scientific reports

Check for updates

OPEN

A multi-classification deep neural network for cancer type identification from high-dimension, small-sample and imbalanced gene microarray data

Yifu Zeng[1,2], Yixiang Zhang[3], Zikai Xiao[1] & He Sui[4,5]✉

Gene microarray technology provides an efficient way to diagnose cancer. However, microarray gene expression data face the challenges of high-dimension, small-sample, and multi-class imbalance. The coupling of these challenges leads to inaccurate results when using traditional feature selection and classification algorithms. Due to fast learning speed and good classification performance, deep neural network such as generative adversarial network has been proven one of the best classification algorithms, especially in bioinformatics domain. However, it is limited to binary application and inefficient in processing high-dimensional sparse features. This paper proposes a multi-classification generative adversarial network model combined with features bundling (MGAN-FB) to handle the coupling of high-dimension, small-sample, and multi-class imbalance for gene microarray data classification at both feature and algorithmic levels. At feature level, a deep encoder structure combining feature bundling (FB) mechanism and squeeze and excite (SE) mechanism, is designed for the generator. So, the sparsity, correlation and consequence of high-dimension features are all taken into consideration for adaptive features extraction. It achieves effective dimensionality reduction without transitional information loss. At algorithmic level, a softmax module coupled with multi-classifier are introduced into the discriminator, with a new objective function is distinctively designed for the proposed MGAN-FB model, considering encode loss, reconstruction loss, discrimination loss and multi-classification loss. We extend generative adversaria framework from the binary classification to the multi-classification field. Experiments are performed on eight open-source gene microarray datasets from classification performance, running time and non-parametric tests, which demonstrate that the proposed method has obvious advantages over other 7 compared methods.

Keywords Cancer diagnosis, Gene microarray data, High dimensional, Low-sample-size, Multi-class imbalance

Accurate identification of cancer types contributes to the correct treatment of cancerous tumors. Gene microarray data provide advanced biological basis for cancer diagnosis, which is a more efficient way to diagnose cancer, compared with traditional ones based on morphological and clinical appearance[1]. The microarray technology based on gene expression gives more accurate results as the cancer is fundamentally a malfunction of genetics[2]. So, it has attracted a lot of attentions and various classification methods have been proposed to identify cancers based on gene microarray data[3].

For multi-class classification tasks of gene microarray data, there are three main types of machine learning (ML) methods. The first type was based on traditional ML methods. One-versus-One (OVO), One-versus-Rest (OVR) and multiple binary classifiers were all tried to deal with this problem[4]. Their strategy is to transform the multi-classification problem into some binary classification problem, which is inefficient and unstable. The

[1]Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou, China. [2]Department of Information Technology, The Second Affiliated Hospital of Fujian Medical University, Quanzhou, China. [3]Department of Infectious Diseases, The Second Affiliated Hospital of Fujian Medical University, Quanzhou, China. [4]College of Aeronautical Engineering, Civil Aviation University of China, Tianjin 300300, China. [5]Information Security Evaluation Center, Civil Aviation University of China, Tianjin 300300, China. ✉email: hsui@cauc.edu.cn

second type was based on swarm intelligence algorithms, such as Particle Swarm Optimization (PSO), Firefly, Flower Pollination Optimization (FPO), Elephant Herding Optimisation (EHO) and Cuckoo Search (CS)[5,6]. They were always used with Gaussian Mixture Model (GMM) and well known for their efficiency and effectiveness as global search agents. However, their scalability for high-dimensional datasets is also challenging. In recent years, various architectures of deep learning (DL) were applied, including fully connected neural networks (also known as multi-layer perceptron NN, or MLP), convolutional neural networks (CNN), recurrent neural networks (RNN), graph neural networks (GNN), transformers neural networks (TNN)[7], and extreme learning machine (ELM)[1]. Despite the recent progress in DL-based cancer classification, various problems remain to be addressed, including high-dimension, small-sample and class imbalance.

The characteristics of gene microarray data, such as high-dimension, small-sample and multi-class imbalance, pose serious challenges to its classification and analysis[8]. Humans have more than 30,000 genes, which means that the gene expression microarray data has tens of thousands of features. We must select hundreds, dozens, or even a few genes associated with a certain cancer from these genes, to guide the early diagnosis of cancer. Undoubtedly, it's a great challenge for traditional methods, that is, curse of dimensionality[9]. Especially, when the sample size n and the feature dimension d satisfy the relation of $n << d$, classical statistical methods encounter the performance degradation for classification[10], which is a coupled problem high-dimension and small-sample. More unfortunately, multi-class imbalance of gene microarray data further increases the complexity and extremity[11]. These factors are always coexisting, making cancer type identification based on gene microarray data as an acknowledged challenging issue.

Data preprocessing including feature selection and resampling and algorithm optimization are common methods to deal with high-dimensional, small-sample and multi-class imbalance problem. Feature selection is usually applied in gene microarray data evaluation[12]. However, when feature selection is carried out, class imbalance is usually not considered, and the selected features may be biased towards the majority category while ignoring the discrimination ability of the minority category. Furthermore, small-sample and imbalance often bring features overlap problem, which makes classifiers more difficult to deal with. Resampling can balance the dataset by generating a few minority samples or deleting most of the majority samples, so that the classifier can get better training[13]. Obviously, in the case of small-sample of the gene microarray data, oversampling methods are mainly used. However, when faced with multi-class imbalance, the number of samples to be generated is huge, and different sampling methods are needed for different minority class to achieve high quality sampling, both of which make it low efficiency. In addition to data preprocessing, some algorithm-level approaches have also been proposed to make the classifier more focused on feature learning of minority samples through some special mechanisms such as cost sensitivity[14], or ensemble learning[15]. However, most of these methods are optimized for specific data sets, have limited scope of application, insufficient generalization ability, and are dependent on data quality.

It is urgent to propose a more robust method with more comprehensive advantages in features, samples and algorithms to realize cancer classification and diagnosis based on gene microarray data. Deep neural network auxiliary diagnosis technology has gradually become a research hotspot, with its adaptive processing ability[16]. The most representative deep learning method in imbalanced data learning is the generative adversarial network[17]. It carries out model training and parameter updating through the antagonistic game between the generator and the discriminator, so that the generator can generate samples more in line with the real distribution, and the discriminator can be more sensitive to anormal samples, which is suitable for small-sample and imbalance problem. However, the existing GAN models are always inefficient when dealing with high-dimensional features. Meanwhile, due to the restriction of discriminator mechanism, GAN model is mainly used in the field of binary classification, and cannot solve the multi-classification problem in gene microarray data. Therefore, a multi-classification generative adversarial network model combined with features bundling (MGAN-FB) is proposed in this paper, to improve the classification performance for high-dimension, small-sample and multi-class imbalance gene microarray data to assist cancer type identification. The main contributions are as follows:

(1) We propose a multi-classification framework derived from generative adversarial network. It consists of two encoders in generator, and one multi-classifier in discriminator, which puts forward an effective model for high-dimension, small-sample and imbalanced gene microarray classification.

(2) A deep encoder structure combining feature bundling (FB) mechanism and feature squeeze and excite (FSE) mechanism, is designed for the generator. So, the sparsity, correlation and consequence of high-dimension features are all taken into consideration for adaptive features extraction. It achieves effective dimensionality reduction without transitional information loss.

(3) A softmax module coupled with multi-classifier are introduced into the discriminator, which realizes the integration of discriminant and classification. It extends GAN framework from the binary classification to the multi-classification field, solving multi-class imbalance problem of gene microarray data classification.

(4) A new objective function is distinctively designed for the proposed GAN model. The encode loss of feature compression and the reconstruction loss of data generation are considered in the generator. Besides, the discrimination loss and data classification loss are considered in the discriminator. It realizes the accurate multi-classification under imbalanced situation.

The remainder of this article is organized as follows. In Sect. 2, we review the related works. In Sect. 3, we outline the proposed MGAN-FB model in detail. In Sect. 4, we present the experimental methodology including benchmarked datasets, evaluation metrics and experimental design. Additionally, in Sect. 5, we report on and analyze the experimental results. Finally, we conclude this paper and look forward to the future work in Sect. 6.

Related works

Feature selection methods

Feature selection is to select optimal feature subset, and discard irrelevant and redundant features to reduce data complexity for better classifier training. It can be divided into three main types: filter based, wrapper based, embedded and deep-learning methods.

Filter based methods only use the characteristics of the sample itself without the need to rely on subsequent models. The main idea is to use the evaluation function to calculate all the feature scores, then rank the scores from smallest to largest, and then intercept the top features. The evaluation criteria are generally the difference, correlation or information entropy among features. Difference evaluation criteria believes that the greater the difference, the higher the distinguishability of labels[18]. Method depending on correlation is generally driven from the correlation coefficient between each feature. And features with higher coefficient would be kept for model training[19]. Information entropy can measure the importance of a feature for a certain class label from the perspective of information theory. The typical method is to calculate the information gain of the feature column and label column[20].

Wrapper based methods consist of a search algorithm and a classifier, which is to embed the feature selection process into the subsequent classifier learning. Common search algorithms are sequential forward/ backward selection (SFS/SBS)[21]. SFS starts with an empty set of features and adds the optimal features to the set one by one. While SBS starts with all features and progressively removes the worst remaining feature sequences. Biological swarm intelligence algorithm[22] is also a typical wrapper-based method. It simulates the gathering behavior of organisms such as birds or fish, and searches for the optimal solution by constantly adjusting the position and speed of particles. Besides, annealing algorithm[23] often have a better selection effect, however, the time complexity is usually high, especially for datasets with high-dimension features.

Embedded method is a compromise between filtering method and packaging method, whose main idea is to integrate feature selection process and model learning process[24]. For embedded methods, the feature selector can usually achieve better learning accuracy. However, they were always tied to specific classifiers to deal with imbalance and multiple classification problems, which is not flexible enough[25]. With the increasing expansion of genetic data, deep learning method plays a more important role in feature selection, for its better ability to describe the implicit relationship between features, such as fDNN[26], DNP-AAP[27] and forge net[28]. fDNN[26] is composed of two parts: multiple decision trees (forest) to obtain features with higher weight, and DNN (deep neural network) for classification. DNP-AAP[27] is also composed of two parts, namely DNP and AAP. DNP makes it possible to select features, and AAP is to evaluate the importance of the selected features. Forge net[28] is a forest graph embedded deep neural network model, which is an improvement of fDNN. The graph model mapped high-dimensional data into sparse space, which solves the problem sparsity well, while increases the utilization of computer memory space to a certain extent.

Multi-class imbalance process methods

Two main types of approaches have been tried to solve multi-class imbalance problem: data-level approaches and algorithm-level approaches[29]. The former focused on preprocessing to rebalance data[30], whereas the latter provided guidance for classifiers to bias toward the minority samples[31].

