifah, “The Empirical Comparison of Machine Learning Algorithm for the Class Imbalanced Problem in Conformational Epitope Prediction,” *JUITA: Jurnal Informatika*, vol. 9, no. 1, pp. 131–138, may 2021.

[30] V. K. Gupta, A. Gupta, D. Kumar, and A. Sardana, “Prediction of COVID-19 Confirmed, Death, and Cured Cases in India Using Random Forest Model,” *Big Data Mining and Analytics*, vol. 4, no. 2, pp. 116–123, 2021.

[31] J. Zeffora and Shobarani, “Optimizing Random Forest Classifier with Jenesis-Index on An Imbalanced Dataset,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 26, no. 1, pp. 505–511, 2022.

[32] A. N. Kasanah, M. Muladi, and U. Pujianto, “Penerapan Teknik SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Objektivitas Berita Online Menggunakan Algoritma KNN,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 3, no. 2, pp. 196–201, 2019.

[33] A. A. Salih and A. M. Abdulazeez, “Evaluation of Classification Algorithms for Intrusion Detection System: A Review,” *Journal of Soft Computing and Data Mining*, vol. 02, no. 01, pp. 31–40, 2021.

[34] A. S. Desuky, A. H. Omar, and N. M. Mostafa, “Boosting with Crossover for Improving Imbalanced Medical Datasets Classification,” *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 5, pp. 2733–2741, 2021.

[35] Z. P. Agusta and Adiwijaya, “Modified Balanced Random Forest for Improving Imbalanced Data Prediction,” *International Journal of Advances in Intelligent Informatics*, vol. 5, no. 1, pp. 58–65, 2019.