Sensors and Materials, Vol. 36, No. 6 (2024) 2439–2458
MYU Tokyo

S & M 3679

Framework Integrating Generative Model with Diffusion Technique to Improve Virtual Sample Generation

Yao-San Lin,[1] Mei-Ling Huang,[1*] Der-Chiang Li,[2] and Jui-Yu Yang[2]

[1]Department of Industrial Engineering and Management, National Chin-Yi University of Technology, No. 57, Sec. 2, Zhongshan Rd., Taiping, Taichung 411, Taiwan (R.O.C.)
[2]Department of Industrial and Information Management, National Cheng Kung University, No. 1, University Road, Tainan 701, Taiwan (R.O.C)

(Received November, 15 2023; accepted June 10, 2024)

Keywords: megatrend diffusion, small sample, generative adversarial network, WGAN_MTD, box plot, punishment

In the field of small-sample domains, since its introduction, the effectiveness and practicality of the megatrend diffusion (MTD) method have been demonstrated in various studies. Recently, with the popularity of generative deep learning, researchers have integrated Wasserstein generative adversarial networks (WGANs) with the MTD method and proposed a novel framework called WGAN-MTD for virtual data generation. It uses the MTD for producing estimates, which restricts the generative model's output value range and generates effective synthetic samples. However, the validity of developing virtual samples using real-world data containing outliers remains controversial, and the weight clipping method in WGAN has been shown to affect the stability of model training. In this study, we propose an advanced framework in which the boxplot is integrated with a penalization term to limit the effect of outliers, especially from small samples. The proposed framework considers the convolutional layers to capture local information features and lower the complexity of the model by reducing the number of parameters between the input and output layers. Additionally, we adopt a WGAN with the gradient penalty (GP) method instead of WGAN alone to improve the training stability and precision in the generative model. Experimental results demonstrate that both the boxplot and the penalization term enhance the accuracy of the generative models for small datasets containing outliers.

1. Introduction

The use of deep learning has become widespread since its outstanding performance in the ImageNet image recognition competition held by Stanford University in 2012, and deep learning is now used in supervised learning, unsupervised learning, identifying the impact of convolutional networks, and image generation through generative adversarial networks (GANs). However, machine learning efficiency depends on the amount of data. The larger the amount of data, the more feature information the model can obtain and the better the recognition effect.

*Corresponding author: e-mail: huangml@ncut.edu.tw
https://doi.org/10.18494/SAM4780

ISSN 0914-4935

Therefore, the amount of data is essential in building a reliable data-driven model. A GAN effectively solves the problem of insufficient data in the image field. The simulated images generated by GAN are combined with real images to learn common features, which can improve the recognition accuracy of convolutional neural networks. However, the problem of insufficient data does not only occur in images, but also in other data types.

With the popularity of data-driven machine learning methods, data has become an increasingly expensive resource in recent years. In the modeling stage of machine learning, it is necessary to mine the characteristic information contained in the experimental objects from among a large amount of experimental data. It is not easy to obtain a large amount of experimental data in all scenarios, and it is not practical to conduct training when the amount of sample information is limited. With the assistance of AI, it is easy to reduce and damage the model's effectiveness when facing the problem of small data. In the past, some methods have used transfer learning to learn the characteristics of other data and collect similar feature information to improve the model's effectiveness. For example, Zhong and Ban[(1)] used transfer learning to deal with the classification problem of small-sample nuclear power plants. However, it might not necessarily apply to other datasets.

Douzas *et al.*[(2)] mentioned that the data features of small samples are usually loose structures with multiple information gaps. Using small samples to train models to extract features is reliable and accurate. To solve the problem of small samples, increasing the sample size should be an effective method. Therefore, many models have involved data amplification methods. The most frequently used methods in the imaging field are sample number amplification through image rotation, cutting, and scaling, as well as the previously mentioned GAN to generate images. Many different methods and models have been proposed for numerical data, such as the Gaussian mixture model GMM-VSG by Shen and Qian[(3)] and the Monte Carlo and particle swarm optimizations proposed by Gong *et al.*[(4)]

The megatrend diffusion (MTD) method is a proven and common data amplification method for data forms. Using virtual samples generated by MTD for training can improve the model's effectiveness. Li *et al.*[(5)] used MTD to generate the value range, combined it with WGAN to verify the virtual sample distribution for virtual sample plug selection, and proposed a new GAN model framework called WGAN_MTD. WGAN_MTD fills some small-sample information gaps by introducing selected virtual samples to simulate the parent distribution, thereby improving sample completeness. However, mean-centered estimation algorithms are easily affected by outliers, leading to overestimation. The Lipschitz method used in the WGAN model has been verified by Gulrajani *et al.*[(6)] to affect training stability and output sample quality. Therefore, the primary research question in our study is how to improve WGAN_MTD to enhance training stability and virtual data effectiveness.

The research objective, derived from the above, is the improvement of the WGAN_MTD model to enhance the training stability and the effectiveness of virtual data and to overcome issues related to outliers and training convergence that can impact the accuracy of the classification model. We proposed a model, called WGANGP_MTD, in which the penalty term and boxplot methods are integrated to generate virtual datasets and enhance the accuracy and stability of the training process. This model is aimed at addressing issues related to outliers and

convergence during training, ultimately improving the performance of the classification model.

The research involves using numerical data, which can be generated by sensors in various fields. By improving data generation methods for these datasets, the results of this research can help to advance sensor applications related to material properties, environmental monitoring, or other relevant fields.

2. Literature Review

2.1 Virtual sample generation method

In data-driven tasks, there are three methods to solve small-sample problems. Support vector machine (SVM), the first method, uses fewer support vectors to split the hyperplane. Although small-sample problems are solved, sample overfitting is a problem that is sensitive to noise and outliers. The second is transfer learning, in which valuable features are extracted from samples of other categories. However, a negative transfer effect would occur if the similarity between the category sample set and the original sample set is insufficient. The third method is to generate virtual samples by extracting useful information from the original data to expand the number of samples.

Efron and Tibshirani[7] proposed the bootstrap method, which repeatedly takes samples from the original sample set on the basis of probability. The disadvantage of bootstrapping is that when the sample set is tiny, it would lead to overfitting and biased samples. Chawla *et al.*[8] proposed the minority class oversampling technique, namely, the synthetic minority oversampling technique (SMOTE), which improves the random oversampling method and selects one of the k neighboring samples for the minority class sample. The limitations of the KNN algorithm restrict the information in the generated samples, resulting in the poor generalization of SMOTE. Huang and Moraga[9] proposed a diffusion neural network (DNN), which is a combination of neural-network-like and information diffusion principles, to improve backpropagation. The backpropagation neural network (BPNN) information diffusion accurately generates new samples by applying the fuzzy theory of symmetric diffusion. Still, DNN must satisfy the correlation between features to be greater than 0.9 and discard asymmetric samples; it is challenging to meet this condition in real data. Therefore, Li *et al.*[10] proposed overall trend diffusion (MTD), considering the data's integrity, with the aim of improving the diffusion of the entire sample set instead of only one-to-one sample diffusion.