Clearly, how to rebalance the training dataset is the key issue in multi-class imbalance problem. In general, resampling approaches have well coped with imbalance, by improving the minority component ratio of the dataset. Considering the particularity of the small-sample gene microarray data, oversampling method is more suitable. The most classic oversampling method is the synthetic minority oversampling technique (SMOTE) with linear inter-potation. Then, several variants have been developed to overcome degeneration associated with noises by weighted cluster, such as noise-immunity majority weighted minority oversampling technique (NI-MWMOTE)[32]. Moreover, Zhu et al.[33] divided the samples using position characteristic-aware interpolation algorithm, and different interpolation strategies were given to different class. Navarro et al.[34] proposed a dynamic oversampling procedure in a memetic algorithm that uses neural networks. First, the examples of the minority classes are oversampled to partially balance the classes. Then, the memetic algorithm is applied to oversample the data and generate new patterns for the class with the least sensitivity. Wang and Yao[35] studied the effect of multi-minority and multi-majority classes on the learning process, and additionally explored AdaBoost.Nc[36] with imbalanced, multi-class datasets. Their method used a negative correlation learning algorithm that utilizes an ambiguity term to add explicit diversity.

Recently, attention is drawn to the more challenging case of algorithm design learning from imbalanced, multi-class data. Xiao et al.[37] applied deep learning to an ensemble framework that incorporates multiple different machine learning models. Data were selected by differential gene expression analysis and a deep learning method was employed to ensemble the outputs of the classifiers. Qi et al.[38] integrated Adaboost with deep support vector machine. Adaboost is applied to select SVMs with the minimal error rate and the highest diversity. By stacking SVMs into layers, the method acquires a new set of deep features. Rasti et al.[39] employed a mixture ensemble of convolutional neural networks for breast cancer detection. Each convolutional neural network is a modular and image-based ensemble, which stochastically partition the image space through simultaneous and competitive learning.

Generative adversarial network-based classification methods

In recent years, Generative Adversarial Network (GAN)-based methods have been developed for small-sample data classification, given their better ability to generate diversiform samples. Similarly, GAN can also be used to solve the problem of insufficient samples and unbalance from the two levels of samples and algorithms.

For sample level, the generation mechanism of GAN can supplement high-quality minority samples. Gayathri et al.[40] used GAN with auxiliary information to further improve the diversity of generated samples.

Engelmann et al.[41] employed Wasserstein distance into conditional GAN model to generate data of the specified class, optimizing the classification performance on strongly nonlinear datasets. Zheng et al.[42] further introduced the penalty coefficients into GAN, which showed greater advantages in terms of model stability. Dlamini and Fahim[43] put forward a conditional GAN model with KL-divergence. This method not only guided learning toward the minority class, but also overcame gradient vanish.

GAN models can also be used for direct classification due to their strong learning ability of sample distribution and deep interaction features. Donahue et al.[44] introduced the encoder into the basic GAN model and built the BiGAN (Bidirectional GAN). This bidirectional generation mechanism enables the generator to learn the inverse mapping from hidden space to real data space and attach inverse mapping labels to the original data. Zenati et al.[45] proposed a more Efficient Gan-based model, EGBAD (Efficient Gan-based Anomaly Detection), and proved that this method is not only applicable to image data, but also to network security data. Aiming at the discrete data classification, Chen and Jiang[46] proposes a model based on BiGAN and the full-connected network of dropout to calculate the discriminant loss by weighted summation of residual loss. Jiang et al.[47] proposed a GAN model for imbalanced data. The scoring function is composed of both apparent loss and potential loss, so as to realize fault classification of rolling bearing data. Li et al.[48] combined GAN with LSTM and RNN to proposed a multivariate method, MAD-GAN (Multivariate Detection with Generative Adversarial Networks), which considered the interaction between hidden spatial features of the dataset. The existing GAN methods have achieved remarkable results in the classification of imbalanced data. However, they can only be used for binary detection, and the research on multi-classification of small-sample and imbalanced data needs to be further developed.

The proposed MGAN-FB method

Motivation

Studies have shown that high-dimensional, imbalanced and multi-classification coupling is a great challenge for cancer type identification based on gene microarray data. High dimension is difficult problem for feature selection, and imbalance is a problem for data the distribution, both of which are sample-level problems. However, multi-classification is a problem for model-level. In order to solve this comprehensive problem well, we need to consider from different aspects and put forward a more comprehensive framework.

The existing solution adopt a divide-and-rule and simply-splice strategy, with feature-selection for high-dimensional, resampling for imbalance, and OvO or OvA schemes for multi-classification. As a result, there is still a lack of effective comprehensive method for the coupling problem. Although some of the above methods have achieved certain results, the stability of classification is not good. The effect is better for some datasets, however, it may be poor for other ones. The reason is that there may be contradictions or conflicts in the mechanism of this splicing solution. For instance, the sample distribution is not considered when feature selection is used for dimensionality reduction, which may lead to increased imbalance and even class overlap. In addition, without considering the imbalance problem in multi-classification may exacerbate the imbalance to higher level and further aggravate the small-sample problem.

The root cause of the aforementioned shortage is that the existing solutions are not under a unified architecture, with the multifaceted nature of the problem and the complexity of the unified architecture. In recent years, with the development of deep learning frameworks, their heuristic learning capabilities can well handle multiple difficult problems at the same time. Its adaptive learning of features avoids the dependence on researchers' experience. The generation mechanism can also overcome the problem of imbalance and small-sample effectively. A simple optimization of the output layer will handle multiple classification problem easily.

Therefore, aiming at the multi-faceted coupling problem faced by cancer type identification based on gene microarray data, this paper intends to design a deep learning framework with more comprehensive performance advantages.

MGAN-FE model design

Framework of MGAN-FE

Figure 1 illustrates the framework of our proposed MGAN-FE method, which contains two sub-networks: generator and discriminator.

In the generator, we use the "encoder + decoder + encoder" architecture, so it consists of a feature bundling module, two encoders, and a decoder. The generator learns the input data representation and reconstructs the input data via the use of an encoder and a decoder network, respectively. The generator G first reads an input data x, where $x \in \mathrm{R}^{w \times h \times c}$, and forward-passes it to feature bunding module FB and encoder network EN_1. EN_1 downscales x by compressing it to a vector $E(x|c)$, where $E(x|c) \in R^d$. z is also known as the bottleneck features of G and hypothesized to have the smallest dimension containing the best representation of x. The decoder network is to upscales the vector $E(x|c)$ to reconstruct the data x as x'. Based on these, the generator network G generates data x' via $x'=DE(E(x|c))$.

The discriminator adopts the architecture of "coding + multi-classifier + softmax" to realize the combination of identification sample generation and multi-classification. To generate a specific class of samples in a directional manner, the input is not only a data x, but also with its class label c as an auxiliary condition, so as to finally generate a specific sample (x', c). Essentially, it is a variant of the CGAN model with adversarial learning by a generator and discriminator. As stated before, the two core models in GAN are the generative model G and the discriminative model D. D can be treated as a binary classifier, which only has ability to judge whether the sample is from the real dataset or not. It does not have ability to further predict the classification of real samples. Additionally, most of classifiers for detection usually belong to the supervised learning, so we need to reconstruct a supervised learning framework based on GAN. In order to supply more information for the multiclass classifier, we take the output of the generative model G as the input of the classifier together with

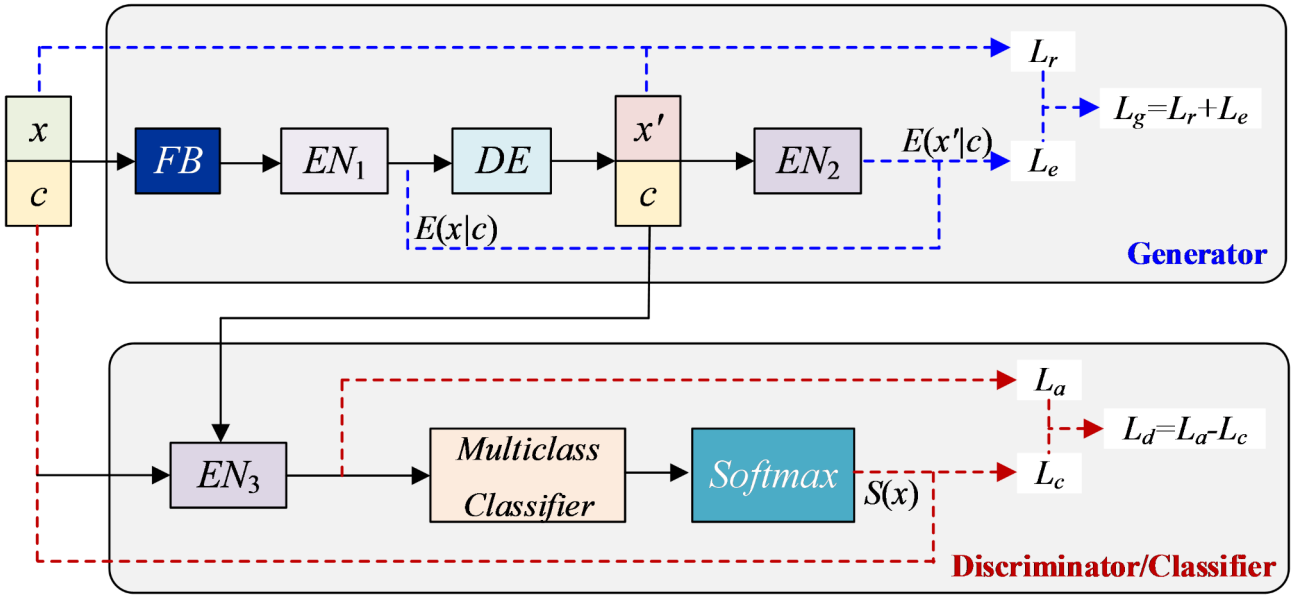

Fig. 1. Framework of the proposed MGAN-FB model.

the original training set. To improve the efficiency of the framework and further simplify the framework, we replace the discriminative model *D* with an encoder and a multiclass classifier. In this way, the classifier does not only undertake the task of classification, but also serves as the role of the discriminative model *D* to determine whether the sample is from *G* or the real dataset. The pseudo-code for the algorithm is shown as follows:

Input: Training set ***x_train***, epochs ***n***.
Initialize: Initialize all parameters of MGAN-FB and Multiclass classifier ***C***.
Output: Trained MGAN-FB model.
Training:
1: **for** $1<i<30$
2: ***x*/*c*** ← Input ***x_train*** into data processer with its class label.
3: Perform dimension reduction for ***x*** with the Feature Bundling Module.
4: $E(\boldsymbol{x}/\boldsymbol{c})$ ← Input ***x*/*c*** into encoder1 EN_1.
5: $\boldsymbol{x}'/\boldsymbol{c}$ ← Input $E(\boldsymbol{x}/\boldsymbol{c})$ into decoder DE.
6: $E(\boldsymbol{x}'/\boldsymbol{c})$ ← Input $\boldsymbol{x}'/\boldsymbol{c}$ into encoder1 EN_2.
7: Calculate encoding loss $L_e = E_{x:\, p_x}\left\|E(x|c) - E(x'|c)\right\|_2$.
8: Calculate generation loss $L_r = E_{x:\, p_x}\left\|x - G(x|c)\right\|_1$.
9: Calculate generator loss $L_g = E_{x:\, p_x}\left\|E(x|c) - E(x'|c)\right\|_{2e} + E_{x:\, p_x}\left\|x - G(x|c)\right\|_1$.
10: Input ***x*/*c*** and $\boldsymbol{x}'/\boldsymbol{c}$ into discriminator.
11: Calculate discrimination loss $L_a = E_{x:\, p_x}\left\|f(x|c) - E_{x:\, p_x} f(G(x|c))\right\|_2$.
12: Calculate classification loss $L_c = \frac{1}{N}\sum_{i=1}^{N} \log p_{model}(y = c|x_i)$.
13: Calculate discriminator loss $L_d = E_{x:\, p_x}\left\|f(x|c) - E_{x:\, p_x} f(G(x|c))\right\|_2 - \frac{1}{N}\sum_{i=1}^{N} \log p_{model}(y = c|x_i)$.
14: **if** $i \geq 500$
15: break and go to line 18
16: **else** update network parameters using Adam optimizer, i++ and go to line 1.
17: **end for**
18: Return the trained parameters of MGAN-FB model

Algorithm 1 . Anomaly detection based on MGAN-FB.

