Computer Methods and Programs in Biomedicine Update 5 (2024) 100152

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine Update

journal homepage:
www.sciencedirect.com/journal/computer-methods-and-programs-in-biomedicine-update

DiGAN Breakthrough: Advancing diabetic data analysis with innovative GAN-based imbalance correction techniques

Puyang Zhao [a,1], Xinhui Liu [b,1], Zhiyi Yue [a], Qianyu Zhao [c], Xinzhi Liu [d], Yuhui Deng [e], Jingjin Wu [e,*]

[a] *Department of Biostatistics, The University of Texas Health Science Center at Houston, Houston, TX, United States of America*
[b] *Department of Statistics, The London School of Economics and Political Science, London, UK*
[c] *Department of Civil and Environmental Engineering, Duke University, Durham, NC, United States of America*
[d] *School of Physics, Southeast University, Nanjing, China*
[e] *Guangdong Provincial Key Laboratory of Interdisciplinary Research and Application for Data Science, Department of Statistics and Data Science, BNU-HKBU United International College, Zhuhai, China*

ARTICLE INFO

Keywords:
Diabetes
Laplacian score
GAN
Imbalanced classification
Correction techniques

ABSTRACT

In the rapidly evolving field of medical diagnostics, the challenge of imbalanced datasets, particularly in diabetes classification, calls for innovative solutions. The study introduces DiGAN, a groundbreaking approach that leverages the power of Generative Adversarial Networks (GAN) to revolutionize diabetes data analysis. Marking a significant departure from traditional methods, DiGAN applies GANs, typically seen in image processing, to the realm of diabetes data. This novel application is complemented by integrating the unsupervised Laplacian Score for sophisticated feature selection. The pioneering approach not only surpasses the limitations of existing techniques but also sets a new benchmark in classification accuracy with a 90% weighted F1-score, achieving a remarkable improvement of over 20% compared to conventional methods. Additionally, DiGAN demonstrates superior performance over popular SMOTE-based methods in handling extremely imbalanced datasets. This research, focusing on the integrated use of Laplacian Score, GAN, and Random Forest, stands at the forefront of diabetic classification, offering a uniquely effective and innovative solution to the long-standing data imbalance issue in medical diagnostics.

1. Introduction

Diabetes is among the most prevalent chronic diseases worldwide, impacting millions of people each year and exerting a significant financial burden on the economy [1,2]. Characterized by impaired insulin production and utilization, diabetes leads to severe complications like heart disease, vision loss, and kidney disease [3]. Early diagnosis is crucial for effective treatment and lifestyle modifications, making predictive models vital tools for healthcare professionals.

The complexity of diabetes classification is compounded by imbalanced datasets, often leading to biased models. While various methods exist for diabetes classification [4,5], most struggle with dataset imbalance, a crucial issue the research addresses.

This paper introduces “DiGAN”, a novel machine learning framework that utilizes a Generative Adversarial Network (GAN) [6] approach for imbalanced diabetes classification. The method innovatively applies GAN, typically used in image processing, to diabetes data, demonstrating its efficacy in balancing datasets and improving classification accuracy. Incorporating the Laplacian Score for feature selection further refines the model’s performance.

In pursuing innovative solutions to dataset imbalance, this paper previously explored SMOTE-based approaches in the domain of Bitcoin addresses classification, involving a three-class scenario described in the earlier work [7]. The success of SMOTE-based methods in that context was encouraging, yielding satisfactory results and highlighting the potential of advanced data augmentation techniques in handling imbalanced datasets.

This prior experience prompted the evaluation of the performance of similar approaches in the context of diabetes classification. The current study extends this exploration by comparing the effectiveness of SMOTE-based methods with a novel GAN approach, specifically tailored for diabetes datasets. This comparison aims to validate the

[*] Corresponding author.
E-mail addresses: Puyang.Zhao@uth.tmc.edu (P. Zhao), x.liu146@lse.ac.uk (X. Liu), Zhiyi.Yue@uth.tmc.edu (Z. Yue), qianyu.zhao@alumni.duke.edu (Q. Zhao), 213203624@seu.edu.cn (X. Liu), ivandeng@uic.edu.cn (Y. Deng), jj.wu@ieee.org (J. Wu).
[1] These authors contributed equally to this work

https://doi.org/10.1016/j.cmpbup.2024.100152
Received 28 December 2023; Received in revised form 25 February 2024; Accepted 1 April 2024
Available online 10 April 2024

utility of GAN in medical data analysis and establish a benchmark against proven methods in data imbalance correction.

The goal is to bridge the gap in diabetes classification models by addressing the imbalance in datasets, often overlooked. This approach is novel and demonstrates significant improvement over existing methods, particularly in handling imbalanced data, a common challenge in medical datasets.

The remainder of the paper is organized as follows. Section 2 provides an overview of existing related works. Section 3 describes the dataset and explains the methodology of the selected approaches. Section 4 compares the effectiveness of various strategies in correctly detecting diabetes. Finally, Section 5 concludes the paper and provides potential future research directions.

2. Related works

When collecting diabetes-related data, various medical organizations list dozens of indicators; some help diagnose diabetes, and some are useless or misleading. In the data analysis stage, too many features will lead to a curse of dimensionality, which will reduce the classifier's performance and significantly slow down the computation speed. Therefore, feature selection is essential. X. He et al. [8] introduced a novel feature selection algorithm called Laplacian score. Compared with data variance (unsupervised) and Fisher score (supervised) on two datasets, Laplacian score performed strong effectiveness and efficiency through experimental results. This proposed algorithm is an unsupervised filter method, while almost all the existing filter approaches are supervised. It brings us to a new way of conducting feature selection.