2.2 Overall trend spreading

The MTD technique estimates the population's distribution by the information diffusion method based on fuzzy theory and fuzzy triangular membership functions. Huang and Moraga[9] proposed the information diffusion method to diffuse each sample individually. Li *et al.*[10] assumed that the diffusion process should be implemented individually for each sample taking into consideration the sample's position in the set. Although MTD uses information trends to

effectively overcome the problem of sample asymmetry in a diffusion sample set, two primary conditions must be met: feature independence and feature value allocation. Zhu *et al.*[11] mentioned that a triangle distribution may not sufficiently describe the actual population. It might be too naive to set a triangular distribution to describe the population since the information available from small samples is sparse and discrete, and there are information gaps. Therefore, multi-distribution megatrend diffusion (MD-MTD) is proposed, in which uniform and triangular distributions are adopted to describe the data distribution. The closer to the center point, the greater the possibility of occurrence.

As mentioned earlier, MTD requires the assumption of feature independence. In the process of collecting data, the dependence would continue to increase. When a dataset possesses features of dependence, the effect of the MTD method is limited, especially for features with a specific order. Lin and Li[12] added the increment concept to MTD and proposed generalized trend diffusion (GTD) in the case of time series data. Yu *et al.*[13] combined the Monte Carlo method and MTD. The a and b obtained by MTD were sampled by the Latin hypercube sampling method and then through extreme learning machine prediction. Research has shown that this method improves prediction accuracy.

2.3 Boxplot

In 1978, McGill *et al.*[14] proposed a graphical narrative statistical method to represent the data distribution pattern, called a box-and-whisker plot. Users can clearly understand the overview of skewness (Skewness) and outliers (Outliers) in the data through the graphics presentation. Among them, Q1 is the first quartile, Q2 is the median, and Q3 is the third quartile. The interquartile range (IQR) can be obtained as the distance between Q1 and Q3. Data exceeding 1.5 times the IQR can be regarded as outliers. Li *et al.*[15] combined the box-and-whisker plot concept with MTD, using the median as the center point and constructing a fuzzy triangular membership function. They used the IQR of the box-and-whisker plot to adjust the estimated matrix value range, that is, the upper and lower bounds. We also consider the problem of insufficient small samples, avoid accidentally deleting a small number of samples, retain outliers, and include them in the estimation range. After estimating the upper and lower bounds and the center point of the matrix using the box-and-whisker plot, the fuzzy triangular membership function (MF) can be constructed.

2.4 Generating adversarial networks and WGAN_MTD

In recent years, GANs have been widely used in image generation and proven to achieve good results. GAN was first proposed by Goodfellow *et al.*[16] and mainly consists of two networks: the generator and discriminator. The generator takes noise or random normal distribution as input and imitates the real sample for output. The discriminator judges whether the input of the generator is the same as the real sample. The two networks fight against each other and adjust the parameters. Finally, the output of the generator can make the discriminator judge it as accurate. GAN has also been extended to many other variants, such as DCGAN, WGAN, TimeGAN, and WGAN_MTD, used for other data types.

WGAN is an improved model proposed by Arjovsky *et al.*[17] to replace traditional GAN. It introduces Wasserstein distance to replace the Jensen–Shannon and Kullback–Leibler divergences to solve the mode-dropping problem of traditional GAN and maintain the diversity of generated samples. Training WGAN does not require maintaining the balance between the generator and the critic (or the discriminator), and the most significant benefit of WGAN is that it can continuously estimate the Wasserstein distance/Earth–mover (EM) distance to draw the learning curve. Arjovsky *et al.* pointed out that traditional GAN cannot converge using the Jensen–Shannon divergence in two nonoverlapping distributions, and thus the gradient disappears, making training difficult. When using the Kullback–Leibler divergence, its asymmetry causes the generator to lose diversity. WGAN effectively improves the vanishing gradients problem when diversity and distribution do not overlap. However, because a rough processing method (when the weight exceeds the critical value, the weight is set to the highest or lowest critical value) was used, they subsequently proposed the WGAN-GP model.

WGAN has made breakthroughs in stable training, but sometimes, there are still problems of low-quality samples being generated and failure to converge. Gulrajani *et al.*[6] found that these problems are a result of WGAN using weight clipping as a Lipschitz constraint, so they proposed a gradient penalty (GP). As an alternative, a GP was added as a constraint in the gradient stage and published as WGAN-GP. However, the disadvantage of using the penalty term is that batch normalization cannot be added to the discriminator because it would rely on other samples in the same batch, which is inconsistent with a GP on individual samples.

Li *et al.*[5] combined MTD with WGAN, replaced CNN with BPN, and set the activation function in the generator output layer to tanh to adjust the applicable MTD. They divided the architecture into two parts. First, the maximum and minimum values of the small-sample data were obtained, and then the average was used as the center point to calculate the estimated upper and lower bound values a and b, based on which U[0,1] and random seeds were uniformly distributed to generate dummy values. The generated virtual value must eventually be pushed back to the original value, so the MF fuzzy triangular membership function was established through the upper and lower bounds and the center point. Li *et al.* set the MF value of the center point to 0, MF(a) to −1, and MF(b) to 1, and used the modified MF as the WGAN excitation function. The input value range was limited through MTD. The value range could be closer to reality than the original random generation. The membership function values generated by MTD were input into WGAN to iteratively identify and repeatedly adjust the noise, generate a virtual membership function value distribution with the original data characteristic distribution, and convert it into actual values. They called this architecture WGAN_MTD.

According to the above literature, as Gulrajani *et al.*[6] mentioned, WGAN has problems that may lead to failure to converge during the training process. In this study, we considered WGAN-GP to be more suitable than WGAN, and estimating the center point with the average value causes estimation errors when converting membership function values. Therefore, we adjusted the WGAN used in the WGAN_MTD generation model, referring to Zhu *et al.*[11] The skewness penalty term and box-whisker plot methods was proposed in 2016 to improve WGAN_MTD, model accuracy, and training stability.

3. Methodology

Given that the original WGAN_MTD approach results in an excessive filling of outliers, we consider that the boxplot method enables the sample points to formulate a membership function whose values are unaffected by outliers. Moreover, if the training sample already contains outliers, the expanded value range could affect the validity of the virtually generated data. The proposed model adopts the penalty term to effectively restrict the scope for sample generation. The penalty term controls the sample scope to within the estimated range and can also affect the virtual sample's quality depending on its size.

The virtual membership function values are mapped back to the original data range using the inverse membership function for each feature. These steps constitute the generation of data using the adjusted WGAN-GP. The training datasets are obtained by incorporating the penalty term and boxplot methods. Additionally, the data generation is performed using this model, which is referred to as WGANGP_MTD. The three virtual datasets, along with the real dataset, are trained and classified by WGANGP_MTD. Finally, the performance characteristics of WGANGP_MTD are compared using evaluation methods to determine their relative merits. Figure 1 shows the flowchart of our research structure.