Feature bundling module

High-dimension is a significant characteristic of gene microarray data. Subsequent deep encoder can adaptively extract features from the perspective of correlation, so as to achieve dimensionality reduction. However, there are still large numbers of features whose values is 0 in gene microarray data, which are called sparse features. While the others whose values are not 0 in all samples are called dense features. The subsequent deep encoder network is undoubtedly inefficient for processing sparse features, and it also increases the learning error. Therefore, we designed a FB module in the generator to process these sparse features in advance, facilitating subsequent learning and classification.

Obviously, we want to only deal with sparse features and keep dense features. The idea is to combine sparse features into one or a few dense features. Specifically, the sparse features are combined according to certain rules, that is, bundling, which is somewhat similar to the inverse process of one-hot, as shown in Fig. 2.

To avoid the impact of different feature value ranges on bundling, we need to normalize all feature value ranges first. As shown in Fig. 2, both Feature 4 and Feature 5 are normalized to the value range of 0–10. Then, when some sparse features are bundled, we adopt the downward binding principle. For example, when Data 1 is performed of feature binding, it is bundled from Feature 2 to Feature 1, with a new feature value $10 + 3 = 13$. Similarly, when Data 2 performs feature binding, it is bundled downward from the Feature 3, and its feature value is $20 + 5.2 = 25.2$. In this way, when binding is carried out, the samples are dispersed to the greatest extent in the same feature dimension, and the sample overlap caused by dimension reduction can be avoided.

It should be noted that when we perform bundling operations, there may be two features with non-zero eigenvalues. For example, Data 3 has non-zero values on both Feature 2 and Feature 3, which is defined as feature conflict. Conflicts are inevitable because it is difficult to find a feature that has a value of 0 on all samples. In the case of completely excluded conflicts, there are very few features that can be combined, so we will presuppose a feature conflict ratio threshold. When the conflict ratio does not exceed the threshold, we ignore the conflict and bundle them together. While if the conflict ratio exceeds this value, they cannot be bundled into one new feature.

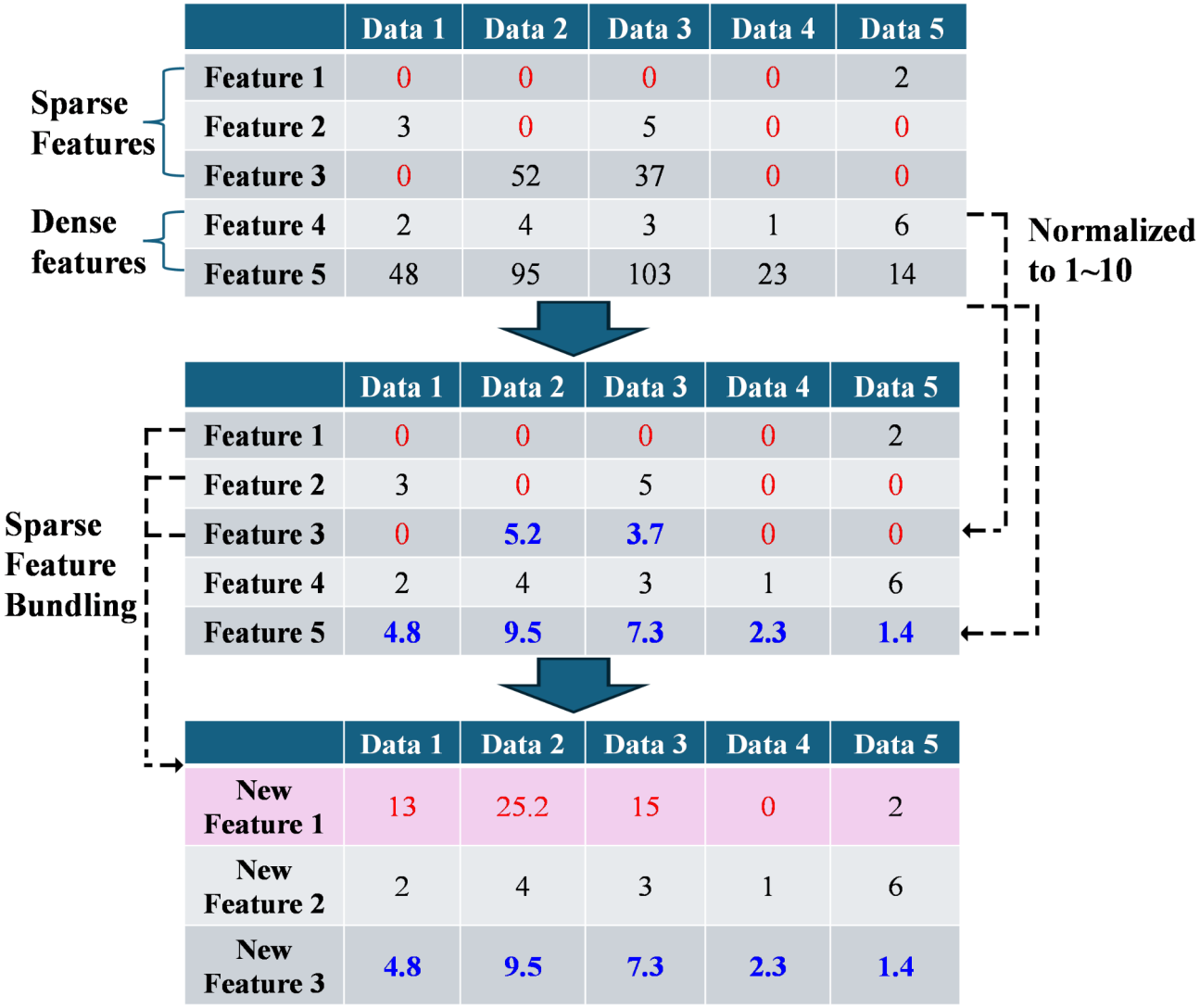

Fig. 2. Illustration of feature bundling principle.

We generally set the threshold as 1/1000, which means that the proportion of two features with non-0 values in all samples does not exceed 0.1%. This threshold can be optimized based on sample size. In this case, the value with the largest value among the two features that are not 0 is taken as the reserved value when bundling is conducted. For example, in Data 3, the value of Feature 2 is 5 and the value of Feature 3 is 3.7. We retain 5 and bundle it to Feature 1, so the eigenvalue after bundling is $10 + 5 = 15$.

Based on the above principles, we also need to find all exclusive features to perform the binding operation, which is a NP problem. Therefore, through the greedy algorithm, we only need to perform one feature traversal to achieve the binding of all sparse features.

Common sparse optimization needs to save non-zero value table, while after bundling, multiple sparse features are bundled into fewer dense features without non-zero value table, which saves memory and time consumption. At the same time, when the traversal is carried out among multiple sparse features, there is a low cache miss problem in each feature switching. After merging into one feature, the number of feature switching is reduced. In addition, the binding not only solves the problem of sparse features, but also realizes dimension reduction.

It is also obvious that we cannot restore the original features after bundling, because we cannot mark that the merged feature values are from those original features. It means that we lose some information when operating sparse feature bundling, which is inevitable for all feature extraction and dimensionality reduction. This part of the loss will be considered in the sample reconstruction. Besides, it should be noted that feature bundling is only applicable to numerical features and cannot be used directly for non-numerical ones. Fortunately, gene microarray data is precisely a numeric array, so feature bundling can be used directly.

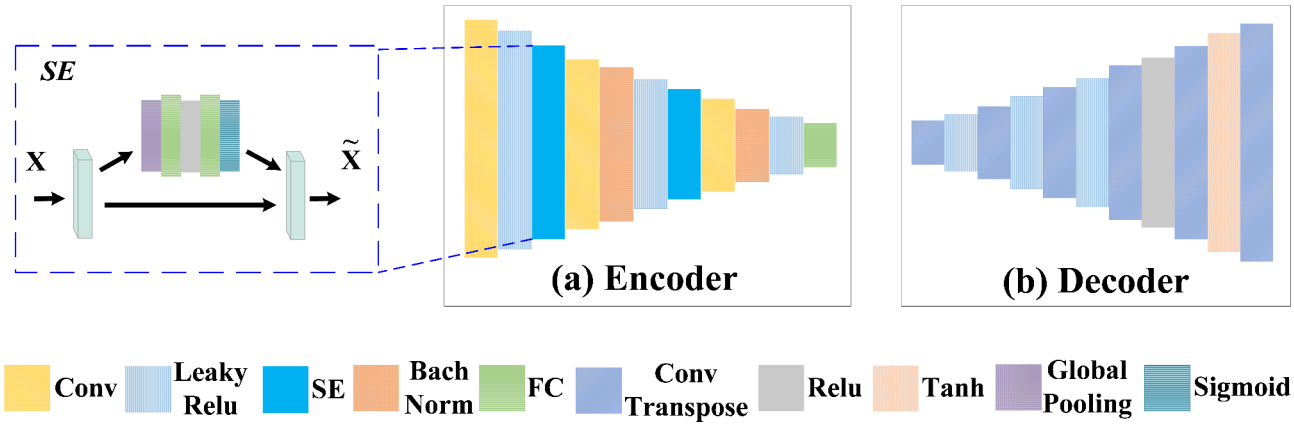

Fig. 3. Subnet structure of the encoder and decoder.

Operation	Kernel	Strides	Feature Maps/Units
Conv/Conv Transpose	3×3	2×2	128
FC Input-output	128 – 12		
Optimizer	Adam ($\alpha = 10^{-5}$, $\beta_1 = 0.6$)		
Batch size	10		
Leaky ReLU slope	0.1		
Weight, Bias Initialization	Isotropic gaussian ($\mu = 0$, $\sigma = 0.01$)		

Table 1. Main parameters of the encoder and decoder.

Encoder and decoder module

Not all of features are associated with a particular type of cancer, and we want to find the ones that were more important for a particular type of cancer. Encoder can automatically mine deep feature interactions to help us find the most important features for a certain cancer expression, which is essentially a heuristic adaptive feature selection.

The generator consists of two encoders and one decoder. The first encoder plays the role of dimensionality reduction to solve the problem of high-dimensional features, and the decoder generates specific class of samples by restoring features to solve the problem of imbalance and small-sample. The second encoder is to extract the depth feature of the generated sample, and its main function is to evaluate the quality of the generated sample. As for decoder, is has only one encoder. The subnet structure of the encoder and decoder is shown as Fig. 3.

