International Journal of Machine Learning and Cybernetics (2023) 14:2877–2886
https://doi.org/10.1007/s13042-023-01804-x

ORIGINAL ARTICLE

A cross-validation framework to find a better state than the balanced one for oversampling in imbalanced classification

Qizhu Dai[1] · **Donggen Li**[2] · **Shuyin Xia**[2]

Received: 28 October 2022 / Accepted: 16 February 2023 / Published online: 1 March 2023

Abstract
Imbalance classification has always been a popular research point in the application of machine learning, data mining and pattern recognition. At present, there are also many techniques to reduce the negative impact of imbalance on classification performance, and oversampling is the most commonly used one. In this paper, we illustrate the relationship between imbalance rate and classification performance in the oversampling process from a novel perspective that oversampling may cause the loss of the distribution while minority class is enhanced. In addition, this paper proposes a novel cross-validation framework called "icross-validation" that can be used in sampling to find a better state than the balanced state. This framework is general and can be applied into various oversampling methods. In comparison with some state-of-the-art and widely used oversampling methods, the experimental results on some real data sets demonstrate the effectiveness of the icross-validation. All code has been released in the open source icross-validation library at https://github.com/syxiaa/icross-valiation.

Keywords Sampling · Cross validation · Oversampling · Imbalanced classification

1 Introduction

An imbalance probably deteriorate the performance of classifiers in imbalanced classification, which is an important and widespread task in machine learning. Therefore, solving the problem of imbalanced learning has far-reaching significance. Many techniques are generally used to solve this problem. They can be roughly divided into two categories: algorithm-level methods and data-level methods [1–3]. One method at the algorithm level is to modify or design a new classification algorithm to improve the performance of the learning algorithm. One method is to design a new classifier performance evaluation index. There is a specific type of algorithm level method called cost-sensitive learning [4, 5].Among them, cost-sensitive learning is a scheme of algorithm-level modification [6]. This scheme does not use a standard loss function, but introduces the concept of misclassification cost to minimize conditional risk. By severely penalizing certain categories of classification errors, the importance of these categories is increased in the process of classification training. For example, Datta et al. proposed a dictionary-based linear programming framework to minimize the maximum deviation of the minimum loss value of a specific category [7]. They also proposed a multi-objective optimization framework called radial boundary intersection by training support vector machines on two and multiple data sets. It overcomes the disadvantage of high cost of adjusting parameters [8]. The data-level method generally balances the majority class and the minority class by resampling the original data. Resampling the original data by oversampling the minority or undersampling the majority is the most popular method of addressing the imbalance problem [9–12], and oversampling is more effective and popular than undersampling [13–17].

For the development of oversampling technology, the synthetic minority oversampling technique (SMOTE) is one of the most well-known data preprocessing methods for balancing different numbers of samples in each class; it

✉ Shuyin Xia
shuyxia@163.com

Qizhu Dai
daiqizhu@cqu.edu.cn

Donggen Li
donggen_li@163.com

[1] College of Computer Science, Chongqing University, Chongqing 400044, China

[2] the College of Chongqing Key Laboratory of Computational Intelligence, Chongqing University of Telecommunications and Posts, Chongqing 400065, China

proceeds by randomly interpolating between the minority points, extending the minority classes, bringing the data to a balanced state, and thereby improving the classification accuracy [18, 19]. There has been a succession of many improved versions of SMOTE, which are roughly divided into two categories. The first category filters out some noise or dangerous points that are not suitable for expansion and then performs SMOTE. For example, Rivera et al. introduced a new oversampling technique that focuses on noise reduction and selective sampling of the minority group, which results in an improvement in the prediction of minority group membership [20]. Saez et al. proposed an iterative ensemble-based noise filter that can overcome the problems produced by noisy and borderline examples in imbalanced data sets [21]. Another category is to improve the sampling strategy to make it general for more data sets with various distributions. In this category, Han et al. proposed two new minority oversampling methods, borderline-SMOTE1 and borderline-SMOTE2, in which only the minority examples near the borderline are oversampled [22]. Das et al. introduced two probabilistic oversampling approaches for synthetically generating and strategically selecting new minority class samples [23]. Abdi et al. suggested an elegant blend of boosting and oversampling paradigms by introducing the Mahalanobis distance-based oversampling technique (MDO) [24]. In addition, some of the latest algorithms combine the above two methods. For example, Xie et al. proposed a minority oversampling technique based on local densities in low-dimensional space [25]. Zhou et al. proposed an oversampling algorithm based on weights to calculate the synthesis position of new samples [26]. He et al. proposed an adaptive synthesis (ADASYN) sampling method, which uses a weighted distribution for different minority examples according to their learning difficulty [27].