As shown in Fig. 1, the first step is to check for outliers by the boxplot method when obtaining a small data sample. If outliers are present, the subsequently implemented MTD method is integrated with the penalty term and boxplot methods to produce membership function values for each feature. The membership function values are equivalent to representing the likelihood of occurrence in the population.

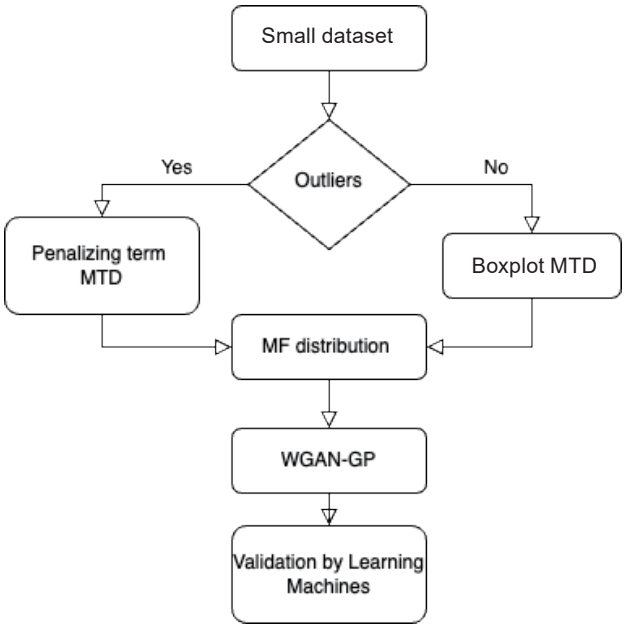


Fig. 1. Research methodology.

3.1 Outliers and WGAN_MTD

Outliers refer to those data points far from the center. Outliers could easily affect sample parameters, and the effect would be more severe when handling a small dataset.[11] The same issues could occur with WGAN_MTD, which starts by computing the center points. WGAN_MTD calculation uses the average value of the training data as the center point. It obtains the estimated upper and lower bounds a and b for its population for further computation of the plausibility through the membership function. Table 1 lists all the symbol definitions.

Furthermore, if a is less than the minimum of the training data, a is set as the minimum, min; if b is greater than the maximum of the training data, b is set as the maximum, max, as shown in Eq. (1).

$$\begin{cases} a, a < min \\ min, a > min \\ min / 5, S^2 = 0 \end{cases} \quad \text{and} \quad \begin{cases} b, b > max \\ max, b < max \\ 5min, S^2 = 0 \end{cases} \tag{1}$$

Virtual values generated using uniform random seeds would help fill the training data gap by smoothing since there are many gaps among points. WGAN_MTD considers every data value inside a small sample essential so that the data sets are regarded as something other than the outlier. Under this concept, the data range should cover all the points. To treat the outliers as ordinary data, we integrated existing improvement methods and proposed two processing methods in which penalty terms and boxplots were added for comparison.

We consider adding the penalty item to the model since we consider that errors caused by outliers affect the data. When calculating the skewness, a penalty term r is added as the degree of influence from outliers, and the original version, WGAN_MTD, is modified to obtain the membership function value for WGANGP_MTD. The overall steps are as follows.

Table 1
Notation table.

Symbol	Definition
a	Lower bound of the estimated range of population
b	Upper bound of the estimated range of population
$\mathbb{D}$	Sample set
$\mathbb{D}_f$	Feature data in $\mathbb{D}$
r	Skewness correction
u_{set}	Center point of the estimated range of population
N_L	Number of samples in $\mathbb{D}$ that are less than u_{set}
N_U	Number of samples in $\mathbb{D}$ that are greater than u_{set}
L	Cost function/loss function
θ	Hyperparameters of generator
w	Hyperparameters of discriminator
m	Batch size
$\hat{x}$	A space in which random interpolation between true and virtual samples is conducted
$\tilde{x}$	Samples generated by the generator using θ and input z and are equal to those produced with $G>(z)$

1. Obtain the maximum value max and minimum value min from the sample set $\mathbb{D}$ and set the center point to its average.
2. Update the upper skewness $Skew_L$ and lower skewness $Skew_U$, and then calculate the estimated range for obtaining the upper and lower bounds a and b. Equation (1) is useful to monitor a and b to prevent excessive discrepancy with the original upper and lower bounds a' and b', as shown in Fig. 2, where $Skew_L$ and $Skew_U$ are defined as follows.

$$Skew_L = N_L / \left(N_L + N_U + r\right)$$
$$Skew_U = N_U \quad \left(N_L + N_U + r\right)$$

3. Apply Eq. (2) to calculate the triangular membership function value with a value range between −1 and 1 and convert it into a new sample feature value.

$$MF\left(i\right) = \begin{cases} \left(i - u_{set}\right) / \left(u_{set} - a\right), i \le u_{set} \\ \left(b - i\right) / \left(b - u_{set}\right), i > u_{set} \end{cases} \tag{2}$$

Replacing the mean with the median as the center point expands the estimation range. However, using the median as the center point for the membership function is not affected by outliers. The adjustment steps are as follows.

1. Take the median from sample set $\mathbb{D}$ and set it as the center point. Calculate the Q1 and Q3 values of each feature to obtain the upper and lower bounds (a and b). If the bounds exceed the maximum and minimum values, replace them as the maximum and minimum values.
2. Obtain the fuzzy triangular membership function values within the range of [−1, 1]. These values serve as the new feature values for the samples.

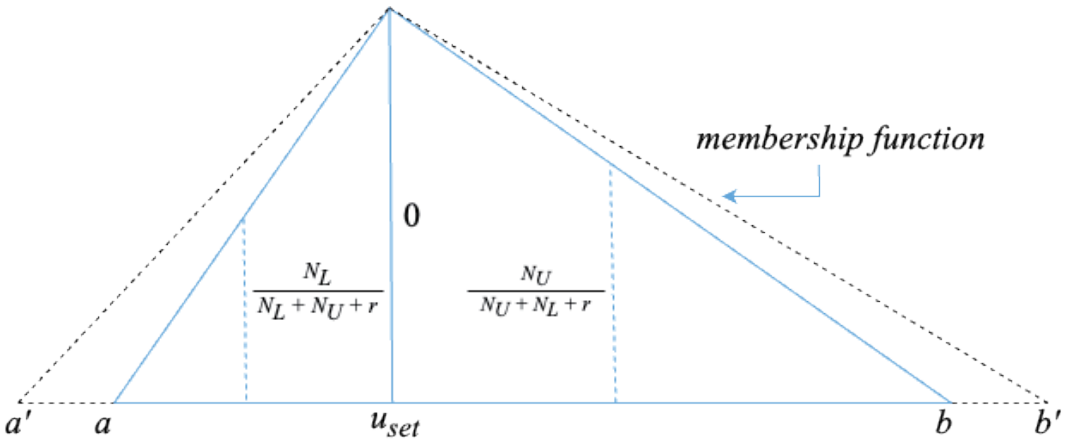


Fig. 2. (Color online) When estimating the range, $Skew_L$ and $Skew_U$ are useful in the formulation.

