Jurnal Ilmu Komputer dan Informasi (Journal of Computer Science and Information)
17/1 (2024), 9-17. DOI: http://dx:doi:org/10:21609/jiki:v17i1.1177

Improving Classification Performance on Imbalance Medical Data using Generative Adversarial Network

Siska Rahmadani[1], Agus Subekti[1], Muhammad Haris[1]

[1]Computer Science, Faculty of Information Technology, Universitas Nusa Mandiri,
Depok, Jawa Barat, 16424, Indonesia

E-mail: 14002456@nusamandiri.ac.id

Abstract

In many real-world applications, the problem of data imbalance is a common challenge that significantly affects the performance of machine learning algorithms. Data imbalance means each target of classes is not balanced. This problem often appears in medical data, where the positive cases of a disease or condition are much fewer than the negative cases. In this paper, we propose to explore the oversampling-based Generative Adversarial Networks (GAN) method to improve the performance of the classification algorithm over imbalanced medical datasets. We expect that GAN will be able to learn the actual data distribution and generate synthetic samples that are similar to the original ones. We evaluate our proposed methods on several metrics: Recall, Precision, F1 score, AUC score, and FP rate. These metrics measure the ability of the classifier to correctly identify the minority class and reduce the false positives and false negatives. Our experimental results show that the application of GAN performs better than other methods in several metrics across datasets and can be used as an alternative method to improve the performance of the classification model on imbalanced medical data.

Keywords: *Classification, GAN, Imbalance, Machine Learning, Oversampling*

1. Introduction

Various fields, such as market analysis, telecommunications, and the health sector, have widely applied data mining. Data-mining processes involve using statistics, mathematics, artificial intelligence, and machine learning techniques to extract information and knowledge from large amounts of data [1]. Data mining extracts patterns from data and uses those patterns to build models that make predictions.

As data processing technology advances, better tools are needed to help us process and analyze the massive amounts of information we generate. One way to improve patient treatment in health care is to use advanced algorithms for data mining and find patterns among similar conditions [2]. Doctors can use the ability to predict disease development to provide early treatment to patients. Medical applications are not the only ones that benefit from data integration and mining; it is also important for companies in other industries. For example, retailers can use data mining tools to understand customer behavior and improve business operations.

Real-world data, such as data related to fault detection [3], [4]; fraud detection [5], [6], and medical diagnosis [7]–[9], often have data imbalance problems. A dataset is called an imbalance if it does not represent the classified categories evenly [10]. Data imbalance occurs when a dataset does not represent the classified categories evenly. In other words, if one or more target variables has many instances while others have very few (or none at all), then this is an example of data imbalance [11].

Uneven classes in the dataset can bias an algorithm towards one class, reducing performance despite high accuracy [12]. An imbalanced dataset can misidentify important classes, especially for medical data classification. Most machine learning models assume even data distribution, which can cause inaccurate predictions. This can have serious consequences in the prediction of infectious diseases. For example, suppose we want to detect Covid-19 patients using a classification model. A False

Positive occurs when a patient who does not have Covid-19 is predicted to be positive by the model. This may lead to unnecessary medical tests for the patient. A False Negative occurs when a patient who has Covid-19 is predicted to be negative by the model. This may result in delayed treatment and increased risk of spreading the virus, which can harm many people.

Traditional machine learning algorithms aim to reduce errors by increasing accuracy but ignore the data imbalance problem [13]–[15]. Therefore, we need to handle the class imbalance in the data used for classification from the start. Resampling is a common technique used to overcome data imbalance. Oversampling and undersampling are common resampling techniques. Undersampling reduces the number of instances or the majority target sample, while oversampling increases the number of instances or minority target samples by generating new instances or repeating several instances [16].

Several studies suggest that the oversampling technique has a better ability than undersampling based on comparing the performance of various classification models [17], [18]. Another study showed that using the oversampling technique with the Random Forest model can improve the accuracy of lung cancer screening and reduce the risk of false diagnosis [19].

The Synthetic Minority Oversampling Technique (SMOTE) algorithm is a simple and effective over-sampling method to generate class samples by balancing the classes in the dataset by increasing the minority classes quantitatively [20]. However, SMOTE has the shortcomings of data with blurred boundaries, much noise, or data with class imbalance because SMOTE makes the algorithm change the characteristics of the spatial distribution of minority samples in the original sample set [21]. Several improved SMOTE algorithms generate new sample regions for synthesis data by combining clusters or selecting samples within classes. In 2020, Naseriparsa et al. proposed RSMOTE, a modification of the SMOTE that divides the minority sample into four regions (normal, semi-normal, semi-critical, and critical) based on a minority sample density analysis [22].

As a machine learning technique, GAN produces high-fidelity new images for new training data. The GAN consists of two models that train simultaneously: a generator trained to produce false or synthetic data and a discriminator trained to distinguish fake data generated by the generator from the original data [23].

The discriminator achieves optimal results when it can no longer determine which input is real. The characteristics of GAN allow applying oversampling studies because constructing a neural network based on adversarial training enables the creation of artificial data similar to the original data.

Several studies use GAN or improved GAN to produce new synthetic image data to overcome an imbalance problem. In 2020, Rezaei et al. used GAN to generate synthetic images [24]. In 2021, Sharma et al. combined features from GAN and SMOTE to address the class imbalance problem using a data augmentation approach [25].