As shown in Fig. 3(a), encoders are designed with 11 layers, including convolution (Conv), Bach Norm, Leaky Relu, squeeze and excite (SE), and full connection (FC). The convolution layer ensures that the connection can extract the features more effectively, and map low- to high-dimensional space for oversampling operation. Hence, it can improve the resolution of the model to the feature, and achieve higher accuracy through learning. The Bach Norm accelerates the convergence rate of the model and effectively avoids gradient disappearance. The scaling factor within the Bach Norm can effectively identify neurons that contribute little to the network, and some neurons can be automatically weakened or eliminated after the activation function. The Leaky Relu is used as the activation function, which will count the part of the input that is less than 0. Then the sawtooth problem in the gradient direction is avoided in the backpropagation process. The squeeze and excite (SE) block adopt feature recalibration strategy by feature compression, excitation and reweighting. It is of five-layer network structure, including global poling layer, two fully connected layers, Relu layer, and sigmoid layer. So as to the weight of effective features is increased, and the weight of invalid or small effect is reduced. By integrating feature representation into a value, it reduces the influence of feature location on the classification results, and improves the robustness of the entire network. As a result, MGAN-FE with encoders can learn the characteristics of sample data in the feature space, simplify the data representation, and then obtain effective patterns, further improving the generation ability. The second encoder network that compresses the data x' that is reconstructed by the network G. With different parametrization, it has the same architectural details as EN_1.

The subnet structure of the decoder is a multi-layer. And for better reconstruction of sample data, three types of activation function are used, including Leaky Relu, Relu and Tanh. The dimension of the vector $E(x'|c)$ is the same as that of $E(x|c)$ for consistent comparison. Unlike the prior autoencoder-based approaches, in which the minimization of the latent vectors is achieved via the bottleneck features, this sub-network explicitly learns to minimize the distance with its parametrization. During the test time, moreover, the detection is performed with this minimization. A standard multiclass classifier usually takes a sample x as input, and outputs a vector that can be turned into one of the possible class probabilities by applying the softmax function. In the supervised learning, such a model is then trained by minimizing the cross-entropy between the real labels and the predictive distribution to obtain the optimal parameters. The main parameters of the encoder and decoder is as follows.

For gene microarray data, convolutional layers have three functions. It is as a filter to perform noise reduction, which makes raw data smoother. In addition, it is also a "compressor" to reduce the dimensions of the gene microarray data, dealing with the dimension problem together with the feature binding module. Besides, the more important role is to capture the local characteristics correlation of the gene microarray data to. Because researches have shown that local expression relationships among gene microarray data may also contain cancer-related information[49].

After the introduction of adversarial training for intrusion detection, the generative model can continually generate 'fake' samples from a random distribution. In the adversarial training, the multiclass classifier identifies whether the sample is normal, or fake, or any one of the other classes, while the generative model dynamically adjusts the strategy for generating more similar fake samples according to the feedback (fake or real) from the multiclass classifier. Thus, the framework can train the classifier together with new augmented training set, which includes original class labeled samples and constantly generated new 'fake' samples.

MGAN-FB model training

To realize multiple classification for gene microarray data, the designed MGAN-FE model needs to be further trained. The key of this training is to determine the target, that is, the loss function of MGAN-FE model. We hypothesize that when a data is forward-passed into the network G, it is not able to reconstruct the abnormalities even though encoder manages to map the input x to the latent vector $E(x|c)$. This is because the network is modeled only on samples of certain class during training and its parametrization is not suitable for generating samples of other classes. An output that has missed other classes can lead to the encoder network mapping it to a vector that has also missed certain feature representation, causing dissimilarity between them. When there is such dissimilarity within latent vector space for an input data, the model classifies them as a certain class. To validate this hypothesis, we formulate our objective function by combining four loss functions, each of which optimizes individual sub-networks.

According to the design of the generator, there are two main losses, namely, encoding loss L_e in the process of feature extraction (dimensionality reduction) and generation loss L_r in the process of decoding and reconstruction of new samples. The two losses introduced above can enforce the generator to produce data that are not only realistic but also contextually sound. Moreover, we employ an additional encoder loss to minimize the distance between the bottleneck features of the input and the encoded features of the generated sample. Encoder loss is formally defined as:

$$L_e = E_{x\sim p_x}\left\|E(x\,|c) - E(x'\,|c)\right\|_2 \tag{1}$$

The reconstruction loss is adequate to fool the discriminator D with generated samples. However, with only an adversarial loss, the generator is not optimized towards learning contextual information about the input data. It has been shown that penalizing the generator by measuring the distance between the input and the generated data remedies this issue. Isola et al.[50] showed that the use of L_1 yields less blurry results than L_2. Hence, we also penalize G by measuring the L_1 distance between the original x and the generated samples using a contextual loss defined as:

$$L_r = E_{x\sim p_x}\|x - G(x\,|c)\|_1 \tag{2}$$

In summary, the total loss of the generator is:

$$L_g = E_{x\sim p_x}\left\|E(x\,|c) - E(x'\,|c)\right\|_{2e} + E_{x\sim p_x}\|x - G(x\,|c)\|_1 \tag{3}$$

The function of discriminator is to combine the two tasks of discriminating generated samples and multi-classification. The difference in distribution between the initial sample and the generated sample is the identification adversarial loss L_a. Original sample class label and multiple classifiers get label differences for classification loss is L_c.

Discrimination loss

We use feature matching loss for adversarial learning. Proposed by Salimans et al.[51], feature matching is shown to reduce the instability of GAN training. Unlike the original GAN where G is updated based on the output of D, here we update G based on the internal representation of D. Formally, let f be a function that outputs an intermediate layer of the discriminator D for a given input drawn from the input data distribution, feature matching computes the L_2 distance between the feature representation of the original and the generated data, respectively. Hence, our discrimination loss is defined as:

$$L_a = E_{x\sim p_x}\|f(x\,|c) - E_{x\sim p_x} f(G(x\,|c))\|_2 \tag{4}$$

Classification loss

It is assumed that (x', c) is a sample from the training set that contains a k-class labels, where $c \in \{c_1, c_2, \ldots, c_k\}$. The generative model generates 'fake' samples (x_f, c_f), where c_f ='fake'. The samples (x, c) and (x', c) are synthetic data samples and generated samples, where the label y contains k classes. For the multiclass classification problem, the classifier inputs a sample x and outputs the classification probabilities for the k classes p_i (i = 1, 2, …, k). Assuming that p is the real probability distribution of the sample and q is the predicted probability distribution of the classifier, the cross-entropy for a given dataset x is defined as:

$$CE(p,q) = -\sum_{x \in X} p(x) \log q(x) \tag{5}$$

The value of Eq. (5) indicates the error between the real classification and the predicted classification. The smaller the value is, the closer the predicted probability distribution is to the real probability distribution, and the more accurate the predicted result will be.

Under the multiclass classification task, the loss function is usually defined as cross-entropy loss. Let $y_{x_i}^j$ represent the real probability distribution of the sample x_i, and let $P_{model}(y=j|x_i)$ represent the predicted probability distribution of the sample x_i, then the corresponding loss function can be defined as:

$$L_c = -\sum_j y_{x_i}^j \log p_{model}(y = j\,|x_i\,), \qquad j \in p_i \tag{6}$$

For dataset X, which is synthetic data samples and generated samples, the corresponding loss function is defined as:

$$L_c = -\frac{1}{N}\sum_{i=1}^{N}\sum_j y_{x_i}^j \log p_{model}(y = j\,|x_i\,), \qquad j \in p_i \tag{7}$$

After one-hot coding, the real category of the sample $y_{x_i}^j$ is mapped into a K-dimension vector. For example, If the sample x_i belongs to category c, then $y_{x_i}^{j=c} = 1$. Besides, all the values of the remaining columns are 0, that is, $y_{x_i}^{j\neq c} = 0$. Therefore, the loss function of the multiclass classifier in the proposed framework can be further expressed as follows:

$$\begin{aligned} L_c &= \frac{1}{N}\sum_{i=1}^{N}[y_{x_i}^{j=c} \log p_{model}(y = c\,|x_i\,) + \sum_{j\neq c} y_{x_i}^j \log p_{model}(y = j\,|x_i\,)] \\ &= \frac{1}{N}\sum_{i=1}^{N}[y_{x_i}^{j=c} \log p_{model}(y = c\,|x_i\,)] \\ &= \frac{1}{N}\sum_{i=1}^{N}\log p_{model}(y = c\,|x_i\,) \end{aligned} \tag{8}$$

In summary, the total loss of the discriminator is:

$$L_d = L_a - L_c = E_{x\sim p_x}\|f(x\,|c) - E_{x\sim p_x} f(G(x\,|c))\|_2 - \frac{1}{N}\sum_{i=1}^{N}\log p_{model}(y = c\,|x_i\,) \tag{9}$$

Because the greater the classification difference, the smaller the discriminator loss, the negative sign is used for L_c. The generator and discriminator are optimally balanced through a binary zero-sum game, therefore, the objective function of MGAN-FE is:

$$\begin{aligned} &\min_G \max_D V(D,G) = \min(L_g - L_d) \\ &= \min[E_{x\sim p_x}\left\|E(x\,|c) - E(x'\,|c)\right\|_{2_e} + E_{x\sim p_x}\|x - G(x\,|c)\|_1 \\ &+\frac{1}{N}\sum_{i=1}^{N}\log p_{model}(y = c\,|x_i\,) - E_{x\sim p_x}\|f(x\,|c) - E_{x\sim p_x} f(G(x\,|c))\|_2] \end{aligned} \tag{10}$$

About the stopping criterion for training, we set an epoch value of the training cycle, and generally the model stops training when the preset epoch is finished. However, in order to further improve the training efficiency and avoid overfitting caused by small-sample, we adopted the early stopping strategy. The dataset is divided into training set, validation set, and test set. Considering the small number of samples and the imbalance problem, the training set used all the data, while the verification set and the test set only selected 20% from raw dataset based on the proportion of each class. The criterion of early stopping is based on the average classification accuracy of all classes. When the accuracy increases less than 1% for 5 consecutive iterations, the training should be stopped.

As for avoiding the network overfitting, strategies such as early stopping, data set amplification, regularization and dropout are adopted generally. According to the specific situation of this paper, it mainly adopts the strategies of early stopping combining with dropout. The early stopping strategy is described above. For dropout strategy, it randomly ignores 40% of hidden layer nodes in each training epoch to avoid training the same nodes repeatedly.

Datasets	#Samples	#Classes	#Features	IR	Classes
Brain Tumor1	90	5	5920	15	Medulloblastoma (Me), Malignant Glioma (NG), AT/RT, Normal Cerebellum (NC), PNET
Brain Tumor2	50	4	10,367	2.14	Classic Glioblastomas (CG), Classic Anaplastic Oligodendrogliomas (CAO), Non-classic Glioblastoma (NG)
9_Tumors	60	9	5727	4.5	NSCLC, Colon, Breast, Ovary, Leukemia, Renal, Melanoma, Prostate, CNS
11_Tumors	174	11	12,533	4.5	Ovary, Bladder/ureter, Breast, Colorectal, Gastroesophagus (Ga), Kidney, Liver, Prostate, Pancreas, Lung Adeno (LA), Lung Squamous (LS)
Leukemia 1	72	3	11,224	4.22	ALL B-cell, ALL T-cell, AML
Leukemia 2	203	3	12,600	1.4	AML, ALL, MLL
Lung Cancer	203	5	12,601	23.17	Adeno, Normal, Squamous, COID, SMC
SRBCT	83	4	2308	2.64	EWS, RMS, BL, NB

Table 2. Details of gene microarray data.