3. These steps outline the adjustment process where the median is used as the center point, ensuring that outliers do not affect the membership function. By using the membership function, we obtain a new distribution of features. The complete process is illustrated in Fig. 3.

3.2 WGAN-GP

WGAN, with the weight-clipping step, sometimes causes convergence issues in the training phase, and the same concerns could occur with WGAN_MTD, which operates WGAN as the backbone network. Therefore, the improved version introducing the GP term, known as WGAN-GP, supersedes the Lipschitz constraint of WGAN/WGAN_MTD since it allows the gradient to maintain a stable and improved convergence speed during the backpropagation process instead of being limited to the set clipping range. Figure 4 illustrates the adjusted model structure.

WGAN_MTD uses batch normalization (BN) in the critic/discriminate model to normalize the batch samples. BN is a commonly used regularization technology that standardizes the features in each minibatch to better control the output range of neurons with the aim of stabilizing the internal distribution of the neural network during training. However, the normalization process of BN could destroy the computation required by the GP, reduce the effect

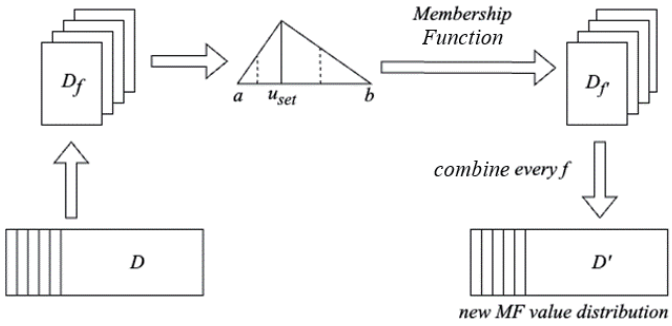


Fig. 3. Flowchart of feature distribution transformation.

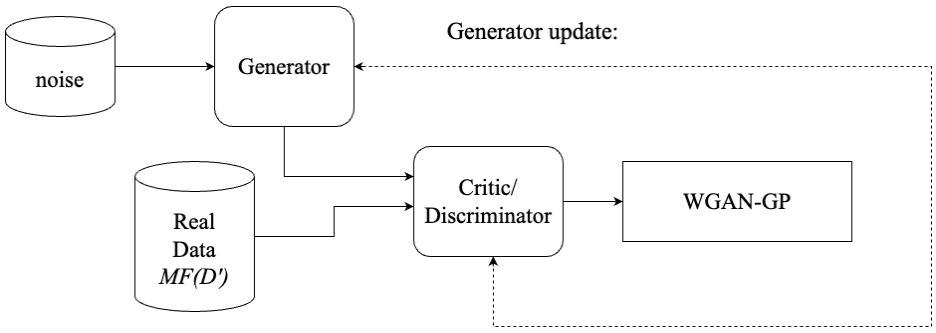


Fig. 4. Structure of the adjusted model.

of WGAN-GP's penalty term, and lead to unstable training. Given the reasons above, we do not employ BN in our research process, and the optimizer is adjusted to change RMSProp to Adam.

Dense structures are suitable for processing more straightforward data, such as vectors or matrices, while convolutional neural networks are mainly presumed to process image data. Convolution layers apply the same convolution kernel map in local areas to learn features for capturing local patterns inside the data. Considering that the features probably have implicit nonlinear relationships through membership functions, convolutional neural networks are usually more suitable than dense neural networks in our research.

Following the process described above, the convolution kernel can be (1, 6), assuming that the feature value is 6, and a new feature value is calculated on each convolution kernel to capture the relationship between different positions in the data.

Since convolutional layers usually process three-dimensional data, it is necessary to convert the values of $M(D')$ into three dimensions before the convolution step. The first dimension is the data length, the second is the feature length, and the third is the feature value. For the discriminator's model of WGAN-GP, the goal is to minimize the Wasserstein distance, whereby the discriminator of the better-trained model would possess a loss close to −1.

As shown in Fig. 5(a), the structure of the discriminant model in this study comprises each layer and the corresponding detail. The training data size is assumed to be 10, with six features as input. The first four layers comprise the convolution layer, LeakyReLu, Dropout, and Flattening. The last layer, Dense, determines the authenticity.

The aim of the generative models is to maximize the Wasserstein distance. When there is a lower GP value, the difference between the data generated by the generator and the real data is negligible. The loss of the generator of the better-trained model would be close to 1. Still, if the generated data are not sufficiently acceptable, an adjustment for the generative model would be required.

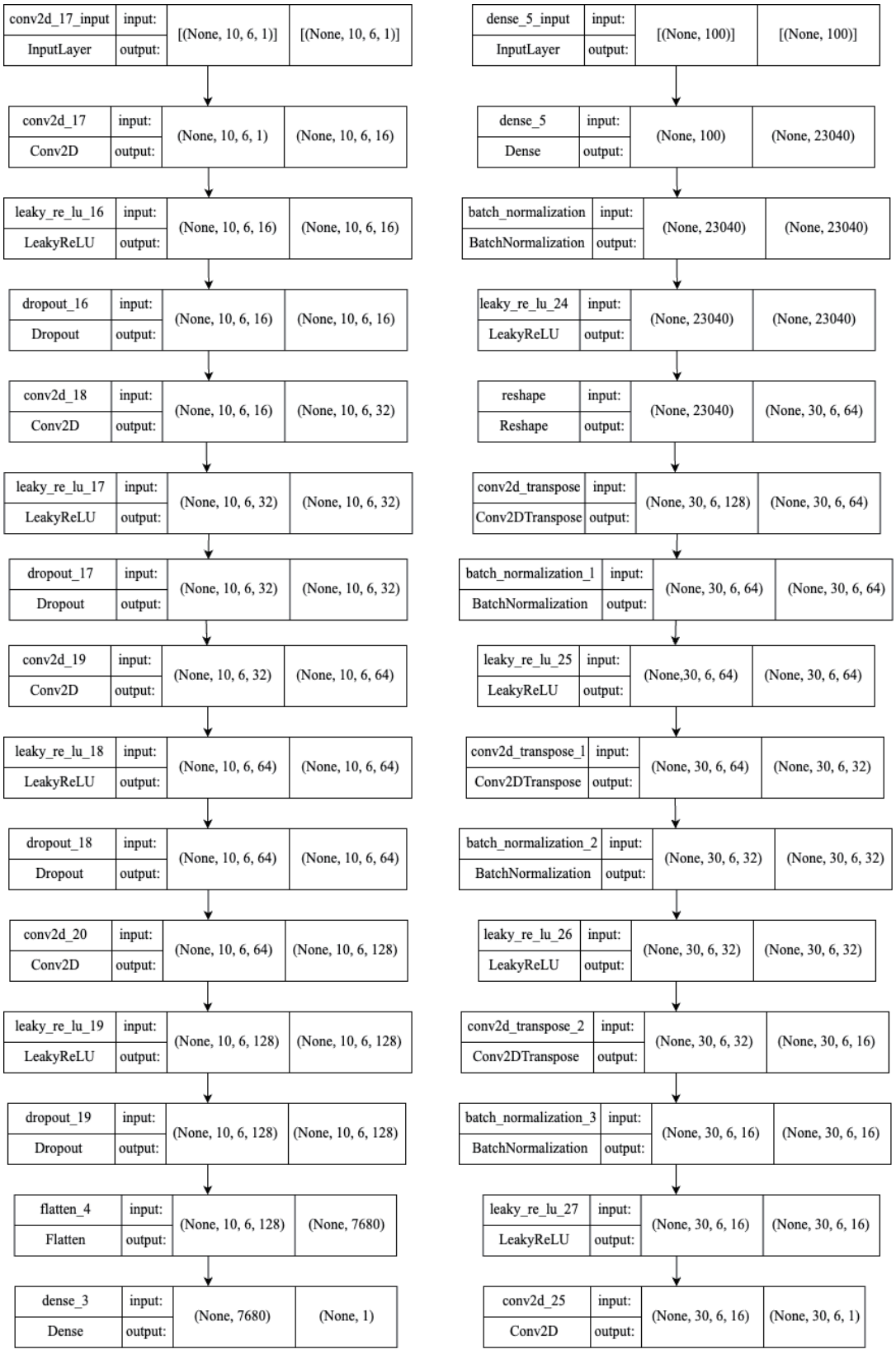
In this study, the input of the generative model is random, normally distributed data. Figure 5(b) presents the layer structure. The first four layers comprise convolutional layers, LeakyReLu and BN, and the last layer comprises tanh activation functions.