Generative adversarial networks (GAN) are not only capable of generating real image data but also tabular data, such as medical or educational records, by using deep neural networks to model the joint distribution of discrete and continuous variables [26]. GAN’s ability to generate new data and study data distribution enables its application to tabular data.

Conditional GANs (CTGAN) proposed by Xu et al. as a synthetic tabular data generator to overcome some of the problems caused by imbalanced data [27]. CTGAN outperformed all methods over Bayesian Networks in at least 87.5% of the datasets used.

GAN generates additional data for minority classes by oversampling with the Conditional Tabular GAN (CTGAN) architecture. The generator adjusts the tabular data input and receives supplementary information to produce samples under the specified class conditions [6]. The experimental results show that the proposed method performs better than other oversampling methods on several evaluation metrics: Accuracy, Precision score, F1 score, and AUC.

Another study, an approach of Conditional Wasserstein GAN, can effectively model tabular datasets with numerical and categorical variables and give special attention to downstream classification tasks through additional loss of classifiers [28]. The results demonstrate that the GAN architecture for tabular data and the proposed extension merit consideration for future research.

A technique to overcome imbalanced data in datasets is a research theme that deserves to be applied and explored because data imbalance problems are often found in the real world and affect the performance of machine learning algorithms. This study examined GAN's ability to improve the classification model using data imbalance within medical datasets, then compared it with SMOTE methods and methods proposed by the previous study, namely RSMOTE [22].

2. Method

We conducted this research in several stages, i.e., data preprocessing, resampling, modeling, and evaluation. Before applying the classification algorithms, we performed data preprocessing, including data cleaning and train test split. We applied k-fold cross-validation to estimate how well the machine-learning model would perform.

We follow the dataset used by Naseriparsa et al. [22], which is the most relevant and recent work on this topic. We took the datasets from the UCI repository [29].

Table 1 presents the specifications of the datasets we used in this study. The value in the class minority and class majority columns indicates the number of samples belonging to the minority and majority classes in the dataset, respectively. In a classification task, the majority class is the class that has the most instances in the dataset, while the minority class is the class that has the least instances in the dataset. The imbalance ratio is a metric that measures the degree of class imbalance in a dataset; it is described as a ratio between the samples in the majority class and the minority class [30]. It shows the hepatitis dataset has a more significant imbalance ratio than the others; thus, the class imbalance on the hepatitis dataset might significantly impact the classification performance.

Table 1. The characteristic of the dataset

Dataset	Atribut	Class Minority	Class Majority	Imbalance Ratio
Hepatitis	20	32	123	3,28
Diabetes	9	268	500	1,86
WDBC	32	212	354	1,68
Heart disease	14	120	150	1,25

Preprocessing

The first step of preprocessing is to clean the data by performing some operations. These operations include checking duplicate data that may cause bias or redundancy, checking missing values that may affect the accuracy or validity of the analysis, and checking zero values that may indicate errors or outliers.

Both the WDBC and the heart disease datasets are clean and have no duplicates or missing values. The only preprocessing step we performed was dropping the *id* column from the WDBC dataset, as it was irrelevant for the analysis.

The diabetes dataset is clean and has no duplicate or missing values. However, some columns contain unrealistic values of 0 for human attributes such as *glucose*, *blood pressure*, *skin thickness*, *insulin*, and *BMI*. To handle these values, we replace them with the mean and value of the particular column.

The hepatitis dataset has no duplicate values, but it has missing values in several attributes. These attributes are *steroids, fatigue, malaise, anorexia, liver_big, liver_firm, spleen_palpable, spiders, ascites*, *varices*, *bilirubin, alk_phosphate, sgot, albumin*, and *protime*. We drop the *protime* column because it has too many missing values to be useful for the analysis. To handle the missing values, we use the fillna method that replaces NA/NaN values with a specified method or value.

For the categorical values, the encoding process is applied to convert them into numeric data types, such as integers or floats, because machine learning algorithms can only process data in numerical form.

The preprocessing process is then continued by splitting the data into two subsets: train and test data. The train data is used to train the machine learning models, while the test data is used to evaluate their performance. The ratio of train and test data is set to 90:10. We chose a 90:10 ratio for train test split as we had a small dataset and wanted to use cross-validation technique to reduce the variance of model evaluation, which required splitting the data into multiple folds [31].

Resampling

This research aims to address the problem of imbalanced medical data using oversampling method. A method is applied to generate new data for the training data, so the data for each class is balanced. The technique used in this research is the traditional SMOTE and GAN, based on oversampling. The SMOTE module is imported using the imblearn library from Scikit-learn.