Mirza et al. [9] applied SMOTE to solve the imbalance problem in the data preprocessing step and then selected the best classifier for a balanced dataset to predict diabetes, concluding that Decision tree performed the best with an accuracy of 94.7013%. SMOTE is a powerful method to address imbalance issues. The intention is to apply GAN [10] to the diabetes dataset and compare the results with SMOTE-based approaches to evaluate the performance of GAN.

The most popular classifiers in existing studies are Decision tree, Support vector machine, Naive Bayes, etc. Deepti and Dilip Singh [11] applied these algorithms to predict diabetes and concluded that Naive Bayes outperformed the best with the highest accuracy of 76.30%. In addition to these algorithms, Kayaer and Yıldırım [12] proposed a diabetes diagnosis system using diverse Artificial Neural Networks, Radial Basis Function and General Regression Neural Network. Compared to Multi-Layer Perceptron and RBF, GRNN performed best, achieving 80.21%

The proposed framework first applies GAN to address a highly imbalanced diabetes dataset and then utilizes Lap score for feature selection. Detailed explanations for the new techniques will be provided in later sections.

3. Methodology

3.1. Data overview

The Behavioral Risk Factor Surveillance System (BRFSS) is a health-related telephone survey collected annually by the CDC [13]. The dataset, derived from a survey, comprises 49,606 observations and 22 variables, encompassing both direct questions posed to participants and calculated variables based on individual responses. The variables are characterized as follows.

- **Diabetes_012.** 0 indicates no diabetes, 1 indicates prediabetes, and 2 indicates diabetes.
- **HighBP.** BP is short for blood pressure. 0 indicates the individual does not have high BP, and 1 indicates high BP.
- **HighChol.** 0 indicates the individual does not have high cholesterol, and 1 indicates high cholesterol.
- **CholCheck.** 0 indicates the individual has not had a cholesterol check in 5 years, and 1 indicates a cholesterol check within five years.
- **BMI.** Body mass index is defined as weight divided by height squared in kg/m^2.
- **Smoker.** 0 indicates the individual has never smoked at least 100 cigarettes in his entire life, and 1 indicates smoking.
- **Stroke.** 0 indicates the individual has never been told that he had a stroke, and 1 indicates having a stroke.
- **HeartDiseaseorAttack.** 0 indicates the individual does not have coronary heart disease (CHD) or myocardial infarction (MI) and 1 indicates the individual has CHD or MI.
- **PhysActivity.** 0 indicates the individual has not done physical activity (not including a job) in the past 30 days, and 1 indicates doing physical activity.
- **Fruit.** 0 indicates the individual does not consume fruit daily, and 1 indicates fruit consumption.
- **Veggies.** 0 indicates the individual does not consume vegetables daily, and 1 indicates vegetable consumption.
- **HvyAlcoholConsump.** The term ‘Heavy drinkers’ is defined as adult men with more than 14 drinks weekly and women with more than seven drinks weekly. 0 indicates the individual is not a heavy drinker, and 1 indicates a heavy drinker.
- **AnyHealthcare.** 0 indicates the individual has no health care coverage, and 1 indicates one or more health care coverage, including health insurance and prepaid plans such as HMO.
- **NoDocbcCost.** 0 indicates that in the past 12 months, he needed to see a doctor but could not because the cost has not happened to the individual. 1 indicates this situation has happened to the individual.
- **GenHlth.** General health status for the individual. 1 represents excellent, 2 represents very good, 3 represents good, 4 represents fair, and 5 represents poor.
- **MentHlth.** Mental health status for the individual. How many days out of the past 30 days did the individual feel stressed, depressed or have emotional problems?
- **PhysHlth.** Physical health status for the individual. How many days out of the past 30 days did the individual have physical illness and injury?
- **DiffWalk.** 0 indicates the individual does not have serious difficulty walking or climbing stairs, and 1 indicates serious difficulty walking or climbing stairs.
- **Sex.** 0 indicates the individual is female, and 1 indicates a male.
- **Age.** 13-level age categories. 1 represents age 18 to 24, 2 represents age 25 to 29, 3 represents age 30 to 34, 4 represents age 35 to 39, 5 represents age 40 to 44, 6 represents age 45 to 49, 7 represents age 50 to 54, 8 represents age 55 to 59, 9 represents age 60 to 64, 10 represents age 65 to 69, 11 represents age 70 to 74, 12 represents age 75 to 79, and 13 represents age 80 or older.
- **Education.** Education level of the individual, which is a scale from 1 to 6. 1 or 2 or 3 represents the individual who did not graduate from high school. 4 represents the individual who graduated from high school. 5 represents the individual who attended college or technical school. 6 represents the individual who graduated from college or technical school.
- **Income.** Income level of the individual, which is a scale from 1 to 8. 1 or 2 represents the individual's income is less than $15,000. 3 or 4 represents the individual's income is $15,000 to less than $25,000. 5 represents the individual's income is $25,000 to less than $35,000. 6 represents the individual's income is $35,000 to less than $50,000. 7 or 8 represents the individual's income is $50,000 or more.

3.2. Feature selection

There are 21 features in the dataset, leading to a longer computation time and more memory to store data. In addition, too many features will affect the result of classification, so it is necessary to make the feature selection, and Laplacian score is the chosen technique.