To summarize, the steps for generating data after adjusting WGAN-GP are as follows.

1. Randomly generate normally distributed noise as the input of the generator.
2. Scale the value to the [−1,1] interval through $MF(\mathbb{D}')$ and convert it into three-dimensional data as the discriminator input.
3. Set the number of training cycles of the model parameters as five times for discriminant models and one time for generative models.
4. Generate the data following the [−1,1] interval after training the generator, and deduce the generated virtual membership function value using the inverse membership function of each feature to estimate the original data range value.

3.3 Assessment methods

We analyze the classification results of each method using the average accuracy of cross-validation, which indicates whether adding or not adding virtual samples can help improve the accuracy of the classification model.



(a) (b)

Fig. 5. (a) Generator model. (b) Critic/discriminator model.

When training the model, the gradient vanishing implies that the gradients used to update the weight dramatically vanish. In contrast, if the update rate is very high, the loss would oscillate and fail to converge, meaning the occurrence of a gradient explosion. Limiting the number of training epochs can be effective in evaluating the gradient vanishing and explosion, while analyzing the loss records for each training cycle helps verify whether this research method can alleviate the gradient issues during training.

For evaluating the quality of the virtual sample, we suggest reducing the dimensionality of the generated virtual data, and we display the regional range of the virtual model in a visual view and analyze the output through the Mann–Whitney U test and the Kruskal–Wallis test to determine whether the two independent virtual and real samples come from similar populations.

In the t-test, based on the assumption of normal distribution, we must assume that the numbers of variations in the two samples are identical. The t-test is effective statistically for processing normally distributed data. However, when the sample does not follow a normal distribution, the t-test might be insufficient for statistical inference. The Mann–Whitney U test does not require a sample from a normal population and is used to compare the difference in medians between two samples. In contrast, the t-test is used to compare the difference in means between two samples. Therefore, the Mann–Whitney U test is more appropriate than the t-test when the sample distribution is asymmetric or has extreme values.

In addition to the Mann–Whitney U test using dimensionality reduction, we also adopted the Kruskal–Wallis test, which is mainly used to compare three or more groups. The Kruskal–Wallis test can handle non-normal data without assuming equal variation among the data groups. It is suitable for comparing more than three independent samples and can compare rank medians with less impact on outliers. At the same time, the multivariate t-test requires the assumption that the data conforms to a normal distribution and therefore might be less effective for non-normal data. Still, it can obtain the average and standard deviation between different groups for further comparison and analysis, and the comparison results are more accurate. The multivariate t-test is suitable for comparing two or more dependent or independent samples but is sensitive to outliers. Both the Mann–Whitney U test and the Kruskal–Wallis test are nonparametric tests. The distribution-assumption-free test does not rely on the parameters of the population but makes inferences based on the order relationship of the samples. It is suitable for situations where the sample data do not conform to the normal distribution, the sample data variations are not equal, and the sample data have missing values.

Our research method combines the WGAN-GP and MTD model structure and is referred to as WGANGP_MTD. The models with the boxplot and penalty term methods are called Box WGANGP_MTD and Punish WGANGP_MTD, respectively.

4. Experiments

4.1 Experimental datasets

In this study, we use the Seeds Dataset and Raisin Dataset of the UCI Machine Learning Repository at the University of California, Irvine.

The Seeds Dataset uses three different categories of wheat grains as category attributes and six features for classification tasks. The categories are Kama, Rosa, and Canadian, and the features include continuous values such as area, perimeter, compactness, kernel width, asymmetry coefficient, and kernel length. There are a total of 210 samples, and each class comprises 70 samples.

The Raisin Dataset uses two varieties of raisins as category attributes and seven features for classification tasks. The categories are Kecimen and Besni, and the features include continuous values such as area, perimeter, proportion, raisin length, eccentricity, and convexity. There are a total of 900 samples, and the number of samples in each category is 450.

Outliers refer to extreme points. They may occur as a result of measurement errors, data errors, or real extreme situations. Since the existence of these values may be detrimental to the training and prediction of the model, to ensure that each training data contains outliers, after introducing all the data, first define the outliers by the boxplot method, classify the data into categories, and then calculate the IQR and upper and lower bounds of all attributes of the category to classify the data into Outlier and Inlier groups.

An outlier is a data value with outlier attributes, and an inlier is a data value with all features within the upper and lower bounds. The number of outliers in the customized training data is randomly selected from the Outlier group. An inlier is added as the small-sample training data, and the remaining outliers and inliers are used as test data.

4.2 Experimental methods and results

To meet the desired generalization performance of the machine learning model on small-sample data, we conducted verification on the basis of the model convergence effect and the accuracy rate under different parameters corresponding to the research purpose. The cross-validation method was used to compare the model effects.

Table 2 gives the experimental environment of the study. Python is used as the programming language to construct the experiment. Python is a high-level programming language. It has concise and easy-to-read syntax and rich third-party libraries and tools, so it is highly suitable as a programming language for machine learning. The development environment uses TensorFlow. TensorFlow is an open-source machine learning framework developed by the Google development team. One of its main features is that it can efficiently perform distributed computing to speed up model training. The operating system and GPU are macOS and Apple M1, respectively.