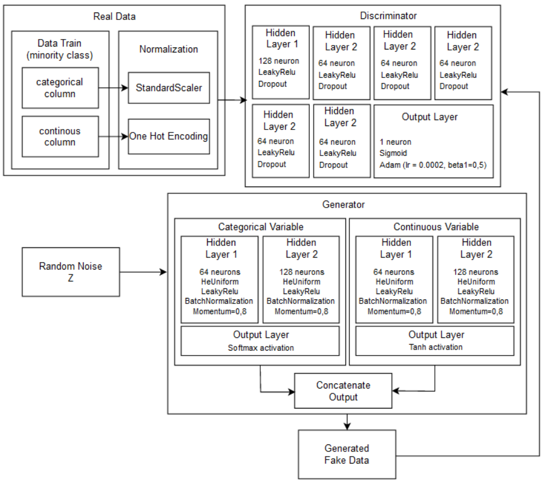

Real Data
Data Train (minority class)
categorical column
continous column
Normalization
StandardScaler
One Hot Encoding
Discriminator
Hidden Layer 1
128 neuron LeakyRelu Dropout
Hidden Layer 2
64 neuron LeakyRelu Dropout
Hidden Layer 2
64 neuron LeakyRelu Dropout
Hidden Layer 2
64 neuron LeakyRelu Dropout
Hidden Layer 2
64 neuron LeakyRelu Dropout
Hidden Layer 2
64 neuron LeakyRelu Dropout
Output Layer
1 neuron Sigmoid Adam (lr = 0.0002, beta1=0,5)
Generator
Random Noise Z
Categorical Variable
Hidden Layer 1
64 neurons HeUniform LeakyRelu BatchNormalization Momentum=0,8
Hidden Layer 2
128 neurons HeUniform LeakyRelu BatchNormalization Momentum=0,8
Output Layer
Softmax activation
Continuous Variable
Hidden Layer 1
64 neurons HeUniform LeakyRelu BatchNormalization Momentum=0,8
Hidden Layer 2
128 neurons HeUniform LeakyRelu BatchNormalization Momentum=0,8
Output Layer
Tanh activation
Concatenate Output
Generated Fake Data

Fig. 1. GAN based oversampling architecture

This research uses the CTGAN architecture to generate synthetic tabular data. CTGAN has been shown to outperform other tabular data generation methods regarding data quality and utility [27]. Based on GAN architecture shown in Fig. 1, the data with minority class is separated into a subset used to train data for GAN implementation. In diabetes and WDBC datasets, the separated dataset in numerical form is forwarded to the GAN for training after the data is normalized using the StandardScaler. The hepatitis and heart disease datasets' subsets are numeric and categorical. Numerical data is normalized using StandardScaler, and categorical data are transformed using OneHotEncoding before being passed to GAN.

The generator takes input from latent space and generates a new synthetic data sample. The LeakyRelu activation unit is used in the generator and discriminator to handle negative values. In addition, batch normalization is used in each layer to standardize the activation of the previous layer and stabilize the training process. In each output layer, the Softmax activation function is used for categorical variables, while the Tanh activation function is used for continuous variables.

In addition, batch normalization is used in each layer to standardize the activation of the previous layer and stabilize the training process. In each output layer, the Softmax activation function is used for categorical variables, while the Tanh activation function is used for continuous variables.

The discriminator model takes the data as a vector sample by the discriminator model, which then displays a classification prediction of whether the selection is real or fake. Because the prediction is a binary classification problem, Sigmoid activation is used at the output layer, and the loss function Binary Cross-entropy is used.

We added dropout on the discriminator layer to prevent overfitting and help stabilize the training. In addition, Adam's optimization algorithm is used with learning rate and beta1 adjusted for each dataset. The training data is then passed to the GAN model to be trained with the training progress up to 300 epochs, batch_size 64. Trained GAN is used to generate new data similar to the minority data.

Modeling and evaluation

We use Kullback-Leibler Divergence (KL-divergence) and Cosine Similarity to calculate the distribution similarity between the synthetic and actual data columns. KL-divergence measures how much two probability distributions differ from each other [32]. The amount of information lost while comparing two distributions is measured using a notion from information theory. The lower the KL divergence, the closer the two distributions are to one another. Cosine similarity measured the extent to which two objects were alike [33]. The more similar the data objects are, the closer the value is to 1.

The classification process uses Random Forest, Logistic Regression, SVM, and Naïve Bayes model. In the medical field, identifying the minority class is considered more important. Because the classifier may produce incorrect information regarding the minority class, accuracy alone is not suitable as an evaluation measure for the classification process [34], [35].

We evaluated the machine learning model using cross-validation with k=5. Cross_val_score accepts an estimator, a dataset, and a scoring parameter as inputs and gives back an array of scores for each fold. We chose the scoring parameter to be: Recall (Weighted Average), Precision (Weighted Average), Precision (Weighted Average), F1-Score (Weighted Average), AUC, and FP-rate. We used Google Colab's jupyter notebook tools in Python to conduct the experiments.

3. Result and Discussion

In Table 2, from the results of the performance evaluation before resampling, it can be seen that the results of the model classification using Hepatitis and WDBC datasets have a low-performance value. However in the WDBC dataset, the Random Forest model can deal with the problem, evident by the pretty good performance outcomes.

Table 2. Classification performance before resampling

Dataset	Model	R	P	F1	AUC	FPR
Hepatitis	RF	0.45	0.50	0.50	0.82	0.06
	LR	0.51	0.61	0.56	0.86	0.06
	SVM	0.00	0.00	0.00	0.65	0.04
	NB	0.72	0.46	0.52	0.81	0.09
Diabetes	RF	0.59	0.69	0.66	0.83	0.16
	LR	0.56	0.74	0.62	0.83	0.14
	SVM	0.47	0.73	0.58	0.82	0.13
	NB	0.61	0.64	0.61	0.81	0.14
WDBC	RF	0.93	0.96	0.94	0.99	0.03
	LR	0.59	0.15	0.22	0.52	0.10
	SVM	0.00	0.07	0.01	0.55	0.07
	NB	0.03	0.30	0.05	0.89	0.06
Heart Disease	RF	0.88	0.81	0.83	0.91	0.13
	LR	0.88	0.82	0.86	0.89	0.12
	SVM	0.87	0.65	0.74	0.72	0.14
	NB	0.82	0.86	0.86	0.90	0.13
Average		0.55	0.58	0.54	0.80	0.10

In the diabetes dataset, the model's performance is moderate, although several matrics, especially Recall (R), need to be improved. Meanwhile, from the evaluation results on the heart disease dataset, the impact of the imbalance data does not have a significant effect in terms of the performance value, which is quite good; this possibility relates to the low imbalance ratio.