The Laplacian score method evaluates the features within the training set by assigning a score to each. Then it takes the lowest k features as the ultimate feature subset, a standard filter method, which enables the identification of the most significant indicators for diabetes classification.

The Lap score of the r_{th} feature is represented by Lr. The dataset contains 21 features, so r can take on values from 1 to 21. The i_{th} sample of the r_{th} feature is represented by the variable f_{ri}. There are m samples, so i can take values from 1 to m.

The algorithm is divided into four steps, which can be stated as follows [8].

- **Construction of connected graphs:** Construct a nearest neighbor graph comprising m nodes, denoted as x_i representing the i_{th} node. An edge can link nodes i and j if they are among the top k nearest neighbors of each other. Alternatively, when label details are available, nodes sharing the same label can be connected by an edge. Consequently, there exist two approaches to establishing a connection between two nodes.

  Method 1:
  x_i and x_j can be connected with an edge if x_i and x_j are among each other's k nearest neighbors.

  Method 2:
  When label information is available, it is possible to establish an edge between two nodes that share the same label, signifying their connection.
  Graph A has different representation methods for learning, where 1 indicates that the two nodes are connected.
  For unsupervised learning:

$$A_{m\times m} : (A_{ij}) = \begin{cases} 1, & \text{if } x_i \text{ close to } x_j \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

  For supervised learning:

$$A_{m\times m} : (A_{ij}) = \begin{cases} 1, & \text{if } L_i = L_j \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

- **Weight matrix:** If nodes i and j are connected in the graph, the weight of the edge between them can be set to $S_{ij} = e^{-\frac{\|x_i - x_j\|^2}{t}}$, where t is a suitable constant. If nodes i and j are not connected, then $S_{ij} = 0$. The weight matrix S reflects the local structure of the data space.
  If the value of the node pair equals 1 in graph A, a weight S_{ij} is assigned to it.
- **Graph Laplacian:**
  The vector $f_r = [f_{r1}, f_{r2}, \dots, f_{rm}]^T$ is defined, where f_{ri} denotes the value of the rth feature for the ith data point. The diagonal matrix B is specified as $diag(S\mathbf{1})$, where S represents the weight matrix of the graph. Thus, the graph Laplacian matrix L is established as $L = B - S$ [14]. Let

$$\tilde{f}_r = f_r - \frac{{f_r}^T B\mathbf{1}}{\mathbf{1}^T B\mathbf{1}}\mathbf{1} \tag{3}$$

- **Laplacian Score:** Compute the Lap score of the r_{th} feature as follows.

$$L_r = \frac{\tilde{f}_r^{\,T} L \tilde{f}_r}{\tilde{f}_r^{\,T} B \tilde{f}_r} \tag{4}$$

Explanation of calculating Laplacian score: A weighted graph is constructed to evaluate the feature importance. The similarity between the i_{th} and j_{th} nodes is measured by S_{ij}. In this context, the feature importance can be considered as the extent to which it preserves the graph's structure. In other words, a "good" feature is one where two points are close to each other only if they are connected by an edge in the graph. The subsequent formula is employed to identify good features, aiming to minimize its value.

$$L_r = \frac{\sum_{ij}(f_{ri} - f_{rj})^2 S_{ij}}{Var(f_r)} \tag{5}$$

where L_r is the Lap score of the r_{th} feature, $f_{ri} - f_{rj}$ is the difference between the i_{th} sample and the j_{th} sample on the r_{th} feature, and $Var(f_r)$ is the estimated variance of the r_{th} feature.

Features that preserve the pre-defined graph structure can be selected by minimizing the function $\sum_{ij}(f_{ri} - f_{rj})^2 S_{ij}$. For a good feature, the larger S_{ij} is, the smaller $f_{ri} - f_{rj}$ will be, and therefore the Laplacian score will tend to be small.

In addition, (5) can be converted to (4) with algebra steps, which can be found in [8].

Table 1
The proportion of types.

Type	0	1	2
The number	33 703	4631	11 272
Proportion	0.68	0.093	0.227

3.3. *Existing imbalance disposal*

As shown in Table 1, the dataset is imbalanced, with type 1 data accounting for only a tiny proportion. Therefore, resampling methods are considered to obtain a balanced sample distribution by changing the original imbalanced sample set and learning an appropriate model.

Three most recently proposed resampling approaches: SMOTE [15], SMOTE with ENN (SMOTE-ENN) and SMOTE with Tomek links (SMOTE-Tomek) [16] are included to deal with the imbalance problem.

3.3.1. *SMOTE*

SMOTE is an oversampling technique introduced through a three-step algorithmic process.

- **Step 1:** Assuming x is a sample from the minority class, the first step is to calculate the Euclidean distance from it to all other samples in the class. Then, from the k smallest distances, the k nearest neighbors are identified.
- **Step 2:** Set a sampling ratio according to the proportions of each type in the imbalanced dataset. For x, N samples (sampling ratio) are randomly selected from their k nearest neighbors.
- **Step 3:** For each nearest neighbor o, a new sample is created according to the formula:

$$o(new) = o + rand(0, 1) \times (x - o) \tag{6}$$

3.3.2. *SMOTE-Tomek*

Over-sampling can solve the imbalance issue in a dataset with skewed class distributions, but other problems often remain. The definition of class clusters may be unclear because the examples from the majority class may take up the space for those of the minority class. Using a classifier in this situation may cause overfitting. To solve this problem, Tomek links are applied to do the data cleaning, which removes examples from both classes that form Tomek links. Then, a balanced dataset with well-defined class clusters can be obtained.