Experimental methodology

Benchmarked datasets

Eight gene microarray datasets were used in this paper to evaluate the performance of our approach and existing works. These datasets are all open-source datasets that are heavily used in the related works[1,4,52]. The details of these datasets are as listed in Table 2.

The imbalance degree is measured by the imbalance ratio (*IR*) that is defined as:

$$IR = \frac{\max(\ \#sample\quad of\quad classi\ )}{\min(\ \#sample\quad of\quad classj\ )} \tag{11}$$

The imbalance ratio of these datasets varies from 1.4 to 23.17, and the number of classes varies from 3 to 11. And on each microarray dataset, the features are normalized into [−1,1] for feature bunding.

Evaluation metrics

Experiments of classification, and non-parametric statistical tests were conducted, as to evaluate the proposed approach. For an in-depth investigation of the overall classification performance, we chose Accuracy, F_1-score and G-mean as metrics. Further, we also focus on the classification performance on each minority class in multi-classification problem. So, ROC curve and average AUC are also used for evaluation. All these metrics are based on four basic results, i.e., *TP* (True Positive), *FP* (False Positive), *TN* (True Negative) and *FN* (False Negative).

Accuracy is defined as:

$$Acc = \frac{\sum_{i=1}^{c} TP_i}{\sum_{i=1}^{c} (TP_i + FN_i)} \tag{12}$$

where, *c* is the number of classes in a dataset.

F_1-score for multi-classification is evolved from the traditional F_1 for binary classification. As known, F_1 for a certain class *i* is defined as:

$$F_{1-i} = \frac{2Precision_i \cdot Recall_i}{Precision_i + Recall_i} \tag{13}$$

where, $Recall_i = \frac{TP_i}{TP_i+FN_i}$,$Precision_i = \frac{TP_i}{TP_i+FP_i}$. Then F_1-score can be defined as:

$$F_1 - score = \frac{\sum_{i=1}^{c} F_{1-i}}{c} \tag{14}$$

G-mean is a main measurement in the field of data classification, which is very suitable for evaluating algorithm performance on datasets with different *IR* values, which is an index comprehensively considering recall rate and specificity:

$$G - mean = \sqrt[c]{\prod_{i=1}^{c} (Recall_i \times Specificity_i)} \tag{15}$$

where, $Specificity_i = \frac{TN_i}{TN_i+FP_i}$. Moreover, the results were supported with non-parametric statistical tests. First, Friedman tests were conducted to detect differences among all methods. Then, the Nemenyi post hoc tests were utilized to distinguish the difference of the sampling methods. This test calculates the critical value (CD) of each average ranking mainly using Eq. (16).

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6t}} \tag{16}$$

where, q_α is the critical value of the tukey distribution, k is the number of algorithms, and t is the number of datasets.

Experimental design and parameters

This paper aims to handle the high-dimension, small-sample and imbalanced gene microarray data. To evaluate the performance of the proposed approach, MGAN-FE is compared against two types of methods to handle this problem:

(1) Two embedded feature selection methods based on the nearest neighbor model combined with SVM classifier: μ-Relief + SVM (μR + SVM)[53] and NCFS + SVM[54]. And the parameters of these conventional methods refer to their classical settings. And 30% features are selected for both of the two methods according to literature experience.

(2) Two methods have demonstrated advantages for gene microarray data: Weighted Extreme Learning Machine (WELM)[1] and Probabilistic Neural Networks (PNN)[4]. And the parameters of these conventional methods refer to the corresponding references.

(3) Three GAN-based methods: BiGAN[44], EGBAD[47], MAD-GAN[48]. With regard to GAN models, the dimensionality of the latent noisy space was set to be 32. The Adam optimizer was selected with the 0.00001 learning rate set and the gradient penalty coefficient to 5.

As for compressing the running time, we use the simple decision tree as the multi-classifier in discriminator. All of experiments were conducted with an Intel Core i7 9750 H processor and an NVDIA GeForce GTX 1650, and were implemented in Python 3.6 using the TensorFlow 1.14.0 architecture. Each dataset is randomly partitioned into three subsets according to the data distribution and the 3-fold cross validation is performed to evaluate the classification performance. Each experiment is individually repeated 30 times. The result is averaged over these 30 runs. As for binary classification models in the experiments, the OvA mechanism is used to achieve multi-classification.

Results and discussion

Classification performance

(1) Accuracy results

Table 3 shows the comparison results of our MGAN-FB method with other classification methods in terms of accuracy. Compared with other two combinatorial solutions, GAN-based methods perform better in average performance. In this case, our proposed MGAN-FB method ranked first in 6 of the 8 gene microarray datasets, obviously superior to other GAN-based methods. This validates that it could improve the classification accuracy of the minority class while maintaining high accuracy for the majority class, especially for the coupling of high-dimension, small-sample and imbalance. However, as for datasets with higher dimensions and lower IR value, such as SRBT, its performance is inferior to traditional feature selection methods μR + SVM and NCFS + SVM. μR + SVM is clearly better than NCFS + SVM in these 8 datasets. In addition, on 11_Tumors, the best method is WELM, which is also one of methods that have demonstrated advantages in the field of multi-class classification tasks for gene microarray data. It is with the most classes, so the specially designed classifiers perform better, while our MGAN-FB only optimized a multi-classification module. Compared to DAGSVM, which is also always used in multi-class classification tasks for gene microarray data, WELM performed better.

Moreover, the results show that it could achieve accuracy more than 0.95 on seven of the eight benchmarked datasets. Ostentatiously, the accuracy of MGAN-FB in 9_Tumors is only 0.8378, which is not satisfactory, although it is highest among methods of comparison. As shown in Table1 2, 9_Tumors dataset has only 50 samples, but 9 classes, with 5727 features. It may be the most typical coupling problem of high-dimension, small-sample and imbalance for gene microarray data classification. So, in such difficult scenario, all methods degrade in performance. If we look at this problem from another aspect, it is not difficult to find that MGAN-FB improves the accuracy more significant. As for all datasets, its improvement seems no more than 0.03. So,

Datasets	μR+ SVM	NCFS+ SVM	WELM	DAGSVM	BiGAN	EGBAD	MAD-GAN	Ours
Brain Tumor1	0.8379	0.8248	0.8626	0.8372	0.8889	0.9091	0.9444	**0.9519**
Brain Tumor2	0.8898	0.8936	0.9136	0.8279	0.9211	0.9487	0.9450	**0.9644**
9_Tumors	0.7656	0.7067	0.8039	0.7967	0.7319	0.7293	0.7134	0.8378
11_Tumors	0.8035	0.8246	**0.9729**	0.9224	0.8889	0.9091	0.9444	0.9619
Leukemia 1	0.9623	0.9109	0.9555	0.9016	0.9236	0.9495	0.9602	**0.9896**
Leukemia 2	0.9619	0.9577	0.9621	0.9235	0.9602	0.9764	0.9644	**0.9867**
Lung Cancer	0.9398	0.9249	0.9396	0.8795	0.9447	0.9485	0.9222	**0.9524**
SRBT	**0.9992**	0.9963	0.9885	0.9831	0.9464	0.9751	0.9750	0.9954

Table 3. Accuracy results *Acc* on cancer gene microarray data.

Datasets	μR+SVM	NCFS+SVM	WELM	DAGSVM	BiGAN	EGBAD	MAD-GAN	Ours
Brain Tumor1	0.8235	0.8163	0.8791	0.8311	0.8018	0.8237	0.8441	**0.8800**
Brain Tumor2	0.9094	0.8925	0.9176	0.8875	0.9201	0.9575	**0.9304**	0.9542
9_Tumors	0.7956	0.7040	0.8833	0.8662	0.7981	0.8248	0.8473	**0.8844**
11_Tumors	**0.9559**	0.8766	0.9233	0.8399	0.9089	0.9218	0.9403	0.9455
Leukemia 1	0.9176	0.8693	0.9436	0.9465	0.9211	0.9385	0.9406	**0.9735**
Leukemia 2	0.9078	0.9063	0.9379	0.9214	0.9587	0.9778	0.9654	**0.9867**
Lung Cancer	0.8924	0.8650	0.9103	0.9094	0.9219	0.9391	0.9248	**0.9642**
SRBT	**0.9998**	0.9923	0.9975	0.9811	0.9749	0.9980	0.9896	0.9996

Table 4. $F\text{-}_1$ score results on cancer gene microarray data.

Datasets	μR+SVM	NCFS+SVM	WELM	DAGSVM	BiGAN	EGBAD	MAD-GAN	Ours
Brain Tumor1	0.9194	0.8635	0.9465	0.9028	0.9442	0.8286	0.8706	**0.9623**
Brain Tumor2	0.9291	0.8092	0.9230	0.9143	0.9691	0.8730	0.8304	**0.9890**
9_Tumors	0.8037	0.8094	0.8120	0.8145	0.8101	0.8411	0.8223	**0.8537**
11_Tumors	0.9233	0.9385	**0.9786**	0.9365	0.9488	0.8503	0.9488	0.9676
Leukemia 1	0.9391	0.8856	0.9148	0.9036	0.9681	0.8849	0.9569	**0.9691**
Leukemia 2	0.9039	0.9101	0.8996	0.8780	0.9104	0.8294	0.5542	**0.9392**
Lung Cancer	0.8812	0.8916	0.9215	0.9214	0.9006	**0.9656**	0.9283	0.9586
SRBT	0.9944	0.9958	0.9855	0.983	0.1879	0.9201	0.9769	**0.9922**

Table 5. G-mean results on cancer gene microarray data.

we can infer that MGAN-FB is more capable of reaching its potential in more comprehensive circumstances of high-dimension, small-sample and imbalance coupling.

(2) $F\text{-}_1$score results

Table 4 shows the comparison results of our MGAN-FB method with other classification methods in terms of $F\text{-}_1$ score, which has same trend shown in Table 3. And MGAN-FB also get the highest $F\text{-}_1$ score among 5 of the 8 cancer gene microarray datasets, with a prominent improvement.

On some datasets, the $F\text{-}_1$ score is degraded by proposed method slightly, compared to the result on Brain_Tumor2 obtained by EGBAD. It may be caused by the decrease of the majority class. Analysis of classification performance of the minority classes in a multi-class problem is complicated because the impact of imbalance to the discriminant among classes is usually heterogeneous. In addition, it is not trivial to characterize the class layout especially for high dimensional scenarios. So as for $F\text{-}_1$ score is a comprehensive classification result for the high-dimension, small-sample and imbalanced gene microarray data. In these datasets, the recalls of these classes are improved but the precisions maybe decreased. The classification performance of the classifier on this dataset is hard improved. When improving the $F\text{-}_1$ measure of some classes, the $F\text{-}_1$ measure of other classes may be degraded. On 11_Tumors and SRBT, μR + SVM performed best, it may a result of better features selection. More cancer-related features were picked out, so there were fewer false positives and missed positives in the classification.