Before we generated new data using GAN, we divided the original data into two subsets: train and test. The test set was kept aside for evaluating the classification models. The train set was used to train GAN, which is a method that can generate synthetic data for the minority class.

Table 3. Data distribution before and after resampling

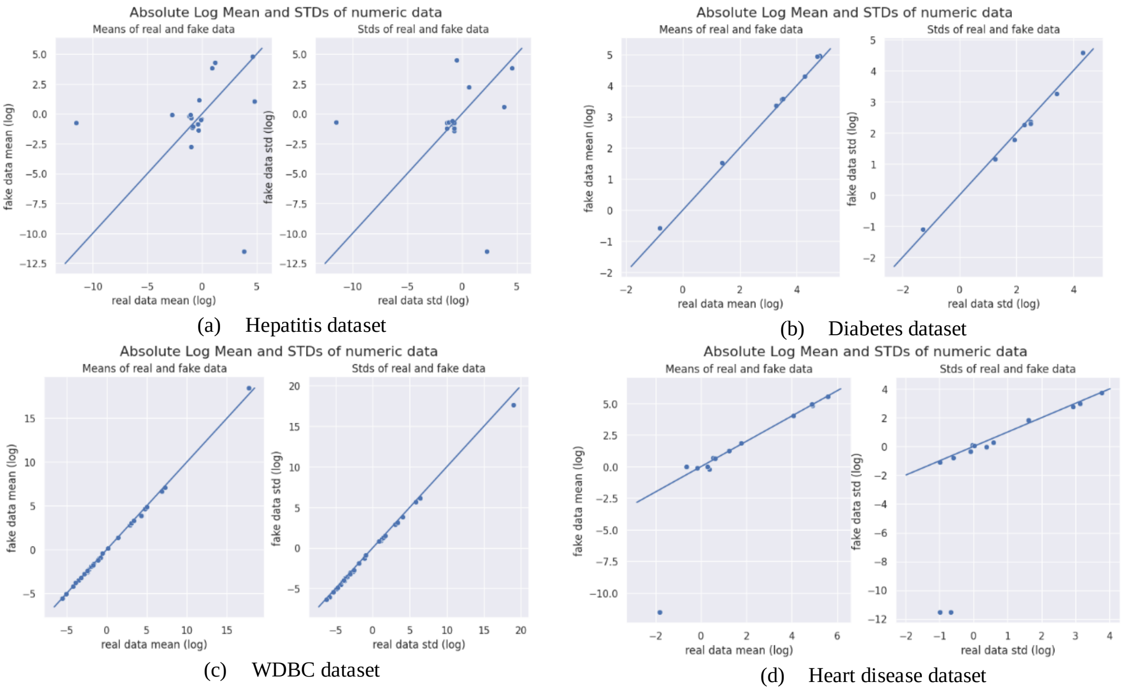
Dataset	Model	Before	After	New Data
Hepatitis	Die=1	29	81	110
	Live=0	110	0	110
Diabetes	Yes=1	241	209	450
	No=0	450	0	450
WDBC	Malignant=1	195	122	317
	Benign=0	317	0	317
Heart disease	Yes=1	109	25	134
	No=0	134	0	134

. To balance the classes in the train set, we needed to generate enough synthetic data for the minority class to match the number of data points in the majority class. To do this, we calculated the ratio of the minority and majority classes in the train set and used it as a guide for how many new data points we should generate using GAN. Table 3 shows the class comparison before and after generating data with GAN.

Fig. 2 shows that the numerical data from GAN is similar to the original data. The log mean and standard deviation values of the new data are close to each other and the line.

Fig. 3 shows that the divergence between real and synthetic data for each column is low in the hepatitis dataset; consider the *steroid* and *fractal_dimension_mean* columns. In the diabetes dataset, 5 of 8 columns have low divergence; in heart disease, 7 of 11 datasets have low divergence. Whereas in the WDBC dataset, 28 of 30 columns have low divergence.

In Fig 4, most columns in each dataset have high cosine similarity values. Between synthetic data and real data, out of 69 columns, only 5 of them, or around 0.07 percent, had a cosine similarity value below 0.5. A low divergence and a high cosine similarity mean the two vectors or distributions are similar.