3.3.3. *SMOTE-ENN*

SMOTE-ENN is similar to SMOTE-Tomek in its motivation, but it takes a more thorough approach to data cleaning by removing more examples. This method uses the class labels of a data point's three nearest neighbors to determine whether it should be removed from the training set. If the class label of the data point differs from that of at least two of its three nearest neighbors, it is removed.

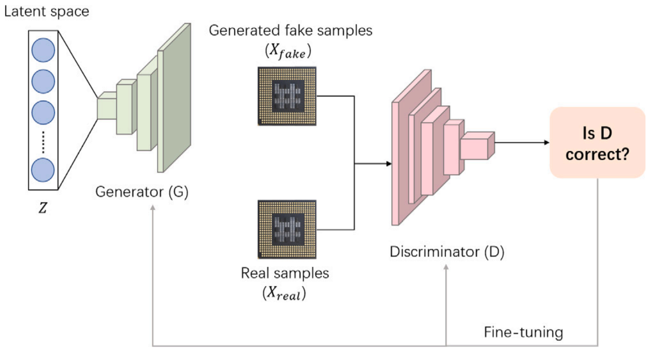


Fig. 1. Flow diagram for GAN.

3.4. *Proposed imbalance disposal: GAN*

The crux of DiGAN's innovation lies in its specialized Generative Adversarial Network (GAN) [6] architecture, designed to address the class imbalance by generating synthetic data statistically similar to the original minority class data in the diabetes dataset. The architecture consists of two main components: the generator (G) and the discriminator (D), which are trained concurrently through adversarial processes.

3.4.1. *Model framework*

The GAN model comprises a generator (G) and a discriminator (D), both multi-layer perceptron structures. The specific number of layers and neurons in each layer can be adjusted to fit the specific needs of the problem (see Fig. 1).

3.4.2. *Generator (G)*

The generator is a neural network that maps vectors from a latent space to the data space, generally conforming to a Gaussian distribution. The purpose of the generator is to learn the distribution characteristics of the diabetes data set and then to generate new data points, which endeavors to produce new data that are nearly indistinguishable from the actual data set.

- An introductory layer that accepts a noise vector z with a dimensionality of 100, sampled from a standard normal distribution.
- A sequence of three fully connected hidden layers consisting of 256, 512, and 1024 units designed to encapsulate the data's complexity. A LeakyReLU activation function with an alpha value set to 0.2 is applied at each layer, permitting a minor gradient when the unit is inactive, thereby facilitating a healthier gradient flow during the training phase.
- A dropout strategy is implemented after each hidden layer with a rate set at 0.3 to mitigate overfitting risks.
- The concluding layer is equipped with units equal to the number of features in the diabetes dataset, utilizing a hyperbolic tangent (tanh) activation function to output the synthetic data points.

The architecture of the generator is deliberated to be sufficiently profound to capture the intricate patterns inherent in the minority class, allowing the generation of synthetic data with high fidelity.

3.4.3. *Discriminator (D)*

The discriminator operates as a neural network that attempts to differentiate between the real diabetes data and those produced by the generator. The configuration of the discriminator in this study includes:

- An input layer is designed to receive data points from the dataset, matching the dimensionality of the output from the generator.
- A mirrored generator structure comprising three fully connected hidden layers with a descending order of units from 1024 to 512 to 256. Consistency in the activation function is maintained by applying LeakyReLU and the same alpha value.
- A solitary neuron at the output layer with a sigmoid activation function tasked with predicting the probability that a data point originates from the actual dataset rather than being generated.

The discriminator's goal is to maximize the likelihood of accurately categorizing both real and synthetic data points.

3.4.4. *Training procedure*

The training of DiGAN constitutes an adversarial contest where the generator and the discriminator are optimized in a synchronized manner. The generator's mission is to deceive the discriminator into categorizing the synthetic data as real, while the discriminator endeavors to differentiate precisely between real and fake data. This process incorporates:

- Generating synthetic data points from input noise vectors for each data batch by the generator.
- The discriminator evaluates the real data from the dataset alongside the new synthetic data.
- Both networks undergo updates based on their respective loss functions, with the gradients of the generator being fine-tuned to increase the likelihood of the discriminator misclassifying the synthetic data as real in subsequent iterations.

3.4.5. *Optimization and loss functions*

The loss function is the expected sum of the probability distributions of real and fake samples, where P_{data} is the probability distribution of real samples X_{real} and P_G is the probability distribution of fake samples X_{fake} generated by G. This loss function is used to measure the quality of the fake samples generated by G and guide the training of the GAN network.

$$V(G,D) = E_{x\sim P_{data}}[\log D(x)] + E_{x\sim P_G}[\log(1-D(x))] \tag{7}$$

For D, its purpose is to make the outcome result $log D(x \sim P_{data})$ of the sample in P_{data} as large as possible and make the outcome result $log(1 - D(x \sim P_G))$ of the sample in P_G as small as possible, leading to a greater value of $V(G,D)$.

For G, its purpose is to generate X_{fake} with noise z to assign a large value to D, making the value of $log(1 - D(x \sim P_G))$ as small as possible, and therefore shrink the value of $V(G,D)$.

In summary,

$$V^*(G,D) = \arg\min_G \max_D V(G,D) \tag{8}$$

The Adam optimizer's hyperparameters, such as the learning rate and beta coefficients, are chosen based on preliminary tests to ensure the adversarial networks' stable training and convergence.