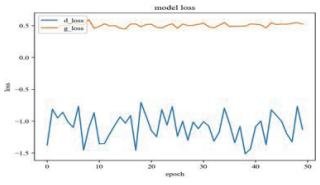
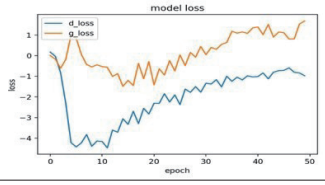
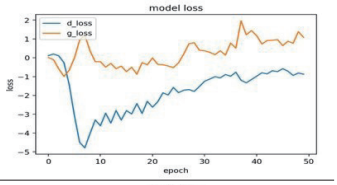
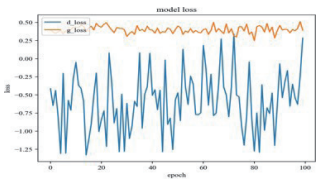
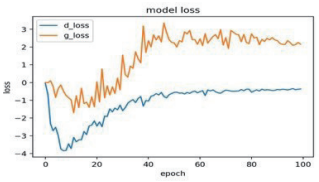
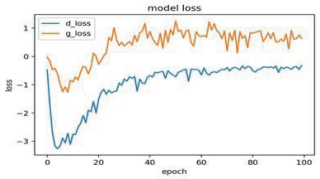
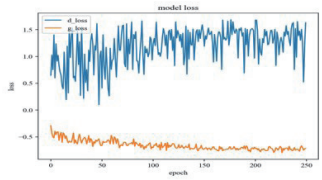
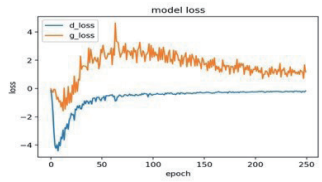
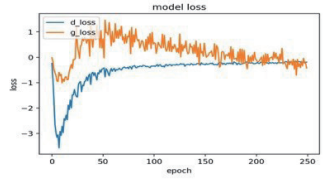
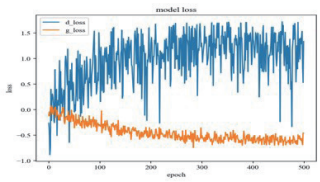
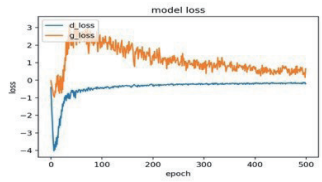
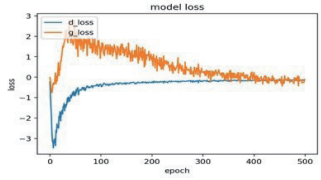
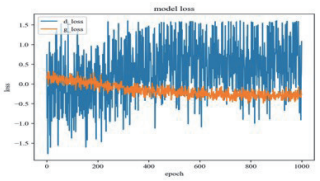
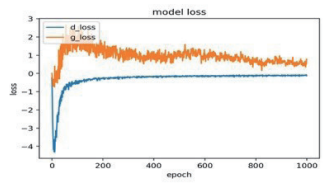
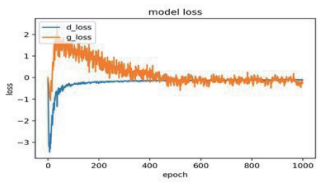
Table 2
Environment details for the experiments.

Framework	Python 3.8, TensorFlow
Operation system	macOS 2.8.0
CPU	16 GB
GPU	Apple M1

When verifying model convergence, we eliminate data imbalance and missing data, classify the Seeds Dataset through data preprocessing, and then extract six pieces of data from each category as training data.

Set the epoch of the generative adversarial model to 50, 100, 250, 500, and 1000, and compare the discriminant model loss value (D Loss) of the WGAN_MTD model with those of the Box and Punish WGANGP_MTD models. The plots in Table 3 illustrate how the generated model loss value (G Loss) has completed convergence, where the Kama category was taken as an example. It is known from the results that the loss function value of WGAN_MTD fluctuates up and down more than those of the other two models. The reason is that the excessive pruning of weights causes loss function values to fail to converge. Neither D nor G Loss can become stable, and the loss function cannot be reduced or decreases rapidly. This would lead to problems such as the overfitting of the generator or gradient explosion. The D Loss of WGANGP_MTD with the boxplot or the penalty term steadily decreases from Epoch 50. The D Loss of Epoch 1000

Table 3
(Color online) The generated model loss value (G Loss) has completed convergence, with Kama taken as an example.

	WGAN_MTD	Box WGANGP_MTD	Punish WGANGP_MTD
Epoch 50			
Epoch 100			
Epoch 250			
Epoch 500			
Epoch 1000			

approaches −1 and the G Loss approaches 1, which aligns with expectations. The generator loss value first decreases and then increases. In the early stages of training, the samples generated by the generator may be very different from the actual samples. At the same time, the discriminator can easily distinguish between actual and generated samples. Therefore, the loss value of the generator is higher in the early stages, and its generated sample is of lower quality. With continuous training, the generator gradually learns how to generate samples closer to real data, so the generator loss value decreases. When the generator has generated relatively realistic samples, the output of the discriminator for actual and generated samples may approach −1. At this point, the generator loss can drop to a value close to 1.

Although WGANGP_MTD can make the model stable to train, it still relies on the classification model to compare the quality of the samples generated and to determine whether it could help improve the model performance.

To test whether the method proposed in this study can improve the accuracy and efficiency, 6, 15, and 25 strokes from each category were selected as training samples. Different numbers of outliers were added to them to train and generate the generative adversarial model. Then, the generative adversarial model comprised the generated 6000 virtual samples combined with the extracted real training samples as training data for the SVM model.

Because the number of categories of training and test samples in this study is balanced, and the Accuracy and F-score are almost the same, the accuracy is selected as the performance evaluation index to verify the performance of each method in the SVM model.

We compare the performance characteristics of different numbers of outliers and Accuracy when the number of samples gradually increases. *Col* represents the number of outliers included in the training data. The experimental results of the Seeds Dataset are given in Table 4, and the experimental results of the Raisin Dataset are shown in Table 5.

Table 4
Experimental results of Seeds Dataset using six original samples of each category with a total of 6000 virtual data.

Sample size	Original		WGAN_MTD		Convolutional WGAN_MTD		WGANGP_ MTD		Box WGANGP_ MTD		Punish WGANGP_ MTD	
	Col=3	*Col*=6	*Col*=3	*Col*=6	*Col*=3	*Col*=6	*Col*=3	*Col*=6	*Col*=3	*Col*=6	*Col*=3	*Col*=6
18	0.772	0.755	0.851	0.785	0.878	0.864	0.877	0.866	0.885	0.897	0.886	0.867
45	0.833	0.830	0.864	0.889	0.895	0.888	0.910	0.92	0.901	0.911	0.913	0.924
75	0.859	0.866	0.837	0.892	0.895	0.924	0.924	0.934	0.906	0.927	0.921	0.935

Table 5
Experimental results of Raisin Dataset using six original samples in each category with a total of 6000 virtual data.