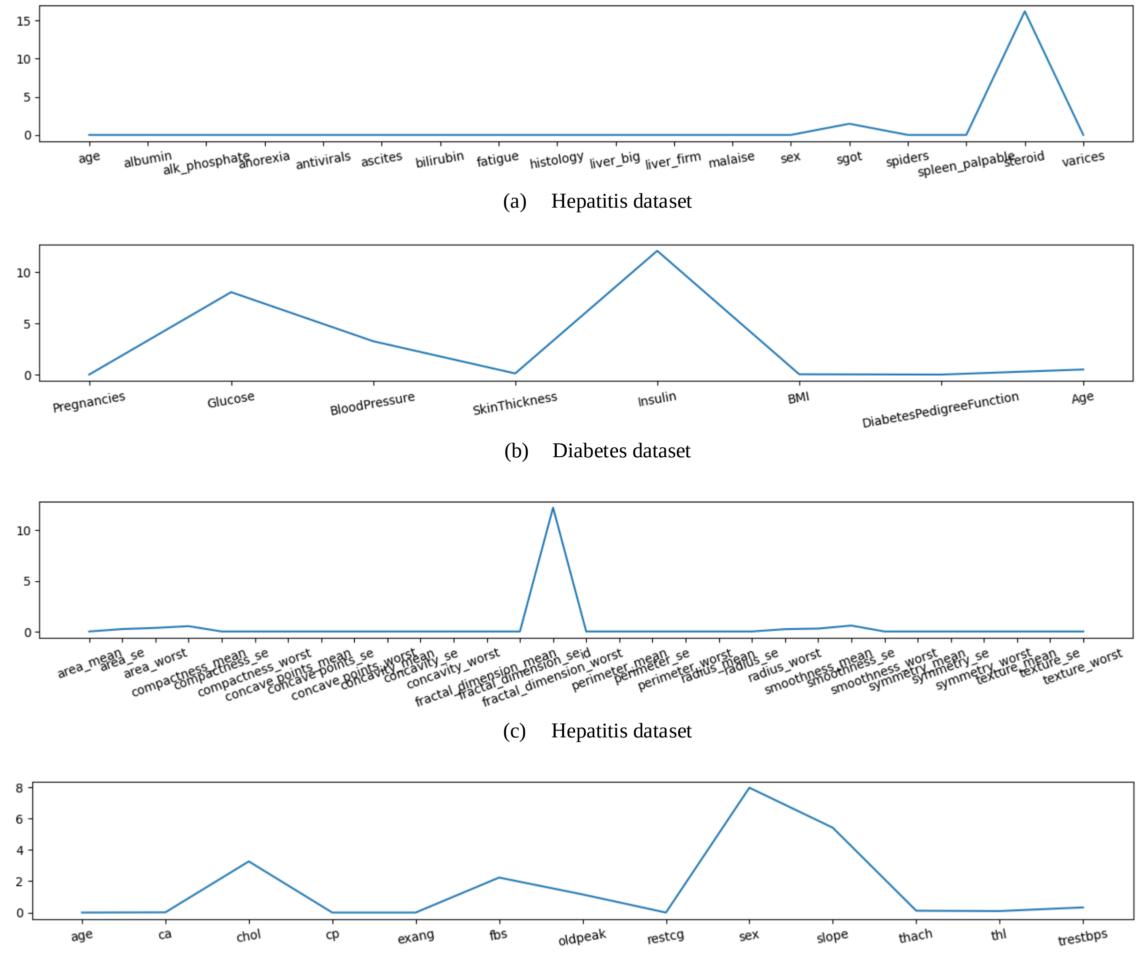
(a) Hepatitis dataset

(b) Diabetes dataset

(c) WDBC dataset

(d) Heart disease dataset

Fig. 2. Log mean values and standard deviation values between real and synthetic data of each database