The ultimate goal of the GAN network is to produce a G that generates high-quality samples that are indistinguishable from real samples by D. This is achieved when D produces a value of 0.5 for fake samples generated by G, indicating that it can no longer tell the difference between real and fake samples. In other words, G has successfully deceived D and can generate high-quality fake samples.

After solving the imbalance issue, the next step is to classify the diabetics. The integration of the above four imbalance processing methods (GAN and SMOTE-based approaches) with various classification models will be undertaken (to be described in the following section).

Algorithm 1 *Generative adversarial network*

```
The batch size and the number of steps applied to the discriminator (D) are assumed to be $m$ and $k$ in the experiments, where $m = 105$ and $k = 1$.
Denote $D(x)$ and $G(z)$ are functions of the discriminator (D) and the generator (G), respectively.
for the number of epochs do
    for $k$ steps do
        Generate $m$ noise samples $z_1, z_2, ..., z_m$ from prior distribution $P_G(z)$
        Generate $m$ real samples $x_1, x_2, ..., x_m$ from real distribution $P_{data}(x)$
        Calculate the loss of D:
        $D_{loss} = \frac{1}{m} \sum_{i=1}^{m} [\log D(x_i) + \log(1 - D(G(z_i)))]$    (9)
        Update the parameters in $D(x)$ by gradient descent.
    end for
    Generate $m$ noise samples $z_1, z_2, ..., z_m$ from prior distribution $P_G(z)$
    Calculate the loss of G:
    $G_{loss} = \frac{1}{m} \sum_{i=1}^{m} \log(1 - D(G(z_i)))$    (10)
    Update the parameters in $G(z)$ by gradient descent
end for
```

3.5. Classification models

This section will introduce some classic machine learning models: K-nearest Neighbor, Random forest and Extreme gradient boosting to classify diabetics.

k-NN can be applied in classification that a new label is assigned for the unlabeled test data based on the majority vote, which is the class represented mainly by its k nearest neighbor points [17]. The value of k is chosen from cross-validation, leading to the minimum error rate in a defined range. Euclidian, Manhattan, and Hamming distances are three primarily used methods for calculating distances between the new test data and surrounding training points.

RF is a classifier that uses multiple trees to train and predict, and there is no correlation between the trees [18]. After the forest is generated, a new sample is classified separately by each decision tree. The category selected by the most decision trees is the ultimate class to which the piece belongs. Random forest reduces the risk of overfitting and works well with high-dimensional data. After training, the significance of features can be judged from the values of Gini importance and mean decrease in impurity (MDI).

XGB is an implementation of the Gradient Boosting framework that uses parallel tree boosting to solve data science problems efficiently and accurately [19]. It is often referred to as GBDT or GBM. The Boosting algorithm aims to integrate many weak classifiers to form a robust classifier. As one of the boosting algorithms, XGBoost integrates many tree models into a robust classifier, and the trees are included in sequential form. All independent variables are given weights and fed into the decision tree to obtain the predicted results. For mispredicted variables, their weights are increased to make it easier to spot errors. Integrating these classifiers significantly improves the algorithm's accuracy and is widely used to solve regression and classification problems.

3.6. Classification evaluation

This section introduces a model selection criterion called the weighted F1-score, which is an appropriate method for a 3-class classification problem.

Table 2
The balanced training set.

Method	# type 0	# type 1	# type 2
GAN	26962 (33.3%)	26962 (33.3%)	26962 (33.3%)
SMOTE	26947 (33.3%)	26947 (33.3%)	26947 (33.3%)
SMOTE-ENN	11996 (31.99%)	13849 (36.93%)	11655 (31.08%)
SMOTE-Tomek	26132 (33.09%)	26592 (33.67%)	26259 (33.24%)

Table 3
Selected features.

Feature	Selection	Feature	Selection
HighBP	Selected	AnyHealthcare	×
HighChol	Selected	NoDocbcCost	Selected
CholCheck	×	GenHlth	Selected
BMI	Selected	MentHlth	Selected
Smoker	Selected	PhysHlth	Selected
Stroke	Selected	DiffWalk	Selected
HeartDiseaseorAttack	Selected	Sex	Selected
PhysActivity	Selected	Age	×
Fruits	×	Education	Selected
Veggies	×	Income	×
HvyAlcoholConsump	Selected		

First, the process is treated as three 2-class classification problems: type 0/others, type 1/others, and type 2/others, to obtain three preclass F1-scores. Then, multiply them by the proportion of each category to get the weighted F1 score.

$$\text{Weighted F1-score} = F_0 \times p_0 + F_1 \times p_1 + F_2 \times p_2 \tag{11}$$

4. Results

This section presents the application of GAN to address imbalance issues in diabetes classification and the comparison with other algorithms, as well as the features selected using the Laplacian score.

The dataset is divided into a training set and a test set with proportions of 80% and 20%, respectively. There are 39,685 observations in the training set and 9,921 in the test set.

4.1. Balanced dataset

Imbalance processing is conducted on the training set using GAN and SMOTE-based approaches. The type 0 data with the largest proportion is selected as the resampling standard to prevent information loss. Consequently, only the type 1 and 2 data are resampled. In the balanced dataset, the number of each type is presented in Table 2.

4.2. Feature selection

The Lap score of each feature in the training set is calculated, and the corresponding 15 features based on the lowest 15 scores are identified. The results are presented in Table 3.