Nevertheless, MGAN-FB obtains higher $F\text{-}_1$ score for each class on most of datasets. And the result on Brain_Tumor2 is still comparable to the best one. Our approach obtains the best result on the cancer microarray data except Leukemia2. The $F\text{-}_1$ scores are respectively improved from about 0.02 and 0.05 on average when it is compared with others. It also should be noted that the $F\text{-}_1$ measure results of Prostate class on 9_Tumors were not obtained, because this class only has 2 samples, and there is always no sample in training or testing set during cross validation.

(3) G-mean results

Table 5 shows the comparison results of our MGAN-FB method with other classification methods in terms of G-mean. It keeps a similar situation with Accuracy and $F\text{-}_1$ score results. And the difference is that the MGAN-FB method is not the preferred plan for Lung Cancer and 11_Tumors dataset, considering G-mean result. With the other 6 datasets, it is the best choice.

Since the *IR* value of Lung Cancer dataset is 23.17, which is the highest in the benchmarked datasets. So, it is a highly imbalanced dataset, which is more sensitive to G-mean result. As a comprehensive solution, MGAN-

FB is more suitable for the coupling problem of high-dimension, small-sample and imbalance. If one aspect of the problem is extremely prominent, more targeted special solutions will have better results. Besides, for 11_Tumors, although its IR value is only 4.5, it has 11 classes. So, the problem of relative imbalance between any two classes is more complicated. As a result, its performance is a little inferior to WELM, but still very competitive. In addition, it is important to note that despite large average accuracy is achieved by our MGAN-FB method, its G-mean results remain very competitive. It shows that our method has better stability for small-sample and imbalanced data.

Based on the G-mean results obtained by MGAN-FB, we also found that some datasets are sensitive to class imbalance but others are not. If the accuracy value is much larger than the G-mean value on a dataset, the corresponding classifier is significantly affected by the imbalance distribution. This situation was observed on several datasets, including Brain_Tumor1, Brain_Tumor2, 9_Tumors, Leukemia1 and Lung Cancer. Leukemia2 and SRBCT are both slightly sensitive to class imbalance. These results are actually related to multiple factors, such as the sample imbalance among classes, the class overlapping and small disjuncts, etc.

(4) ROC and AUC results

In the multi-classification problem, the AUC values of different classes in one dataset are always different. In order to comprehensively reflect the classification performance of a certain classifier, we adopt the average AUC values of all classes in the same dataset as the evaluation metrics, as shown in Fig. 4.

Figure 4 shows the average AUC results. In this case, the results of our proposed MGAN-FB method with decision tree multi-classifier are ranked first in 6 of the 8 datasets. In addition, GAN-based deep learning methods performed better than the traditional combinatorial solutions for highly imbalanced dataset. Feature selection-based methods perform better on high-dimension data. For dataset with more classes, WELM reflects advantages.

In the case of the Brain Tumor1 dataset, the AUC value of MGAN-FB is 0.9764, which is improved about 0.1 compared to DAGSVM; and it also improved by 0.0184, in the next best WELM method, whose AUC value is 0.9580.

However, nor is the improvement evident across all datasets. For instance, in 11_Tumors dataset, MGAN-FB seems to be no obvious advantage. It only came in third place, not as good as μR + SVM and WELM. A similar situation can be seen on SRBT. As known in Sect. 4.1, 11_Tumors dataset has 11 classes, which is the most among the 8 benchmarked datasets. And the more classed there are, the greater the possibility of misclassifying a class. So, the average AUC value maybe pulled down by these few misclassifying classes. From another aspect, the

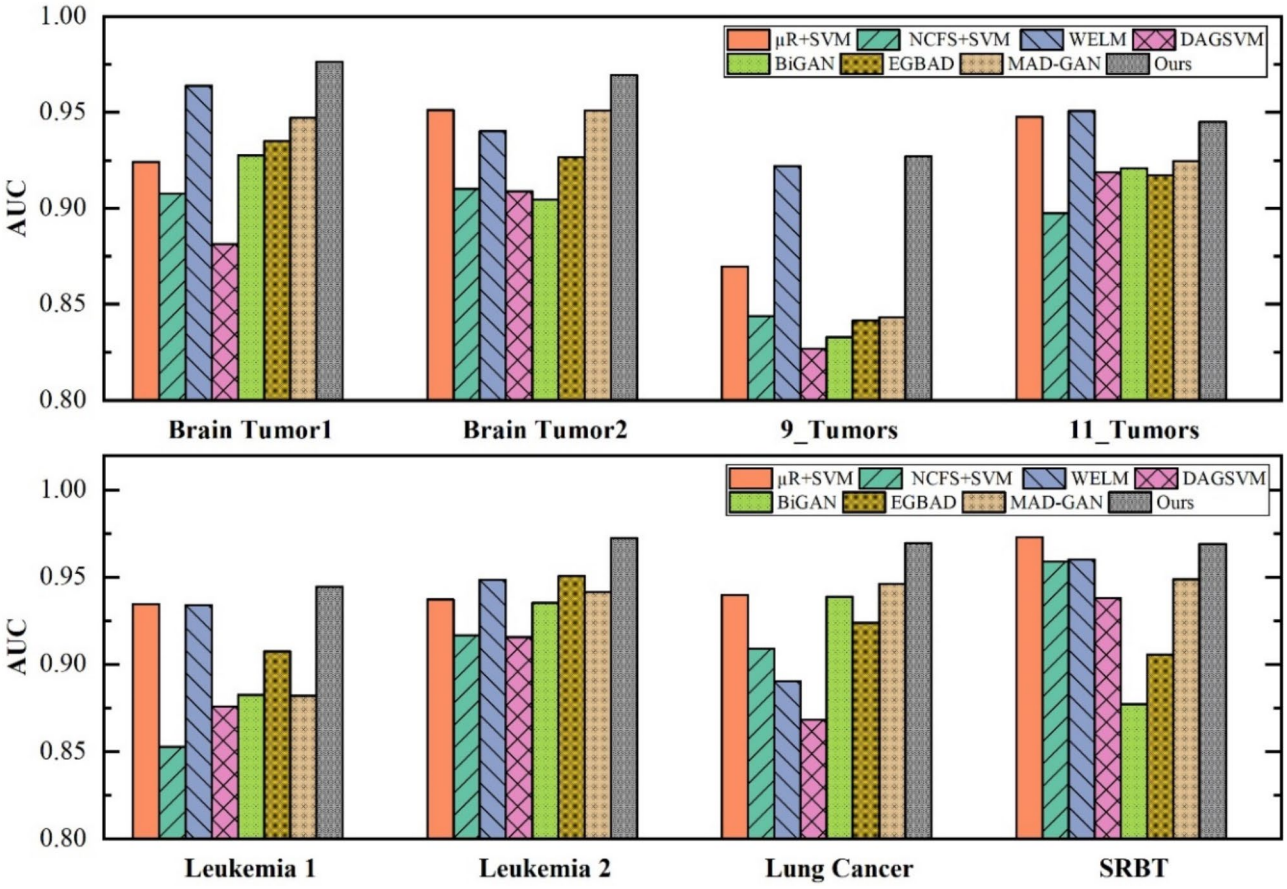

Fig. 4. AUC results cancer gene microarray data.

Datasets	μR + SVM	NCFS + SVM	WELM	DAGSVM	BiGAN	EGBAD	MAD-GAN	Ours
Brain Tumor1	2.7051	2.0654	0.3158	2.7052	0.797	0.8321	0.8621	**0.1498**
Brain Tumor2	2.2198	1.7982	0.9935	2.1203	1.444	1.199	1.022	**0.6445**
9_Tumors	2.3992	1.9614	0.4680	2.2582	1.238	1.467	1.297	**0.2172**
11_Tumors	4.6554	3.2581	0.7814	4.5871	0.795	1.178	1.202	**0.5760**
Leukemia 1	2.7435	0.773	0.3548	2.2279	0.386	1.113	0.658	**0.2803**
Leukemia 2	3.1089	1.336	0.5548	2.3918	3.850	1.363	1.159	**0.539**
Lung Cancer	3.1698	0.60802	**0.4445**	1.9268	1.359	1.029	0.78	0.4610
SRBT	0.9265	1.3164	0.5560	1.6856	0.605	0.654	0.659	**0.465**

Table 6. Running time (s) of various methods.

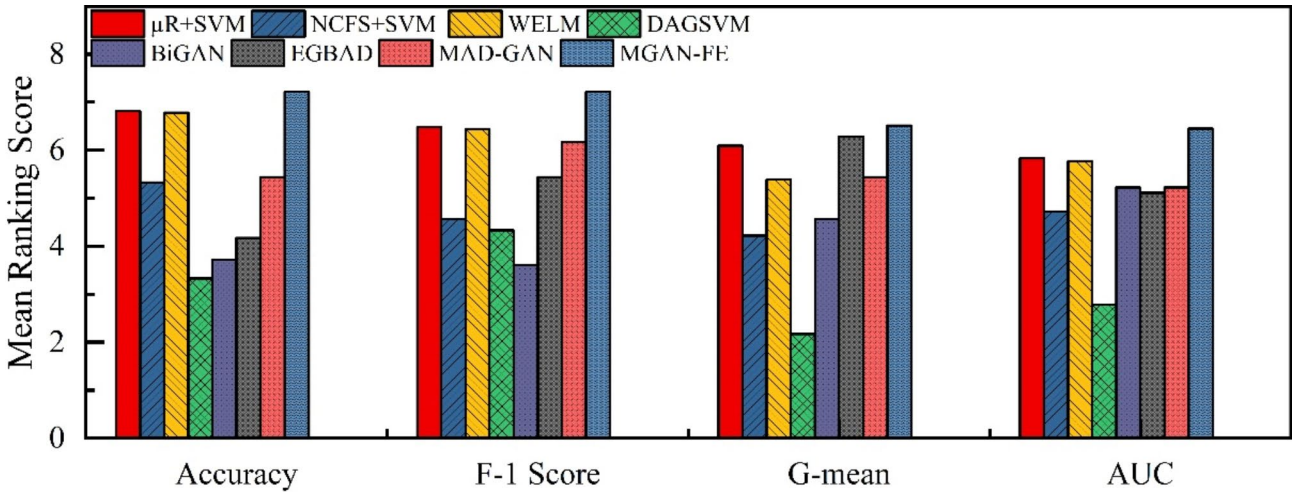

Fig. 5. Friedman's test rankings for various evaluation metrics with decision tree multi-classifier.

proposed MGAN-FB completes a multi-classification task with more classes, with maintaining or even slightly improving overall performance. It also fully proves the superiority of this method in multi-classification task.

Running time

Furthermore, we compare the running time of various hybrid methods in Table 6. The computational cost is a major consideration in the employment of ensemble learning algorithms. With the iterative training scheme, it usually takes longer time than any single classifier, which becomes an issue when dealing with large, high-dimension datasets. This section evaluates the computation cost of each method by measuring the training and testing time. Each experiment also runs 30 times. The running time are averaged over the 30 runs. It can be observed that our MGAN-FB method ranks as a leading and efficient method in all 8 datasets.