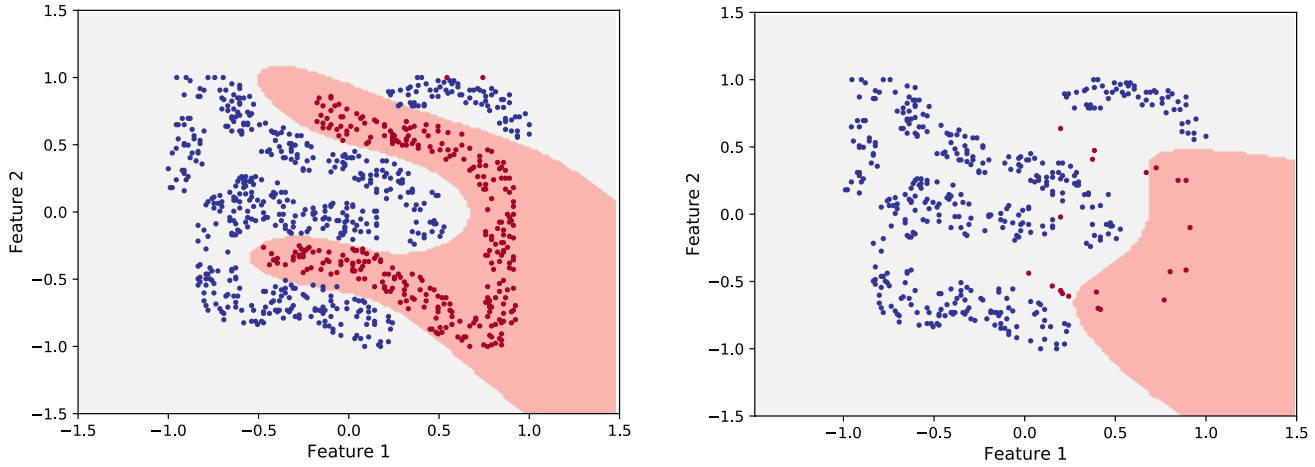
Although these oversampling methods have achieved satisfactory results on some imbalanced data sets, almost all of them default the balance to the final state of sampling. Nevertheless, balance is not necessarily the optimal state for classification. For example, Prati et al. conducted a detailed study on the class imbalance, and evaluated the performance of the class imbalance treatment method through a large-scale experimental design, proving that the higher the class imbalance degree, the greater the loss, and the class imbalance treatment method can only restore part of the performance loss [28]. Barella et al. determined the convex relationship between the class imbalance rate and the accuracy rate by learning the imbalance of the data in the classification [29]. Thabtah et al. believe that when the complexity of the data in the sampling process is high, imbalanced data will bring greater difficulties to classification. Therefore, the data complexity measurement is used to estimate the optimal sample size of the data imbalance pre-processing technique [30]. However, the measurement for data complexity degree is a difficult problem in itself, so this method is limited to some data sets with specific distributions and not general. This paper shows a fact that in the process of data gradually becoming balanced, the difference between the distribution of the current data and that of the original data may become increasingly large. When the sampling method is not suitable for data with a certain distribution, this negative effect will be considerable. There are two contributions of this paper, as follows:

- We analyze the relationship between balanced rate and classification performance in the sampling process from a novel perspective that oversampling may cause the loss of the distribution while minority class is enhanced.
- This paper proposes a novel cross-validation framework called "*k*-fold icross-validation" that can be used in sampling to find a better state than the balanced state. This framework is general and can be applied into various oversampling methods.