Through extensive experiments, it has been observed that selecting the top 15 features yields the best results. If the number of features is less than 15, the F1-scores will decrease, while when the number is 15, the F1-scores will increase by about 0.1%. Moreover, when the number exceeds 15, the F1-scores almost stay the same.

It is worth noting that, in addition to medical indicators, social factors such as education influence diabetes are selected.

4.3. Classification results

Now, numerical results are presented to demonstrate the performance comparison of different machine learning methods. Fig. 2 below compares weighted F1-scores between machine learning algorithms. As shown in Fig. 2, the F1-scores with SMOTE-based approaches are all obviously worse than GAN. Among them, RF with GAN has the

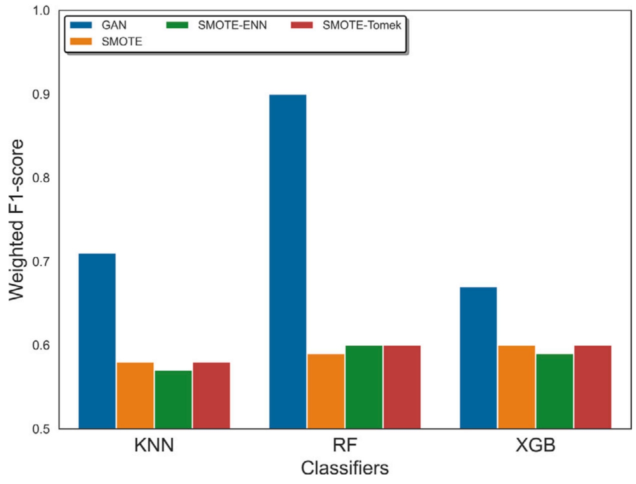


Fig. 2. Classification results.

best performance, reaching an F1-score of 90%, which is well ahead of others.

The excellent performance of GAN is closely related to how it generates data, which absorbs more original information. The generator (G) captures the distribution of sample data and is a binary classifier to determine whether the input is real data or generated samples. By learning the characteristics of real data, the distribution of sample data can be estimated, and then new samples similar to training samples can be generated. The parameters of G are much less than the amount of training data, so G can discover and internalize the nature of the data to generate it better.

In the next section, a visual approach will explain why GAN achieves better results in dealing with imbalance issues.

	Setup parameters
KNN	n_neighbors = 10, leaf_size = 30, weights = uniform, metric = minkowski
RF	n_estimators = 100, min_samples_split = 2, criterion = Gini, min_weight_fraction_leaf = 0, max_features = None, bootstrap = True
XGB	objective = binary : logistic, random_state = 2,max_depth=8, n_estimators = 50

4.4. *Visualization of imbalance processing*

The generated data is plotted versus the original data to explore why GAN performs better than SMOTE-based approaches in dealing with imbalance issues. The comparison of data distributions generated by different techniques helps in identifying the reasons behind the observed performance differences.

4.4.1. *Dimension reduction*

With 15 features influencing diabetes, dimensionality reduction is necessary for visualization. Therefore, Principal Component Analysis (PCA) is conducted on the balanced training set to extract the principal components first. The cumulative contributions of principal components are presented in Table 4.

Table 4 shows that the cumulative contributions of the first two principal components exceed 75% for each imbalance technique. Therefore, the first two principal components are utilized to draw 2-dimensional graphs for visualization.

Table 4
Cumulative contributions of different imbalance techniques.

PCs	PC1	PC2	PC3	PC4	PC5	PC6	...
GAN	0.524	0.768	0.946	0.988	0.993	1	1
SMOTE	0.516	0.753	0.932	0.985	0.991	1	1
SMOTE-ENN	0.519	0.764	0.95	0.99	0.994	1	1
SMOTE-Tomek	0.522	0.762	0.947	0.989	0.994	1	1

4.4.2. *Visualization*

Once the principal components are selected, the generated and original data are plotted to observe distribution differences.

From Fig. 3, it is evident that SMOTE effectively captures the boundary, encompassing both type 1 and type 2 points. The distributions of the generated points almost coincide with the original points, with a few points outside the boundary.

Fig. 4 shows that the overall shapes of the points generated by SMOTE-ENN are similar to the original points. However, their boundaries do not overlap. The boundaries of the generated points shift slightly to the left. There are also a few points beyond the boundary.

From Fig. 5, it is observed that the distributions of the generated points by SMOTE-Tomek are remarkably similar to SMOTE. This similarity is one of the reasons why these two imbalance processing methods achieve comparable weighted F1-scores. Most of the generated points are within the boundaries of the original points.

GAN will do the factorization when augmenting data, resulting in the scale being different from the original data. So, the generated points are drawn separately from the original ones, their distributions observed, and compared with SMOTE-based approaches.

From Figs. 6 and 7, it can be observed that the distributions of the points generated by GAN and the original points appear different. Still, upon drawing the outlines of the generated points, they are found to be highly similar to the outlines of the original points. It is also how GAN generates data, which can estimate the data distribution by learning noise.