In the Brain Tumor1 dataset, the average time of the comparison GAN-based algorithms is approximately 0.8s, while our method can improve the speed to 0.1498 s, which is approximately 80% improvement; in the Brain Tumor2 dataset, the average time of the comparison GAN-based algorithms is approximately 1.2 s, while our method can improve the speed to 0.6445 s, which is approximately 45% improvement. In other dataset, the average time difference of the comparison algorithms is also high, with the fastest being 10^{-1} s magnitude and, the slowest being 1s magnitude, while our method still ranks the highest.

Since μR + SVM and NCFS + SVM, 30% of the useful features need to be pre-selected, so a lot of relevant calculations need to be done. As a result, it seems to take a longer time to complete training. For DAGSVM, large number of directed acyclic graphs need to be built for multi-classification, so it seems take the most time. WELM takes less time, because as a DNN model, feature selection and multiple classification tasks can be carried out by itself, which is similar to MGAN-FB. In Lung Cancer, it only needs 0.4445s, which is the best. However, with complexity of the coupling problem increases, using GAN as the base classifier also requires less training time. It is evident that our proposed method is very predictable in its superior efficiency among all methods. In average, our proposed method reduces the training time by more than 50%.

Non-parametric statistical tests

To reflect the difference in generalization performance between our MGAN-FB method and other methods, we used the Friedman test for a comprehensive comparison, and the average ranking results in Fig. 5.

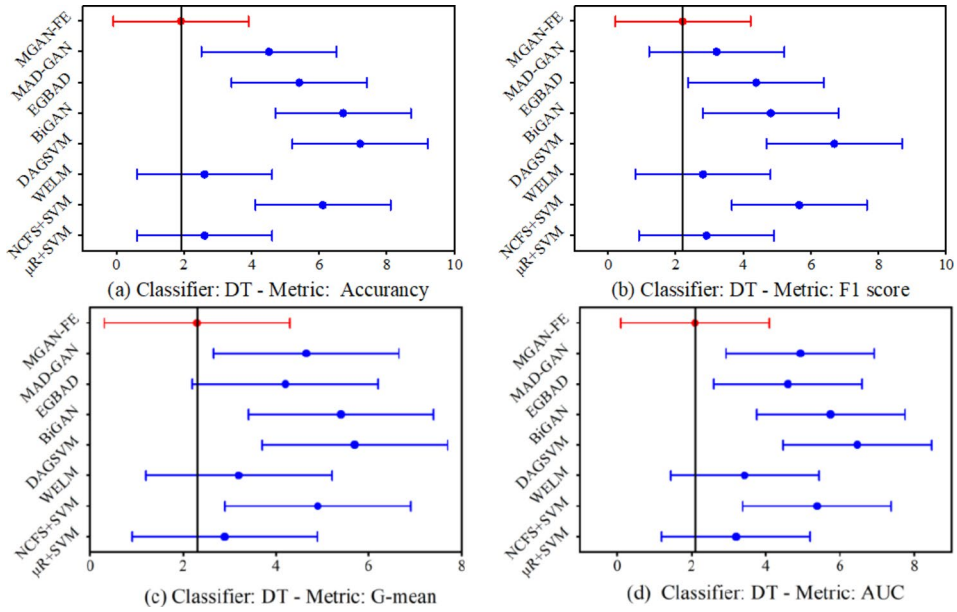

Fig. 6. Nemenyi test rankings for various evaluation metrics with decision tree multi-classifier.

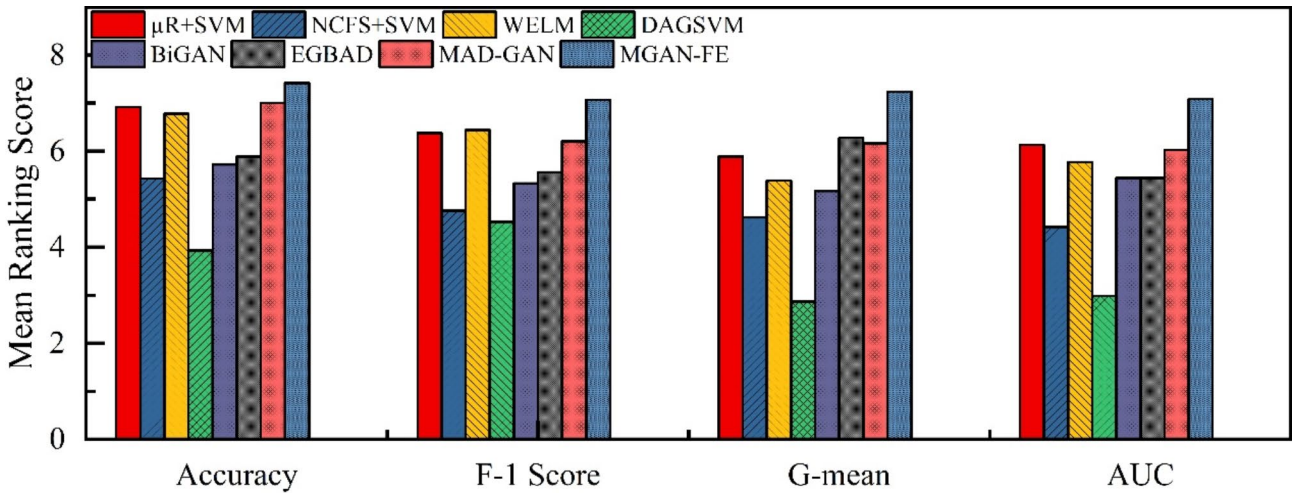

Fig. 7. Friedman's test rankings for various evaluation metrics with BP multi-classifier.

It can be inferred that all obtained results by each method for four evaluation metrics rejected the Friedman test at the $\alpha = 0.05$ confidence level (all significance is less than 0.05), which indicates a significant difference between the performance of all the methods. MGAN-FB obtains high rankings on four metrics with decision tree multi-classifier.

Then, the Nemenyi post hoc tests were utilized to distinguish the difference of the sampling methods. This test calculates the critical value (CD) of each average ranking mainly using Eq. (16). The CD value is the same for each metric because k and t in Eq. (16) are constants. In general, if the average ranking of an algorithm is greater than the CD value, the hypothesis is rejected with the corresponding confidence level. We have $k = 9$, $t = 8$, and q_{α}=0.05 to get CD value. The results are shown in Fig. 6. MGAN-FB has outstanding advantages as a control method in most cases.

To verify that it is the effect of the framework of MGAN-FB in this paper and not the effect of the decision tree multi-classifier themselves, we conducted a supplementary experiment with BP multi-classifier replacement decision trees in the discriminator and obtained similar results, as shown in Figs. 7 and 8. Thus, it can be concluded that the MGAN-FB method has more positive variability compared to other methods.

Ablation experiments

MGAN-FE has three key modules: Feature Bundling (FB), Encoder and Decoder (ED), and Multiclass Classifier Softmax (MCS). For ablation experiments, we set up four comparable methods on three metrics

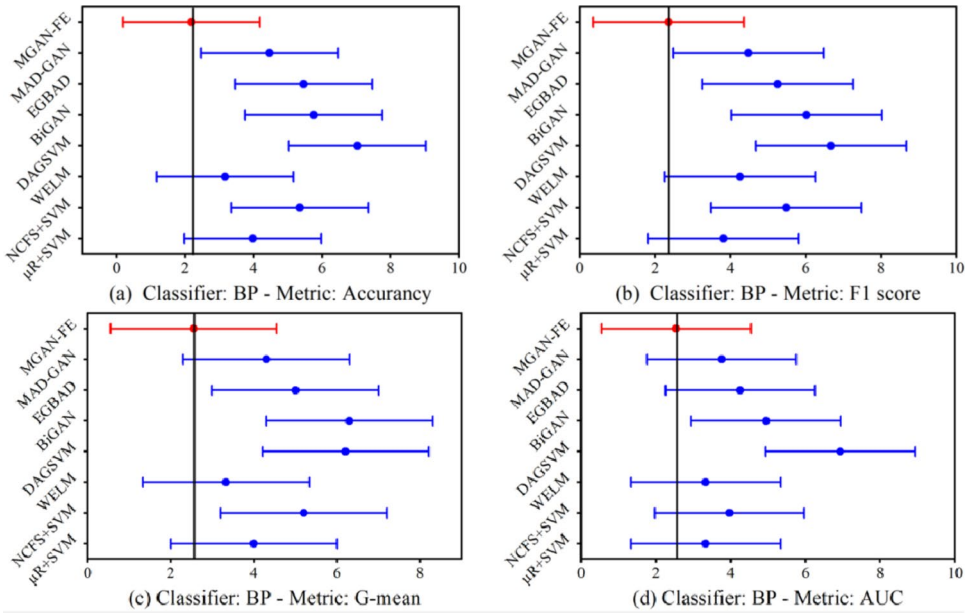

Fig. 8. Nemenyi's test rankings for various evaluation metrics with BP multi-classifier.

		Datasets							
		Brain Tumor1	Brain Tumor2	9_Tumors	11_Tumors	Leukemia 1	Leukemia 2	Lung Cancer	SRBCT
Acc	GAN+ ED+ MCS	0.8547	0.8212	0.6871	0.7566	0.8854	0.8801	0.8015	0.9422
	GAN+ FB+ MCS	0.8345	0.9245	0.6779	0.7562	0.8964	0.9533	0.7869	0.9552
	GAN+ FB + ED	0.9135	0.9554	0.7119	0.8336	0.9764	0.9744	0.9264	0.9635
	GAN+ FB + ED + MCS	**0.9519**	**0.9644**	**0.8378**	**0.9619**	**0.9896**	**0.9867**	**0.9524**	**0.9954**
$F\text{-}_1$	GAN+ ED+ MCS	0.8348	0.8134	0.8377	0.8264	0.8569	0.8477	0.8299	0.9653
	GAN+ FB+ MCS	0.8114	0.9022	0.8106	0.8016	0.8498	0.9466	0.8021	0.9668
	GAN+ FB + ED	0.8521	0.9306	0.8019	0.8633	0.9611	0.9682	0.9213	0.9732
	GAN+ FB + ED + MCS	**0.8800**	**0.9542**	**0.8844**	**0.9455**	**0.9735**	**0.9867**	**0.9642**	**0.9996**

Table 7. Ablation experiment results on *Acc* and $F\text{-}_1$.

for ablation experiments. MGAN-FE model is equal to GAN + FB + ED + MCS, three comparable methods are GAN + ED + MCS (without FB), GAN + FB + MCS (without ED), and GAN + FB + ED (without MCS), respectively. The results are as shown in Table 7.

Conclusions and future works

As gene expression data face high dimension, small-sample and multi-class imbalance. A hybrid deep learning method named MGAN-FE is proposed to handle this coupling problem of cancer gene microarray data at both feature and algorithmic levels. This method is based on GAN and tries to obtain better classification results than GAN. At feature level, the feature bunding mechanism is applied to merged sparse feature for dimensionality reduction. At algorithmic level, an optimal framework is devised by modifying the subnet structure of GAN, in which deep encoders and decoder are introduced into generator, and a softmax module coupled with multi-classifier are introduced into the discriminator. With a new objective function is distinctively designed for the proposed MGAN-FB model, considering encode loss, reconstruction loss, discrimination loss and multi-classification loss, it extends generative adversaria framework from the binary classification to the multi-classification field.