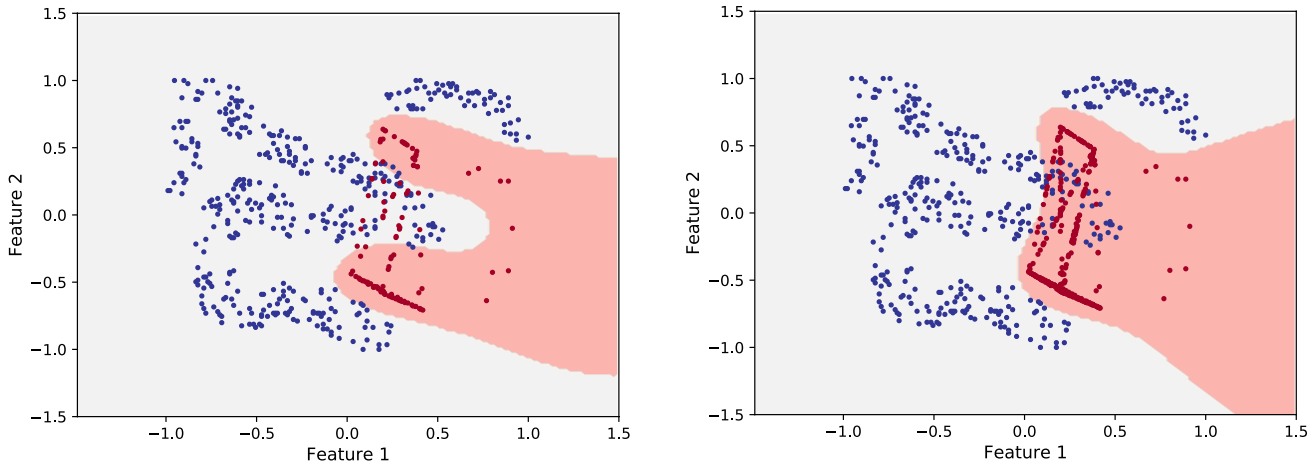
2 Relationship between prediction performance and imbalance rate

As pointed out by Chawla [19], an imbalance has a negative effect on a classifier because the majority class of data points always has a greater impact on the classifier than the minority points. Therefore, in imbalanced classification, current oversampling methods always aim at reaching an absolutely balanced state in the data set. However, in the process of sampling, the data distribution is likely to differ increasingly from the distribution of the original data set, and the difference in the distribution between the original data set and the sampled data set will obviously deteriorate the generalizability of the classifier. If the negative effect of the distribution difference is larger than the positive effect of balance, the classification performance can deteriorate. This phenomenon is particularly noticeable in some cases in which the distribution of a data set is very easy for a sampling method to change, such as when the data set contains noise [21].

Figure 1 shows the change in the decision boundary when borderline-SMOTE is used for oversampling on fourclass(the benchmark data set). In Fig. 1b, the minority class of data points in the original imbalanced data set is very sparse, and many data points in the minority class are incorrectly classified when the support vector machine (SVM) classifier is directly implemented on the original imbalanced data set. Therefore, the decision boundary is very unreasonable. As shown in Fig. 1c, when borderline-SMOTE runs to the 20th iteration, the decision boundary is very close to that in the original data set shown in Fig. 1a. Therefore, it performs well and can correctly classify much



(a) The original dataset (b) Imbalanced dataset



(c) 20^{th}: The optimal state (d) 60^{th}: The balanced state

Fig. 1 The decision boundaries of a data set and its sampled results based on the SVM classifier. The blue dots denote the majority vote category, and the red dots denote the minority vote category. The dark areas represent the decision range of the minority vote category, and the remaining light areas represent the decision range of the majority vote category

more of the minority class of data points than in Fig. 1b. Furthermore, the decision boundary in Fig. 1c is also closer to that in Fig. 1a than that in Fig. 1d, in which the data set is balanced when borderline-SMOTE is run to the 60th iteration. The reason is that although the balanced state can make the number of points of the minority class almost equal to that of the majority class, excessive sampling exists in the balanced state, and the data distribution and decision boundary are very different from those of the original data set.

The classification performance can be represented by the evaluation index in imbalanced classification, such as the *G-mean*. The existing KL divergence metric can calculate the distribution difference between two data sets, but unfortunately it is only applicable when the size of the two data sets is equal. So, referring to the characteristics of KL divergence, we propose a novel measurement for measuring the difference between two distributions containing different numbers of points. The first characteristics is that, the distribution difference between two data sets containing the same data points is equal to zero; the second is that, for a data set D and its oversampled result D', the distribution difference is larger if the data points in D' are farther from those in D. Based on above characteristics, we design the distribution difference for oversampling as follows:

Definition 1 (Division): Given two data sets $D = \{x_i, i = 1...n\}$ and D', the number of data points in D is denoted as $| D |$. Assuming that $| D' | > | D |$, a division of D on D' implements an initial operation of n-means clustering on D' with $x_i (i = 1...n)$ as centers.