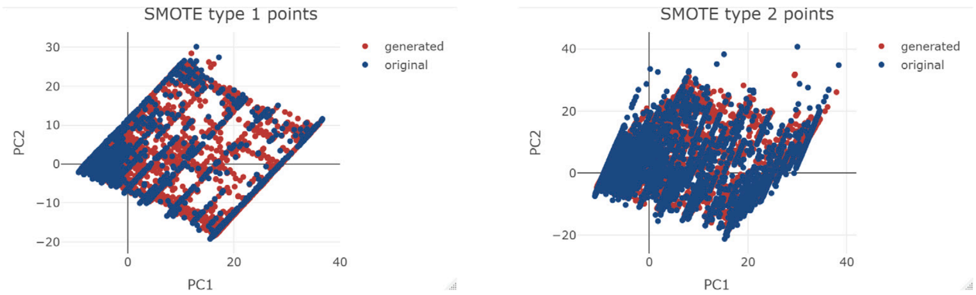
By contrast with the visualization, the distributions of the points generated by GAN are pretty different from SMOTE-based approaches. Moreover, it is speculated that GAN achieves better results than SMOTE-based techniques.

5. Conclusion

This study marked a significant stride in diabetes classification by ingeniously integrating the Laplacian score, Generative Adversarial Network (GAN), and Random Forest (RF). The approach's novelty lied in utilizing GAN for data augmentation, effectively countering the challenges posed by imbalanced datasets. The result showed a notable enhancement in model performance, evidenced by a 20% increase in the F1-score compared to traditional SMOTE methods, thus demonstrating a more reliable solution for diabetes diagnosis and management.

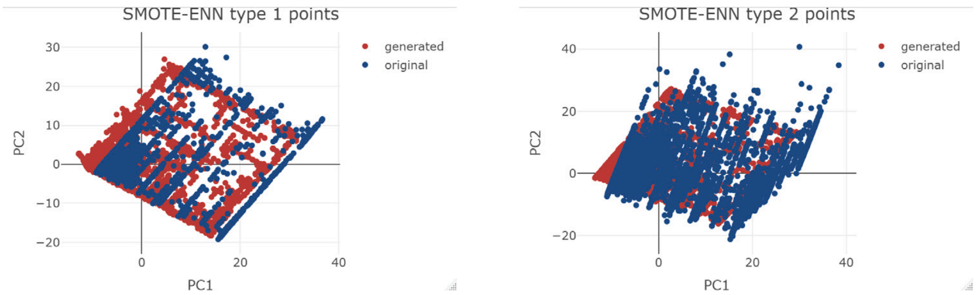
This work also addressed a critical gap in diabetes classification and paved the way for future research. It is envisioned that the principles of the method could be applied to other medical conditions where data imbalance poses a significant challenge. The next phase of research will aim to refine the theoretical foundations of the approach and improve model interpretability, which is essential for real-world applications.

The method's ingenuity can be extended beyond mere performance metrics. By adapting GAN, traditionally linked with image data, to a multi-featured diabetes dataset, its versatility and untapped potential in diverse applications are showcased. As a result of this successful adaptation, GAN is becoming more applicable to complex, multidimensional datasets typical of medical research and beyond.



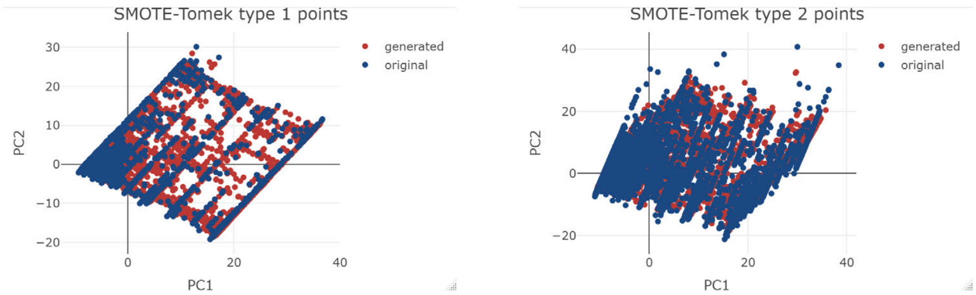
(a) SMOTE type 1 (b) SMOTE type 2

Fig. 3. Visualization of SMOTE.



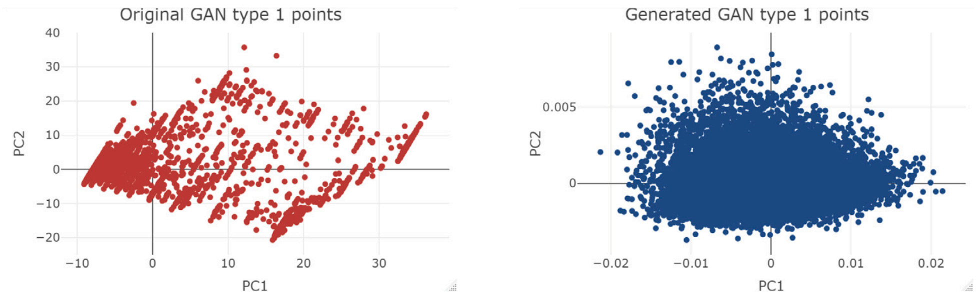
(a) SMOTE-ENN type 1 (b) SMOTE-ENN type 2

Fig. 4. Visualization of SMOTE-ENN.



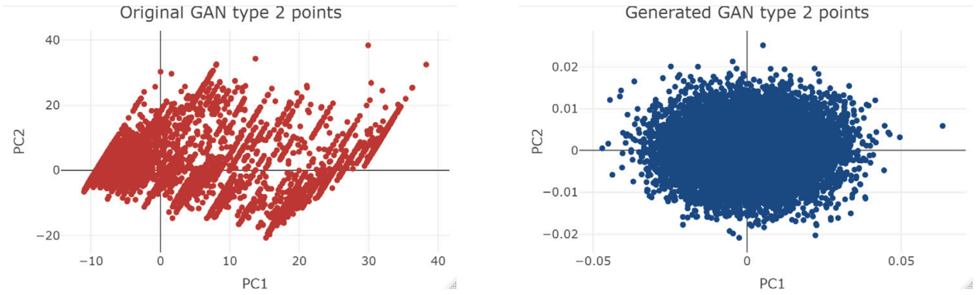
(a) SMOTE-Tomek type 1 (b) SMOTE-Tomek type 2

Fig. 5. Visualization of SMOTE-Tomek.