Sample size	Original		WGAN_MTD		Convolutional WGAN_MTD		WGANGP_ MTD		Box WGANGP_ MTD		Punish WGANGP_ MTD	
	Col=2	*Col*=4	*Col*=2	*Col*=4	*Col*=2	*Col*=4	*Col*=2	*Col*=4	*Col*=2	*Col*=4	*Col*=2	*Col*=4
12	0.705	0.676	0.771	0.733	0.788	0.766	0.767	0.739	0.789	0.769	0.770	0.744
30	0.754	0.737	0.819	0.799	0.816	0.806	0.818	0.784	0.822	0.818	0.816	0.785
50	0.789	0.771	0.799	0.801	0.815	0.818	0.797	0.800	0.822	0.822	0.797	0.797

It can be seen from Table 4 that when the cases include three and six outliers within the uniform sample size of 18, the SVM classification model without added virtual data (which is the original dataset) performs poorly and results in the worst accuracy. After adding 6,000 virtual data, the WGAN model adopting the Convolution layer performs better than that with the Dense layer. Moreover, the classification results obtained using WGAN-GP are better than those obtained using WGAN. In the case of three outliers, the classification model trained with the virtual data augmented by the penalty term (Punished WGANGP_MTD) has the highest accuracy, which is 14.2% higher than that of the original SVM classification model. In the case of six outliers, the classification model trained with the virtual data generated by Box WGANGP_MTD also has a higher accuracy that is 11.4% higher than that of the original SVM classification model.

With the uniform sample size of 45, Punish WGANGP_MTD has the highest accuracy in the cases of three and six outliers. The accuracies in the three- and six-outlier cases are 8 and 9.4% higher than those of the original SVM classification model, respectively. Furthermore, we found that with three and six outliers within the uniform sample size of 75, the SVM classification model WGAN_MTD performs poorly with a lower learning accuracy than the original SVM classification model. With three outliers, the classification accuracies of WGANGP_MTD and Punish WGANGP_MTD are about 6.5 and 6.9% higher than that of the original SVM classification model, respectively.

As seen in Table 5, when there are two and four outliers within the uniform sample size of 12, the SVM classification model without added virtual data (original dataset) performs poorly and results in the worst accuracy. After adding 6000 virtual data, the WGAN model adopting the Convolution layer performs better than that with the Dense layer. Moreover, the classification results of WGAN-GP are better than those of WGAN. In the cases of two and four outliers, the classification model trained with the virtual data augmented by the penalty term (Punish WGANGP_MTD) has the highest accuracy, 8.4 and 9.3% higher than those of the original SVM classification model, respectively. This shows that overall, in the cases of two and four outliers in the uniform sample sizes of 30 and 50, Box WGANGP_MTD performs better with higher accuracy.

From the above research results, we know that when the small sample dataset contains outliers, it affects the judgment of the classification model. As the number of data items increases, the effect of outliers gradually decreases. In the small sample dataset, the virtual generation method by Box WGANGP_MTD performs better than other models. The noise and inconsistency in data lacking feature values may be more apparent, and the boxplot method can improve the model's performance. Moreover, Punish WGANGP_MTD performs better than WGAN_MTD and WGANGP_MTD for the Seeds Dataset with small sample sizes of 45 and 75.

The Convolution and Dense layers lead to different performance characteristics of WGAN_MTD for different datasets. For the Seeds and Raisin Datasets, WGAN_MTD using the Convolution neural network method is, on average, better than WGAN_MTD using the Dense neural network.

Overall, the MTD method with a penalty term can help reduce the bias between features and improve the consistency and comparability of data, thereby improving the performance and

stability of the classification model. For the small sample dataset, Punish WGANGP_MTD is very practical for improving the model's generalization performance. However, when the number of samples is less than 30 or the training model performance is poor, the limited MTD range method could weaken the model's prediction performance, rendering Box WGANGP_MTD a better alternative method.

To compare the quality of virtual samples, in this study, we adopted the Kruskal–Wallis test and the Mann–Whitney U test after dimensionality reduction to determine whether there are statistical differences among the 50 virtual samples. The results are listed in Tables 6–9. Assume that

$$H_0 : \mu_d = 0 \quad \text{vs} \quad H_1 : \mu_d \neq 0 , \tag{3}$$

where, for H_0, all samples are assumed to come from the same population distribution, and for H_1, at least one sample is assumed to come from a different population distribution. Among them, d = 18 is the error between n virtual and real samples, and the verification level α is 0.05.

Since there is no significant difference in the number of virtual samples when the Seeds sample size is less than 30, Box WGANGP_MTD is better than Punish WGANGP_MTD. For more than 30 transactions, Punish WGANGP_MTD outperforms other models and its accuracy is similar to the average accuracy. For Raisin samples, Punish WGANGP_MTD outperforms other models, followed by Box WGANGP_MTD. Both Box WGANGP_MTD and Punish WGANGP_MTD perform better than WGAN_MTD. It can be inferred from the above results that if the number of samples is small, the boxplot method can help reasonably expand the sample estimation range from the features and obtain more complete matrix estimation features. However, if the prediction model's performance is good, using the penalty term method can more effectively deal with the problem of outliers and improve the prediction model's performance.

Table 6
Mann–Whitney test results for virtual Seeds samples.

Sample size	WGAN_MTD		Box WGANGP_MTD		Punish WGANGP_MTD	
	Col=3	*Col*=6	*Col*=3	*Col*=6	*Col*=3	*Col*=6
18	0	0	13	9	8	3
45	0	0	29	27	29	29
75	0	0	41	30	44	43.

Table 7
Kruskal–Wallis test results for virtual Seeds samples.

Sample size	WGAN_MTD		Box WGANGP_MTD		Punish WGANGP_MTD	
	Col=3	*Col*=6	*Col*=3	*Col*=6	*Col*=3	*Col*=6
18	0	0	13	9	8	3
45	0	0	29	27	29	29
75	0	0	41	30	44	43.

Table 8
Mann–Whitney test results for virtual Raisin samples.

Sample size	WGAN_MTD		Box WGANGP_MTD		Punish WGANGP_MTD	
	Col=3	*Col*=6	*Col*=3	*Col*=6	*Col*=3	*Col*=6
12	0	0	0	1	0	0
30	0	0	0	0	1	0
50	0	0	0	2	0	1

Table 9
Kruskal–Wallis test results for virtual Raisin samples.

Sample size	WGAN_MTD		Box WGANGP_MTD		Punish WGANGP_MTD	
	Col=3	*Col*=6	*Col*=3	*Col*=6	*Col*=3	*Col*=6
12	0	0	0	1	0	0
30	0	0	0	0	1	0
50	0	0	0	2	0	1

5. Conclusions

The aim of this research was to develop a model, called WGANGP_MTD, combined with the penalty term and boxplot methods to generate virtual datasets and improve model accuracy and training stability. The model adjusts the WGAN used in WGAN_MTD to address issues related to outliers and convergence during training. By incorporating the penalty term and boxplot methods, the model aims to produce high-quality virtual datasets for training and classification, ultimately enhancing the performance of the classification model.