Definition 2 (Distribution difference): Given a data set $D = \{x_i, i = 1...n\}$ and its oversampled result $D' = \{x'_i, i = 1...m\}$, where n and m denote the number of data points in D and D', respectively, n clusters $\{C_i, i = 1...n\}$ are generated by generating a division of D on D'. The number of data points in D is denoted as $| D |$. For two points A and B, their distance is denoted as $dist(A, B)$. The distribution difference of C_i relative to x_i is defined as:

$$DD[x_i \mid C_i] = \frac{1}{| C_i |} \sum_{x_i \in C_i} \text{dist}(x_i, \bar{C}_i), \tag{1}$$

where $\bar{C}_i$ represents the center of C_i. The distribution difference of D' relative to D is defined as:

$$DD[D \mid D'] = \frac{1}{n} \sum_{i=1}^{n} DD[x_i \mid C_i]. \tag{2}$$

The visualization results of changing G-means (i.e., classification performances) and distribution differences using various sampling methods on some real-world data sets are shown in Figs. 2 and 3. Some rules can be extracted and represented, as shown in Fig. 4. The balance rate used in this paper is defined in Definition 2, and the imbalance rate used in this paper is equal to the the reciprocal of balance rate.

Definition 3 (Balance rate): Given a data set D consisting of a majority class D_- and a minority class D_+ and given that the number of data points in a data set P is denoted as $| P |$, the balance rate of D is defined as:

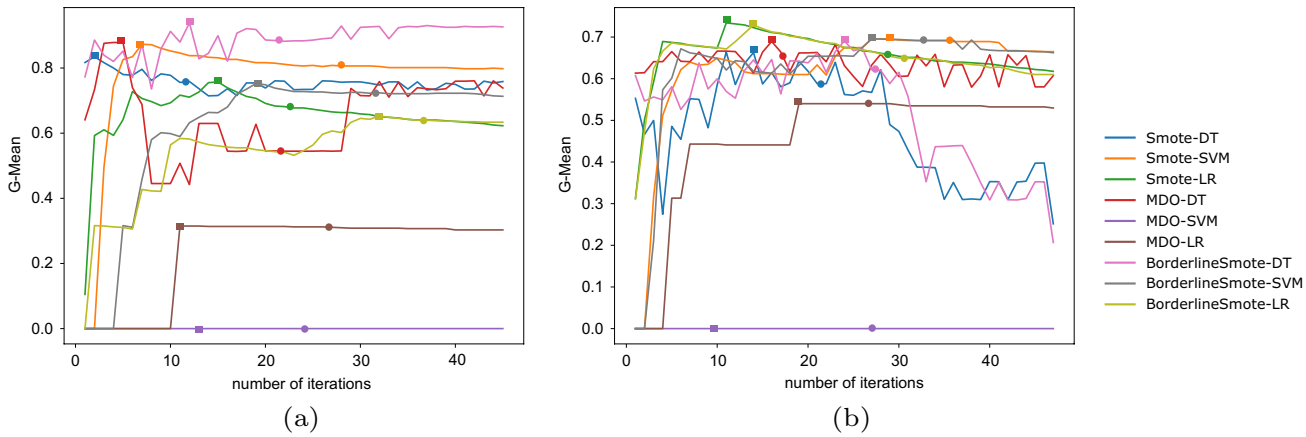


Fig. 2 Visualization results of the changing G-mean (i.e., classification performance) using various sampling methods on two real-world data sets. (a) Average G-mean on the data set vowel using various sampling methods. (b) Average G-mean on the ThoracicSurgery data set using various sampling methods. The square represents the optimal state, and the dots represent the balanced state