(a) Original GAN type 1 (b) Generated GAN type 1

Fig. 6. Visualization of GAN type 1 points.



(a) Original GAN type 2 (b) Generated GAN type 2

Fig. 7. Visualization of GAN type 2 points.

CRediT authorship contribution statement

Puyang Zhao: Writing – review & editing, Writing – original draft, Formal analysis, Methodology, Investigation, Conceptualization. **Xinhui Liu:** Writing – review & editing, Writing – original draft, Visualization, Software, Validation, Conceptualization. **Zhiyi Yue:** Formal Analysis, Writing – review & editing. **Qianyu Zhao:** Methodology, Software. **Xinzhi Liu:** Visualization, Validation. **Yuhui Deng:** Resources, Funding acquisition. **Jingjin Wu:** Writing – review & editing, Supervision, Project administration.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Jingjin Wu reports financial support was provided by Guangdong University Innovation and Enhancement Programme Funds.

Funding source

This research was funded by the Guangdong Provincial Key Laboratory of Interdisciplinary Research and Application for Data Science, China, BNU-HKBU United International College, project code 2022B1212010006 and UIC, China research grant R0400001-22, featured Innovation Project, Guangdong University Innovation and Enhancement Programme Funds, China 2018KTSCX278, R5201910 and UIC Research Grants, China R201809.

References

[1] American Diabetes Association, Economic costs of diabetes in the US in 2017, Diabetes Care 41 (5) (2018) 917–928.

[2] D.N. Koye, D.J. Magliano, R.G. Nelson, M.E. Pavkov, The global epidemiology of diabetes and kidney disease, Adv. Chronic Kidney Dis. 25 (2) (2018) 121–132.

[3] M.A. Pfeifer, J.B. Halter, D. Porte Jr., Insulin secretion in diabetes mellitus, Am. J. Med. 70 (3) (1981) 579–588.

[4] I. Kavakiotis, O. Tsave, A. Salifoglou, N. Maglaveras, I. Vlahavas, I. Chouvarda, Machine learning and data mining methods in diabetes research, Comput. Struct. Biotechnol. J. 15 (2017) 104–116.

[5] A. Dagliati, S. Marini, L. Sacchi, G. Cogni, M. Teliti, V. Tibollo, P. De Cata, L. Chiovato, R. Bellazzi, Machine learning methods to predict diabetes complications, J. Diabetes Sci. Technol. 12 (2) (2018) 295–302.

[6] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, A.A. Bharath, Generative adversarial networks: An overview, IEEE Signal Process. Mag. 35 (1) (2018) 53–65.

[7] P. Zhao, W. Tian, L. Xiao, X. Liu, J. Wu, An attention-based long short-term memory framework for detection of Bitcoin scams, in: 2022 International Conference on High Performance Big Data and Intelligent Systems, HDIS, IEEE, 2022, pp. 21–26.

[8] X. He, D. Cai, P. Niyogi, Laplacian score for feature selection, Adv. Neural Inf. Process. Syst. 18 (2005).

[9] M. Shuja, S. Mittal, M. Zaman, Effective prediction of type ii diabetes mellitus using data mining classifiers and SMOTE, in: Advances in Computing and Intelligent Systems, Springer, 2020, pp. 195–211.

[10] S. Ghosh, C. Boucher, J. Bian, M. Prosperi, Propensity score synthetic augmentation matching using generative adversarial networks (PSSAM-GAN), Comput. Methods Programs Biomed. Update 1 (2021) 100020.

[11] D. Sisodia, D.S. Sisodia, Prediction of diabetes using classification algorithms, Procedia Comput. Sci. 132 (2018) 1578–1585.

[12] K. Kayaer, T. Yildirim, et al., Medical diagnosis on Pima Indian diabetes using general regression neural networks, in: Proceedings of the International Conference on Artificial Neural Networks and Neural Information Processing, Vol. 181, ICANN/ICONIP, 2003, p. 184.

[13] Behavioral risk factor surveillance system, 1988–2023, URL https://www.cdc.gov/brfss/annual_data/annual_data.htm#print.

[14] F.R. Chung, Spectral graph theory. Number 92 in regional conference series in mathematics, Am. Math. Soc. (1997).

[15] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, J. Artif. Intell. Res. 16 (2002) 321–357.

[16] M. Zeng, B. Zou, F. Wei, X. Liu, L. Wang, Effective prediction of three common diseases by combining SMOTE with tomek links technique for imbalanced medical data, in: 2016 IEEE International Conference of Online Analysis and Computing Science, ICOACS, IEEE, 2016, pp. 225–228.

[17] C. Chethana, Prediction of heart disease using different KNN classifier, in: 2021 5th International Conference on Intelligent Computing and Control Systems, ICICCS, 2021, pp. 1186–1194, http://dx.doi.org/10.1109/ICICCS51141.2021.9432178.

[18] Y. Qi, Random forest for bioinformatics, in: Ensemble Machine Learning, Springer, 2012, pp. 307–323.

[19] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794.