Experiments are conducted on eight open-source cancer gene microarray datasets. Our approach is compared against 7 recent works including combinatorial solutions with feature selection and multi-classifier, and GAN-based methods. Experimental results show that it has obvious advantages over other 7 compared methods in classification performance, running time and non-parametric tests. In the future, the proposed MGAN-FE should be further optimized considering class overlap and label missing, which are also typical problems for gene microarray data. Other future work includes introducing more kinds of GAN variants into consideration, and applying the proposed method to more datasets with large variety in class distribution.

Data availability

All data supporting the findings of this study are available within the paper.

Received: 22 October 2024; Accepted: 5 February 2025
Published online: 12 February 2025

References

1. Liu, Z., Tang, D. Y., Cai, R. Y. & Chen, F. H. A hybrid method based on ensemble WELM for handling multi class imbalance in cancer microarray data. *Pattern Recognit. Neurocomputing*. **266**, 641–650 (2017).
2. Kar, S., Sharma, K. D. & Maitra, M. Gene selection from microarray gene expression data for classification of cancer subgroups employing PSO and adaptive *K*-nearest neighborhood technique. *Expert Syst. Appl.* **42**, 612–627 (2015).
3. Hung, L. C., Hu, Y. H., Tsai, C. H. & Huang, M. W. A dynamic time warping approach for handling class imbalanced medical datasets with missing values: a case study of protein localization site prediction. *Expert Syst. Appl.* **192**, 116437 (2022).
4. Alexander, S., Constantin, F. A., Ioannis, T., Douglas, H. & Shawn, L. A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. *Bioinformatics* **21** (5), 631–643 (2005).
5. Jeremiah, I. et al. Optimizing microarray cancer gene selection using swarm intelligence: recent developments and an exploratory study. *Egypt. Inf. J.* **24**, 100416 (2023).
6. Ajin, R. N., Harikumar, R., Karthika, M. S. & Keerthivasan, C. Metaheuristic integrated machine learning classification of colon cancer using STFT LASSO and EHO feature extraction from microarray gene expressions. *Sci. Rep.* **14**, 16485 (2024).
7. Fadi, A. & Aleksandar, V. Machine learning methods for Cancer classification using gene expression data: a review. *Bioengineering* **10**, 173 (2023).
8. Hamzeh, O. et al. Prediction of tumor location in prostate cancer tissue using a machine learning system on gene expression data. *BMC Bioinform.* **21** (2), 1–10 (2020).
9. Liang, S. & Yang, M. A. A review of matched-pairs feature selection methods for gene expression data analysis. *Comput. Struct. Biotechnol. J.* **16**, 88–97 (2018).
10. Yin, Q., Adeli, E., Shen, L. & Shen, D. Population-guided large margin classifier for high-dimension low-sample-size problems. *Pattern Recogn.* **97**, 107030 (2020).
11. Shen, L. R., Meng, J. E., Liu, W. J., Fan, Y. S. & Yin, Q. B. Population structure-learned classifier for high-dimension low-sample-size class-imbalanced problem. *Eng. Appl. Artif. Intell.* **111**, 104828 (2022).
12. Maldonado, S. & López, J. Dealing with high-dimensional class-imbalanced datasets: embedded feature selection for SVM classification. *Appl. Soft Comput.* **67**, 94–105 (2018).
13. Santos, M. S., Abreu, P. H., Japkowicz, N., Fernández, A. & Santos, J. A unifying view of class overlap and imbalance: key concepts, multi-view panorama, and open avenues for research. *Inform. Fusion*. **89**, 228–253 (2023).
14. Peng, P. et al. Cost sensitive active learning using bidirectional gated recurrent neural networks for imbalanced fault diagnosis. *Neurocomputing* **407**, 232–245 (2020).
15. Sun, J., Li, J. & Fujita, H. Multi-class imbalanced enterprise credit evaluation based on asymmetric bagging combined with light gradient boosting machine. *Appl. Soft Comput.* **130**, 109637 (2022).
16. Yuan, X. H., Xi, L. J. & Abouelenien, M. A regularized ensemble framework of deep learning for cancer detection from multi-class, imbalanced training data. *Pattern Recogn.* **77**, 160–172 (2018).
17. Zhou, F. N., Yang, S., Fujita, H., Chen, D. M. & Wen, C. L. Deep learning fault diagnosis method based on global optimization GAN for unbalanced data. *Knowl. Based Syst.* **187**, 104837 (2020).
18. Wang, L., Wang, Y. & Chang, Q. Feature selection methods for big data bioinformatics: a survey from the search perspective. *Methods* **111**, 21–31 (2016).
19. Cui, X. T., Li, Y., Fan, J. H. & Wang, T. A novel filter feature selection algorithm based on relief. *Appl. Intell.* **52** (5), 5063–5081 (2022).
20. Xue, B. et al. A survey on evolutionary computation approaches to feature selection[J]. *IEEE Trans. Evol. Comput.* **20** (4), 606–626 (2016).
21. Alakuş, T. B. & Türkoğlu, İ. Feature selection with sequential forward selection algorithm from emotion estimation based on EEG signals. *Sakarya Univ. J. Sci.* **23** (6), 1096–1105 (2019).
22. Lisnianski, A., Frenkel, I. & Ding Yi. *Multi-state System Reliability Analysis and Optimization for Engineers and Industrial Managers*. (Springer, 2010).
23. Lin, S. W., Tseng, T. & Chou, S. A simulated-annealing-based approach for simultaneous parameter optimization and feature selection of back-propagation networks. *Expert Syst. Appl.* **1**, 1491–1499 (2008).
24. Lin, X. et al. Selecting feature subsets based on SVM-RFE and the overlapping ratio with applications in bioinformatics. *Molecules* **23** (1), 52 (2017).
25. Brock, G. N. et al. Which missing value imputation method to use in expression profiles: a comparative study and two selection schemes. *BMC Bioinform.* **9** (1), 1–12 (2008).
26. Kong, Y. & Yu, T. A graph-embedded deep feedforward network for disease outcome classification and feature selection using gene expression data. *Bioinformatics* **34** (21), 3727–3737 (2018).
27. Shi, J. H. et al. Antimicrobial resistance genetic factor identification from whole-genome sequence data using deep feature selection. *BMC Bioinform.* **20**, 535 (2019).
28. Kong, Y. C. & Yu, T. W. Forge net: a graph deep neural network model using tree-based ensemble classifiers for feature graph construction. *Bioinformatics* **36** (11), 3507–3515 (2020).
29. Das, S., Datta, S. & Chaudhuri, B. Handling data irregularities in classification: foundations, trends, and future challenges. *Pattern Recogn.* **81**, 674–693 (2018).
30. Vuttipittayamongkol, P. & Elyan, E. Neighbourhood-based undersampling approach for handling imbalanced and overlapped data. *Inf. Sci.* **509**, 47–70 (2020).
31. Zhao, Y. D. et al. A conditional variational autoencoder based self-transferred algorithm for imbalanced classification. *Knowl. Based Syst.* **218**, 106756 (2021).

32. Wei, J. et al. NI-MWMOTE: an improving noise-immunity majority weighted minority oversampling technique for imbalanced classification problems. *Expert Syst. Appl.* **158**, 113504 (2020).
33. Zhu, T., Lin, Y. & Liu, Y. Improving interpolation-based oversampling for imbalanced data learning. *Knowl. Based Syst.* **187**, 104826 (2020).
34. Fernández-Navarro, F., Hervás-Martínez, C. & Antonio Gutiérrez, P. A dynamic over-sampling procedure based on sensitivity for multi-class problems. *Pattern Recogn.* **44** (8), 1821–1833 (2011).
35. Wang, S. & Yao, X. Multiclass imbalance problems: analysis and potential solutions. *IEEE Trans. Syst. Man. Cybern Part. B.* **42** (4), 1119–1130 (2012).
36. Wang, S., Chen, H. & Yao, X. Negative correlation learning for classification ensembles. In *The International Joint Conference on Neural Networks, Barcelona, Spain*. 1–8 (2010).
37. Xiao, Y., Wu, J., Lin, Z. & Zhao, X. A deep learning-based multi-model ensemble method for cancer prediction. *Comput. Methods Programs Biomed.* **153**, 1–9 (2018).
38. Qi, Z., Wang, B., Tian, Y. & Zhang, P. When ensemble learning meets deep learning: a new deep support vector machine for classification. *Knowl. Based Syst.* **107**, 54–60 (2016).
39. Rasti, R., Teshnehlab, M. & Phung, S. L. Breast cancer diagnosis in DCE-MRI using mixture ensemble of convolutional neural networks. *Pattern Recogn.* **72**, 381–390 (2017).
40. Gayathri, R. G., Sajjanhar, A., Xiang, Y. & Ma, X. J. *Multi-class Classification Based Anomaly Detection of Insider Activities*. arXiv:2102.07277 (2021).
41. Engelmann, J. & Lessmann, S. Conditional Wasserstein GAN-based oversampling of tabular data for imbalanced learning. *Expert Syst. Appl.* **174**, 1–13 (2021).
42. Zheng, M. et al. Conditional Wasserstein generative adversarial network-gradient penalty-based approach to alleviating imbalanced data classification. *Inf. Sci.* **512**, 1009–1023 (2020).
43. Dlamini, G. & Fahim, M. Dgm: a data generative model to improve minority class presence in anomaly detection domain. *Neural Comput. Appl.* **33** (20), 13635–13646 (2021).
44. Donahue, J., Krähenbühl, P. & Darrell, T. *Adversarial Feature Learning*. arXiv2016:1605.09782 (2016).
45. Zenati, H. et al. *Efficient Gan-Based Anomaly Detection*. arXiv 2018:1802.06222 (2018).
46. Chen, H. & Jiang, L. GAN-based method for cyber-intrusion detection. *CoRR* (2019).
47. Jiang, W. et al. A GAN-based anomaly detection approach for imbalanced industrial time series. *IEEE Access.* **7**, 43608–143619 (2019).
48. Li, D. et al. MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks. In *International Conference on Artificial Neural Networks*. 703–716 (Springer, 2019).
49. Askari, N. et al. Investigating the function and targeting of MET protein as an oncogene kinase in pancreatic ductal adenocarcinoma: a microarray data integration. *BioImpacts* **15**, 30187 (2025).
50. Isola, P., Zhu, J., Zhou, T. & Efros, A. A. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 5967–5976 (2017).
51. Salimans, T. et al. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*. 2234–2242 (2016).
52. Li, J. et al. Leukemia 1: feature selection: a data perspective. *ACM Comput. Surv.* **94**, 1–45 (2016).
53. Nitisha, A. et al. Mean based relief: an improved feature selection method based on ReliefF. *Appl. Intell.* **53**, 23004–23028 (2023).
54. Yang, W., Wang, K. Q. & Zuo, W. M. Neighborhood Component feature selection for high-dimensional data. *J. Computers.* **7** (1), 161–168 (2012).

Acknowledgements

This work was supported by the Fundamental Research Funds for the Central Universities of Civil Aviation University of China (Grant no. 3122023033) and the Open Fund project of Information Security Evaluation Center of Civil Aviation University of China under Grant no. ISECCA-202103.

Author contributions

Y.F. Zeng: conceptualization, methodology, formal analysis, writing—original draf; Y.X. Zhang: methodology, software, writing—original draft; Z.K. Xiao: formal analysis; H. Sui: methodology, writing—review and editing.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to H.S.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.