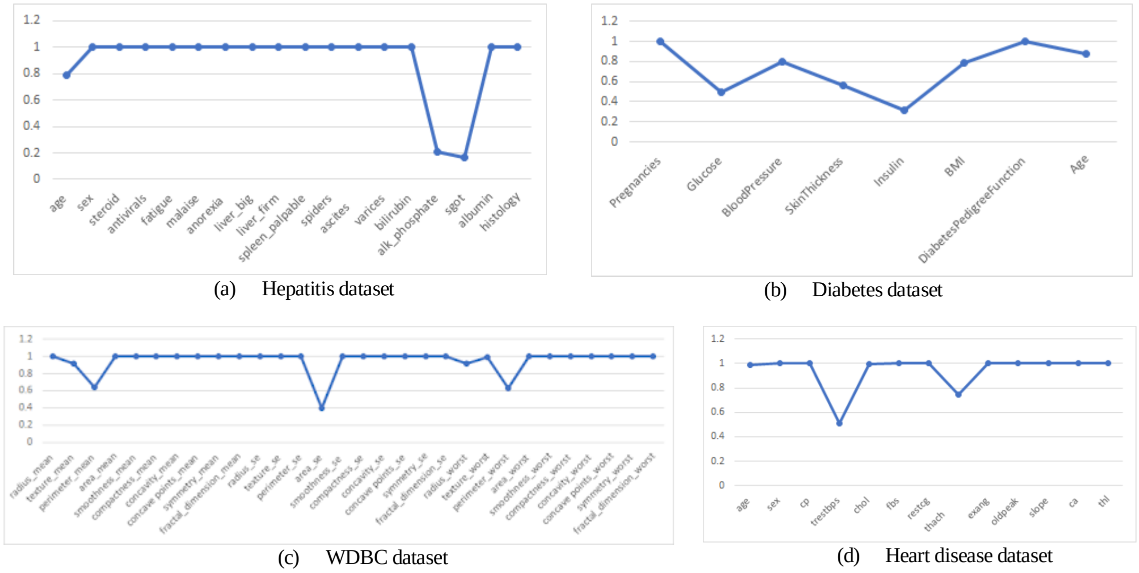
(a) Hepatitis dataset

(b) Diabetes dataset

(c) Hepatitis dataset

(d) Heart disease dataset

Fig. 3. Plot divergence between real and synthetic data



(a) Hepatitis dataset

(b) Diabetes dataset

(c) WDBC dataset

(d) Heart disease dataset

Fig 4. Plot cosine similarity between real and synthetic data

Table 4. Performance comparison classification between SMOTE and GAN

Dataset	Model	Recall		Precision		F1 score		AUC score		FP rate	
		SM	GAN	SM	GAN	SM	GAN	SM	GAN	SM	GAN
Hepatitis	RF	0.93	0.92	0.86	**0.91**	0.86	**0.90**	0.96	**0.98**	0.07	0.06
	LR	0.88	0.89	0.83	0.86	0.84	0.88	0.92	0.93	0.07	0.06
	SVM	0.70	0.65	0.61	0.72	0.67	0.68	0.65	0.80	0.05	**0.04**
	NB	0.94	**0.94**	0.72	0.79	0.83	0.84	0.90	0.91	0.10	0.09
Diabetes	RF	**0.84**	0.81	0.80	**0.82**	0.80	**0.81**	**0.90**	**0.90**	0.15	0.16
	LR	0.71	0.72	0.74	0.73	0.74	0.75	0.81	0.85	0.14	0.14
	SVM	0.67	0.75	0.74	0.76	0.69	0.76	0.82	0.85	**0.12**	**0.12**
	NB	0.63	0.69	0.75	0.76	0.68	0.73	0.81	0.82	0.14	0.14
WDBC	RF	0.97	0.96	**0.98**	**0.98**	0.96	**0.97**	**0.99**	**0.99**	0.03	0.03
	LR	**1.00**	**1.00**	0.50	0.50	0.54	0.67	0.46	0.65	0.12	**0.01**
	SVM	0.19	0.46	0.61	0.81	0.27	0.58	0.46	0.65	0.08	**0.01**
	NB	0.08	0.44	0.78	0.82	0.24	0.57	0.90	0.90	0.06	**0.01**
Heart disease	RF	0.82	0.84	0.84	0.84	0.82	0.83	0.90	**0.92**	**0.13**	**0.13**
	LR	0.83	**0.87**	0.80	0.85	0.82	**0.86**	0.89	0.90	**0.13**	**0.13**
	SVM	0.78	0.75	0.62	0.66	0.71	0.65	0.72	0.74	0.15	0.14
	NB	0.85	0.83	0.83	**0.86**	0.84	0.85	0.90	**0.92**	0.15	0.14
Average		0.74	**0.78**	0.75	**0.79**	0.71	**0.77**	0.81	**0.86**	0.11	**0.09**

The bold shows the highest performance value between SMOTE and GAN among various classifiers.

Table 4 compares the classification model performance using data generated by GAN and data oversampled by SMOTE. GAN and SMOTE can enhance the performance of classification models with imbalanced data, according to the average results of all performance metrics for all datasets and models. Moreover, GAN outperforms SMOTE in Recall, Precision, F1 score, AUC, and FP rate values.

GAN achieves the highest Recall, Precision, F1 score, AUC score and FP-rate in the hepatitis dataset. In the diabetes dataset, SMOTE has the highest Recall, while GAN has the highest Precision and F1 score. GAN and SMOTE have a similar effect on the Random Forest Model classification performance in the WDBC dataset, which is already good in the baseline. However, for other classification models in the WDBC dataset, GAN and SMOTE can significantly improve the classification performance, especially on Recall values.

Classification performance using the GAN and SMOTE methods does not significantly impact the classification performance of the heart disease dataset. The insignificant impact might be because the GAN method only generated 25 data points, which needed to be revised to make a difference. The SMOTE method might also have introduced noise or overfitting to the data, which could affect the classification performance.

Table 5. Performance comparison classification between RSMOTE and GAN

Dataset	Recall		Precision		F1-Score		AUC		FP-Rate	
	RSM	GAN	RSM	GAN	RSM	GAN	RSM	GAN	RSM	GAN
Hepatitis	0.88	**0.94**	**0.93**	0.91	0.89	**0.90**	0.94	**0.98**	0.04	0.04
Diabetes	0.79	**0.81**	0.82	**0.82**	0.80	**0.81**	0.87	**0.90**	0.18	**0.12**
WDBC	0.93	**1.00**	0.97	**0.98**	0.95	**0.97**	0.99	0.99	0.03	**0.01**
Heart Disease	0.84	**0.87**	**0.89**	0.86	0.85	**0.86**	**0.92**	0.90	**0.11**	0.13
Average	0.86	**0.91**	**0.90**	0.89	0.87	**0.88**	0.93	**0.94**	0.10	**0.08**

As shown in 5, GAN outperforms RSMOTE on specific metrics across different datasets. We compare the best values GAN achieves and previous studies for each metric. GAN has a higher Recall, F1 score, AUC score, and FP rate values than RSMOTE for hepatitis and diabetes datasets. GAN improves the Precision, Recall, F1 score, and FP rate values for the WDBC dataset. GAN increases the Recall value and F1 score for the heart disease dataset compared to RSMOTE. Based on the average performance results of all models and datasets, we can see that GAN has higher Recall, F1 score, AUC, and FP rate values than RSMOTE. These metrics indicate that GAN is more effective at identifying the minority class and reducing the false positives and false negatives. However, RSMOTE has a superior precision value than GAN, which means that RSMOTE is more accurate at predicting the true positives.