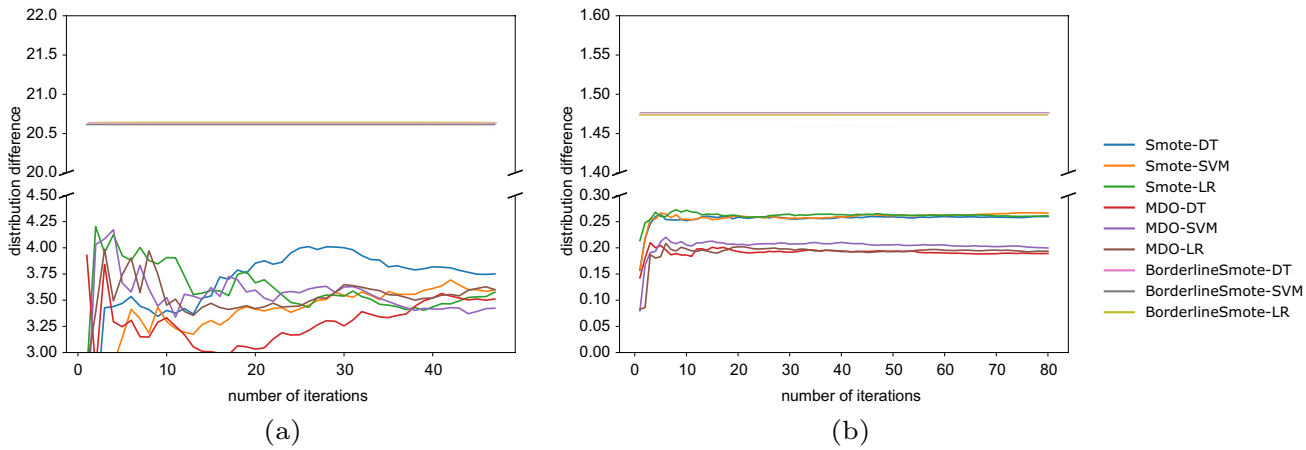


Fig. 3 Visualization results of changing the distribution difference using various sampling methods on two real-world data sets. (a) Average distribution difference on the vowel data set; (b) Average distribution difference on the ThoracicSurgery data set

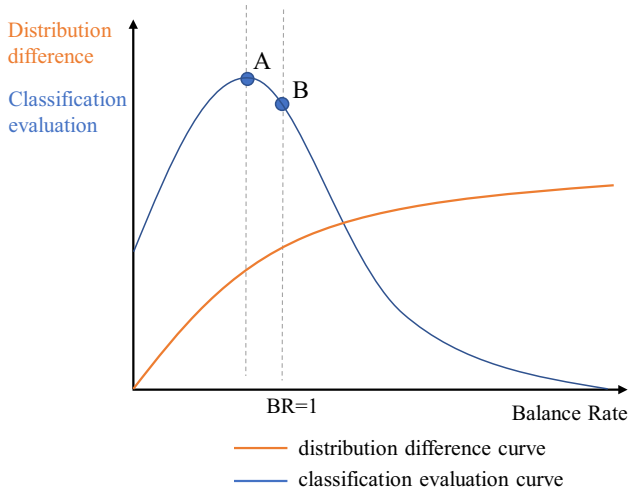


Fig. 4 Distribution difference curve and classification evaluation curve. The orange line represents the curve of the distribution difference as the balance rate changes, and the blue curve represents the curve of the classification evaluation as the balance rate changes. The blue points represent the different results of classification evaluation on different balance rates

$$BR(D) = \frac{|D_+|}{|D_-|}. \tag{3}$$

Supposing that the other factors of classification performance are equal, classification performance is better when the balance rate is closer to 1. The characteristics in Fig. 4 can be summarized as follows:

- The orange curve will increase when new sampled points are added to the original data because the distribution difference will increase. When an increasing number of points are generated, as shown in Fig. 1d, those points are limited to a local area, resulting in a stable distribution difference at the end of the orange curve.
- The blue curve changes along a parabola, where A is the point with the best classification performance. At the front of the parabola (that is, between the origin and A), due to the high proportion of original data points, the sampling result is reliable, resulting in a low distribution difference and a small negative impact on classification performance. Meanwhile, the balance rate continues to increase from a small value to almost 1. Its benefits to classification outweigh the negative effects of the initial balance rate. Therefore, the classification performance is improved.
- In the second half of the blue curve (that is, after A), due to the small proportion of original data points, the reliability of the sampling results is low, resulting in a high distribution difference and a large negative impact on classification performance. The resulting negative impact on classification is larger than the benefits of the increasing balance rate. Therefore, the classification performance is decreased.
- For the blue curve, similar to the phenomenon in Fig. 2, point B, where the balance rate is equal to 1, is located in the second half of the parabola (that is, after A). In other words, the classification performance cannot increase after the point where the balance rate is equal to 1. The reason is that not only has the balance rate moved far from 1 but also the distribution difference has expanded. Both of these effects are negative. When this process has continued sufficiently far, the classification performance will be lower than that of the original data set and may even fall to 0.

To the best of our knowledge, there are no methods of finding the optimal state of a sampling method in imbalanced classification. We will present a method of finding the optimal state in the sampling process in the next section.