The proposed model's goal of refining the WGAN_MTD approach to overcome issues related to small sample sizes and data amplification methods is particularly relevant to sensor technology, where dealing with limited data and enhancing data quality are common challenges in developing efficient sensor systems. Overall, our focus on improving model accuracy, stability, and data generation methods aligns with the goals of sensor technology to enhance data processing, classification, and overall performance in sensor applications.

In processing sample generation, outliers are a problem that cannot be ignored. In this study, we compared the method of limiting the MTD value range and combining the boxplot to prevent the generative adversarial network from overestimating the value range when generating virtual data. At the same time, the weight processing method of WGANGP was introduced to improve the training stability.

Experiments were conducted using the Seeds Dataset containing outliers. When the number of training samples is greater than 30, the prediction accuracy of the Punish WGANGP_MTD method is better than those of other methods. The reason is that more real samples can be shown since the number of real samples increases because the average deviation is slight, and narrowing the estimation range has a specific effect on generating virtual samples for generating adversarial networks. However, when the number of training samples is less than 30, the Box WGANGP_MTD method has higher prediction accuracy and performance than the Punish WGANGP_MTD method. The characteristics of the boxplot chart can give reasonable estimates of the range affected by outliers.

In the field of small-sample learning, the restricted MTD method can help reduce the deviation between features and improve the consistency and comparability of data, thereby improving the performance and stability of the classification model. However, the effect of restricting MTD is limited without real features. Therefore, different methods must be selected for different situations to obtain the best results when using MTD with a generative adversarial model.

During the practical research process, it was considered that 1.5IQR of the combined boxplot would widen the range and that 0.5IQR would be suitable for computing the membership function value. It was found that adjusting the range of the combined boxplot IQR and the penalty term can increase the model accuracy, but overadjustment would reduce the model accuracy. Moreover, the quality of the training model is related to the accuracy. Other models or encode and decode methods can be alternatives for processing the data. Therefore, the most suitable generalization parameter values and models are worth exploring.

Acknowledgments

This study is partially supported by the National Science and Technology Council, Taiwan (grant contract no. NSTC 112-2221-E-167-032-MY2).

References

1 X. Zhong and H. Ban: Ann. Nucl. Energy **175** (2022) 109201. https://doi.org/10.1016/j.anucene.2022.109201
2 G. Douzas, M. Lechleitner, and F. Bacao: PLoS One **17** (2022) :e0265626. https://doi.org/10.1371/journal.pone.0265626
3 L. Shen and Q. Qian: Comput. Mater. Sci. **211** (2022) 111475. https://doi.org/10.1016/j.commatsci.2022.111475
4 H. F. Gong, Z. S. Chen, Q. X. Zhu, and Y. L. He: Appl. Energy **197** (2017) 405. https://doi.org/10.1016/j.apenergy.2017.04.007
5 D. C. Li, S. C. Chen, Y. S. Lin, and K. C. Huang: Appl. Sci.-Basel **11** (2021) 10823. https://doi.org/10.3390/app112210823
6 I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville: arXiv.1704.00028 (NIPS, 2017) 30. https://doi.org/10.48550/arXiv.1704.00028
7 B. Efron and R. J. Tibshirani: An Introduction to the Bootstrap. (CRC press, New York, 1994) 1st ed. https://doi.org/10.1201/9780429246593
8 N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer: J. Artif. Intell. Res. **16** (2002) 321. https://doi.org/10.1613/jair.953
9 C. Huang and C. Moraga: Int. J. Approximate Reasoning **35** (2004) 137. https://doi.org/10.1016/j.ijar.2003.06.001
10 D. C. Li, C. S. Wu, T. I. Tsai, and Y. S Lina: Comput. Oper. Res. **34** (2007) 966. https://doi.org/10.1016/j.cor.2005.05.019
11 B. Zhu, Z. Chen, and L. Yu: CIESC J. **67** (2016) 820. https://doi.org/10.11949/j.issn.0438-1157.20160154
12 Y. S. Lin and D. C. Li: Cent. Eur. J. Oper. Res. **207** (2010) 121. https://doi.org/10.1016/j.ejor.2010.03.026
13 X. Yu, Y. He, Y. Xu, and Q. Zhu: J. Phys. Conf. Ser. **1325** (2019) 012079. https://doi.org/10.1088/1742-6596/1325/1/012079
14 R. McGill, J. W. Tukey, and W. A. Larsen: Am. Stat. **32** (1978) 12. https://doi.org/10.2307/2683468
15 D. C. Li, C. C. Chen, C. J. Chang, and W. C. Chen: Int. J. Prod. Res. **50** (2012) 1539. https://doi.org/10.1080/00207543.2011.555430
16 I. Goodfellow, J. Pouget-Abadie, Mirza M, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio: NeurIPS Proc. (NIPS, 2014) 27. https://papers.nips.cc/paper/5423-generative-adversarial-nets
17 M. Arjovsky, S. Chintala, and L. Bottou: Proc. 34th Int. Conf. Machine Learning **70** (PMLR, 2017) 214. https://proceedings.mlr.press/v70/arjovsky17a.html

About the Authors

Yao-San Lin is an associate professor of the Department of Industrial Engineering and Management at National Chin-Yi University of Technology, Taiwan, and has taught statistical analysis, information management, and machine learning for many years. In his past ten years of research, he won the IEEE Best Paper Award (2019 and 2013), and his research results have been published in international academic journals: Applied Soft Computing, Decision Support Systems, Neurocomputing, European Journal of Operational Research, International Journal of Production Research, Computers & Operations Research, and Expert Systems with Applications. His current research interests are mainly in the fields of manufacturing and natural language processing. (yslin@ncut.edu.tw)

Mei-Ling Huang received her M.S. and Ph.D. degrees in industrial engineering from the University of Wisconsin–Madison and National Chiao Tung University, respectively. Currently, she is affiliated with the Department of Industrial Engineering and Management at National Chin-Yi University of Technology. Her research interests include quality management, quality engineering, data mining, and medical diagnosis. (huangml@ncut.edu.tw)

Der-Chiang Li is a distinguished professor at the Department of Industrial and Information Management, National Cheng Kung University, Taiwan. He received his Ph.D. degree from the Department of Industrial Engineering at Lamar University Beaumont, Texas, USA, in 1985. As a research professor, his current interest focuses on machine learning with small datasets. His articles have appeared in Decision Support Systems, Omega, Information Sciences, European Journal of Operational Research, Computers & Operations Research, International Journal of Production Research, and other publications. (lidc@mail.ncku.edu.tw)

Jui-Yu Yang earned his M.S. degree in 2023 from the Department of Industrial and Information Management, National Cheng Kung University, Taiwan. His research interests include small dataset forecasting and machine learning. (othellin@gmail.com)