The limitation of our study is that we only compared our method with two other oversampling methods from recent research. This may limit the generalizability of our results and the evaluation of our method's performance. Future research could compare our method with more oversampling methods and more datasets with various imbalance ration, especially those that have different characteristics from the ones we used

4. Conclusion

In this study, we proposed to explore the ability of GAN to improve classification performance on imbalanced data using medical datasets. From the results of our research, GAN based on oversampling can be used to generate new data to balance the class distributions and improve the classification model's performance. The result of the evaluation model from the four medical datasets based on calculations from Recall, Precision, F1 score, AUC score, and FP rate shows that the application of GAN can outperform SMOTE and RSMOTE methods in several metrics and algorithms.

GAN training has challenges, such as training for stability or balancing it so that the generator and discriminator learn simultaneously. The commonly used implementation is to modify the loss generator to make it more stable. One of the difficulties in using GAN to generate synthetic tabular data is how to train the model to produce realistic and diverse data. Tabular data are complex and heterogeneous because they can have different data types, such as numerical or categorical, and distribution shapes, such as normal, uniform, or skewed. Therefore, GAN needs to learn how to capture the characteristics and relationships of the original tabular data and generate new data that preserve these features. Future research may propose new techniques to improve the quality and diversity of the synthetic tabular data.

References

[1] W. Dari, N. Miranda, S. Informasi, and U. N. Mandiri, “Implementation of c4.5 algorithm in classifying breast cancer based on menopause age,” *J. Pilar Nusa Mandiri*, vol. 17, no. 2, pp. 137–142, 2018.

[2] R. Jain and D. V, “Data mining algorithms in healthcare: an extensive reviewno title,” *Fifth Int. Conf. I-SMAC (IoT Soc. Mobile, Anal. Cloud)*, pp. 728–733, 2021.

[3] L. Chen, G. Xu, Q. Zhang, and X. Zhang, “Learning deep representation of imbalanced scada data for fault detection of wind turbines,” *Meas. J. Int. Meas. Confed.*, vol. 139, pp. 370–379, 2019, doi: 10.1016/j.measurement.2019.03.029.

[4] T. Pan, J. Chen, J. Xie, Z. Zhou, and S. He, “Deep feature generating network: a new method for intelligent fault detection of mechanical systems under class imbalance,” *IEEE Trans. Ind. Informatics*, vol. 17, no. 9, pp. 6282–6293, 2021, doi: 10.1109/TII.2020.3030967.

[5] A. Singh, R. K. Ranjan, and A. Tiwari, “Credit card fraud detection under extreme imbalanced data: a comparative study of data-level algorithms,” *J. Exp. Theor. Artif. Intell.*, vol. 00, no. 00, pp. 1–28, 2021, doi: 10.1080/0952813X.2021.1907795.

[6] R. A. Nugraha, H. F. Pardede, and A. Subekti, “Oversampling based on generative adversarial networks to overcome imbalance data in predicting fraud insurance claim,” *Kuwait J. Sci.*, pp. 1–12, 2022, doi: 10.48129/kjs.splml.19119.

[7] J. M. Johnson and T. M. Khoshgoftaar, “Survey on deep learning with class imbalance,” *J. Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0192-5.

[8] A. Bria, C. Marrocco, and F. Tortorella, “Addressing class imbalance in deep learning for small lesion detection on medical images,” *Comput. Biol. Med.*, vol. 120, no. February, p. 103735, 2020, doi: 10.1016/j.compbiomed.2020.103735.

[9] Z. Qin, Z. Liu, P. Zhu, and Y. Xue, “A gan-based image synthesis method for skin lesion classification,” *Comput. Methods Programs Biomed.*, vol. 195, p. 105568, 2020, doi: 10.1016/j.cmpb.2020.105568.

[10] N. V Chawla, “Data mining for imbalanced datasets: an overview,” in *Data Min. Knowl. Discov. Handb.*, O. Maimon and L. Rokach, Eds. Boston, MA: Springer US, 2010, pp. 875–886. doi: 10.1007/978-0-387-09823-4_45.

[11] F. Thabtah, S. Hammoud, F. Kamalov, and A. Gonsalves, “Data imbalance in classification: experimental evaluation,” *Inf. Sci. (Ny).*, vol. 513, pp. 429–441, 2020, doi: 10.1016/j.ins.2019.11.004.

[12] J. L. Leevy, T. M. Khoshgoftaar, R. A. Bauder, and N. Seliya, “A survey on addressing high-class imbalance in big data,” *J. Big Data*, vol. 5, no. 1, 2018, doi: 10.1186/s40537-018-0151-6.

[13] L. E. Boiko Ferreira, H. Murilo Gomes, A. Bifet, and L. S. Oliveira, “Adaptive random forests with resampling for imbalanced data streams,” *Proc. Int. Jt. Conf. Neural Networks*, vol. 2019-July, no. July, pp. 1–6, 2019, doi: 10.1109/IJCNN.2019.8852027.

[14] R. Ghorbani and R. Ghousi, “Comparing different resampling methods in predicting students' performance using machine learning techniques,” *IEEE Access*, vol. 8, pp. 67899–67911, 2020, doi: 10.1109/ACCESS.2020.2986809.

[15] R. M. Pereira, Y. M. G. Costa, and C. N. Silla, “Toward hierarchical classification of imbalanced data using random resampling algorithms,” *Inf. Sci. (Ny).*, vol. 578, pp. 344–363, 2021, doi: 10.1016/j.ins.2021.07.033.

[16] 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*, pp. 243–248, 2020, doi: 10.1109/ICICS49469.2020.239556.

[17] M. Buda, A. Maki, and M. A. Mazurowski, “A systematic study of the class imbalance problem in convolutional neural networks,” *Neural Networks*, vol. 106, pp. 249–259, 2018, doi: 10.1016/j.neunet.2018.07.011.

[18] S. Fotouhi, S. Asadi, and M. W. Kattan, “A comprehensive data level analysis for cancer diagnosis on imbalanced data,” *J. Biomed. Inform.*, vol. 90, no. October 2017, p. 103089, 2019, doi: 10.1016/j.jbi.2018.12.003.

[19] M. Khushi *et al.*, “A comparative performance analysis of data resampling methods on imbalance medical data,” *IEEE Access*, vol. 9, pp. 109960–109975, 2021, doi: 10.1109/ACCESS.2021.3102399.

[20] R. Geetha, S. Sivasubramanian, M. Kaliappan, S. Vimal, and S. Annamalai, “Cervical cancer

identification with synthetic minority oversampling technique and pca analysis using random forest classifier," *J. Med. Syst.*, vol. 43, no. 9, 2019, doi: 10.1007/s10916-019-1402-6.

[21] X. Li, Y. Wu, and Q. Jia, "Anomaly detection of bolt tightening process based on improved smote," *ACM Int. Conf. Proceeding Ser.*, pp. 9–14, 2020, doi: 10.1145/3449301.3449304.

[22] M. Naseriparsa, A. Al-Shammari, M. Sheng, Y. Zhang, and R. Zhou, "RSMOTE: improving classification performance over imbalanced medical datasets," *Heal. Inf. Sci. Syst.*, vol. 8, no. 1, pp. 1–13, 2020, doi: 10.1007/s13755-020-00112-w.

[23] H. Rashid, M. A. Tanveer, and H. Aqeel Khan, "Skin lesion classification using gan based data augmentation," *Conf. Proc. ... Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. IEEE Eng. Med. Biol. Soc. Annu. Conf.*, vol. 2019, no. 10016, pp. 916–919, 2019, doi: 10.1109/EMBC.2019.8857905.

[24] M. Rezaei, T. Uemura, J. Näppi, H. Yoshida, C. Lippert, and C. Meinel, "Generative synthetic adversarial network for internal bias correction and handling class imbalance problem in medical image diagnosis," *Proc. SPIE 11314, Med. Imaging 2020 Comput. Diagnosis*, vol. 113140E, no. 16 March 2020, 2020, doi: https://doi.org/10.1117/12.2551166.

[25] A. Sharma, P. K. Singh, and R. Chandra, "SMOTified-gan for class imbalanced pattern classification problems," *IEEE Access*, vol. 10, pp. 30655–30665, 2022, doi: 10.1109/ACCESS.2022.3158977.

[26] L. Xu and K. Veeramachaneni, "Synthesizing tabular data using generative adversarial networks," 2018, [Online]. Available: http://arxiv.org/abs/1811.11264

[27] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional gan," *Adv. Neural Inf. Process. Syst.*, vol. 32, no. NeurIPS, 2019.

[28] J. Engelmann and S. Lessmann, "Conditional wasserstein gan-based oversampling of tabular data for imbalanced learning," *Expert Syst. Appl.*, vol. 174, no. September 2020, p. 114582, 2021, doi: 10.1016/j.eswa.2021.114582.

[29] "UCI machine learning repository." https://archive.ics.uci.edu/ml/index.php (accessed Sep. 03, 2022).

[30] M. Abdul Lateh, A. K. Muda, Z. Izzah Mohd Yusof, N. Azilah Muda, and M. Sanusi Azmi, "Handling a small dataset problem in prediction model by employ artificial data generation approach: a review," *J. Phys. Conf. Ser.*, vol. 892, no. 1, 2017, doi: 10.1088/1742-6596/892/1/012016.

[31] V. R. Joseph, "Optimal ratio for data splitting," *Stat. Anal. Data Min.*, vol. 15, no. 4, pp. 531–538, 2022, doi: 10.1002/sam.11583.

[32] L. Feng, H. Wang, B. Jin, H. Li, M. Xue, and L. Wang, "Learning a distance metric by balancing kl-divergence for imbalanced datasets," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 49, no. 12, pp. 2384–2395, 2019, doi: 10.1109/TSMC.2018.2790914.

[33] S. H. Pratiwi, P. Shaniya, G. Jati, and W. Jatmiko, "Improved mask rcnn and cosine similarity using rgbd segmentation for occlusion handling in multi object tracking," *J. Ilmu Komput. dan Inf.*, vol. 16, no. 1, pp. 1–13, 2023, doi: 10.21609/jiki.v16i1.1073.

[34] S. Liu, Y. Wang, J. Zhang, C. Chen, and Y. Xiang, "Addressing the class imbalance problem in twitter spam detection using ensemble learning," *Comput. Secur.*, vol. 69, pp. 35–49, 2017, doi: 10.1016/j.cose.2016.12.004.

[35] W. Feng, W. Huang, and J. Ren, "Class imbalance ensemble learning based on the margin theory," *Appl. Sci.*, vol. 8, no. 5, 2018, doi: 10.3390/app8050815.