3 Validation framework to optimize the balance rate

3.1 Motivation

To find a better state than balance, the most direct method is to use conventional cross-validation. However, conventional cross-validation may result in an unstable and unreliable sampled data set because minority samples are likely to be unevenly distributed in different validation sets. Additionally, minority samples are very important for the validation index in imbalanced classification, and conventional evaluation indexes, such as test accuracy, are seldom used in imbalanced classification. For example, consider a data set D that consists of two kinds of points, the numbers of which are 1000 and 10. If 5-fold cross-validation is used, one possible case is that the number of minority samples in the five sets are 0, 0, 0, 0, and 10. In this case, the validation indexes of the imbalanced data in the first four sets are nonsensical and harmful for the average validation.

3.2 A cross-validation method for imbalance classification

To address the above problem, we design a validation framework for imbalanced classification by applying the

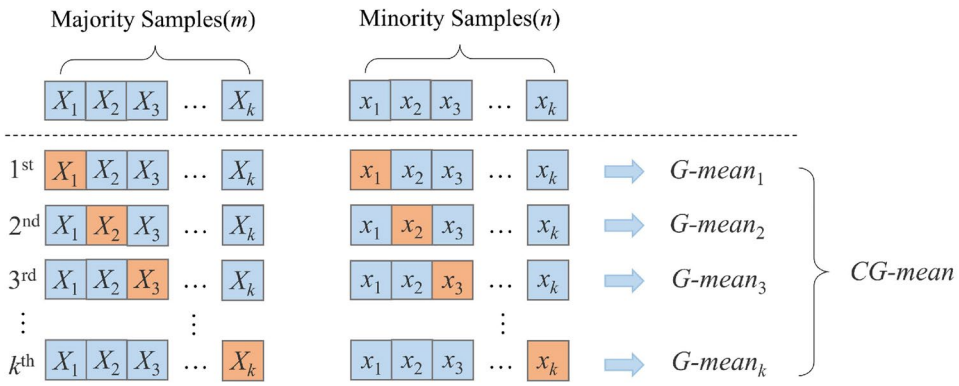


Fig. 5 Schematic diagram of the proposed leave-one-out cross-validation for imbalance classification

leave-one-out strategy for the minority class samples. In addition, to protect the majority samples and keep the overall distribution consistent, the majority samples are divided into sets whose number is equal to the number of minority samples. As shown in Fig. 5, m denotes the number of majority samples, and n denotes the number of minority samples. $x_i(i = 1, 2...n)$ denotes each minority sample point, and the majority samples are divided into n equal parts $\{X_i \mid i = 1...n\}$. So, the number of samples in each majority set X_i is equal to $\frac{m}{n}$. After the division is completed, we merge a majority class data set X_i and a minority class sample point x_i into a set, and use this set as one fold of the leave-one-out cross-validation. However, we consider that if the minority sample size is large, the implementation cost of the framework will be quite high. Therefore, as shown in Fig. 6, we proposed a k-fold cross-validation framework for imbalance classification. In this method, the majority and minority samples are divided into k parts respectively and then cross-validated is used.

In Fig. 6, one majority set $X_i(i = 1...k)$ and one minority set $X_i'(i = 1...k)$ are combined as a cross-validation set, and the remaining points make up the cross training set. In particular, when $k = n$, X_i' represents a minority sample, and the framework is converted to be leave-one-out framework. In this way, the minority samples are evenly distributed in each validation set, and the majority samples are also considered according to the balance rate. We can calculate $G\text{-}mean_i(i = 1...k)$ for each validation set using the proposed k-fold cross-validation, and average the k *G-mean* metrics to get the final *CG-mean*. *CG-mean* represents the average value of $G\text{-}mean_i$. We can calculate *CG-mean* for each sampling iteration, and the balance rate corresponding to the highest *CG-mean* value is the optimal balance rate. We named the proposed framework for imbalance classification as k-fold icross-validation.

Algorithm 1 k-fold icross-validation

Input: Training set $D = P \cup Q$, P represents the minority samples, Q represents the majority samples;
Output: the sampled result D'.

1: $s = int(num(P)/num(Q))$, and $D_0' = D$;
2: **for** $1 \le j \le s$ **do**
3: D_j is generated from D using an oversamper and $num(D_j) = num(Q)$;
4: $D_j' = D_j \cup D_{j-1}'$;
5: Compute $CG\text{-}mean_j$ on D_j';
6: **end for**
7: $CG\text{-}mean^* = \max(CG - mean_j(1 \le j \le s))$, and D' is the sampled result corresponding to $CG\text{-}mean^*$;
8: **return** D'.

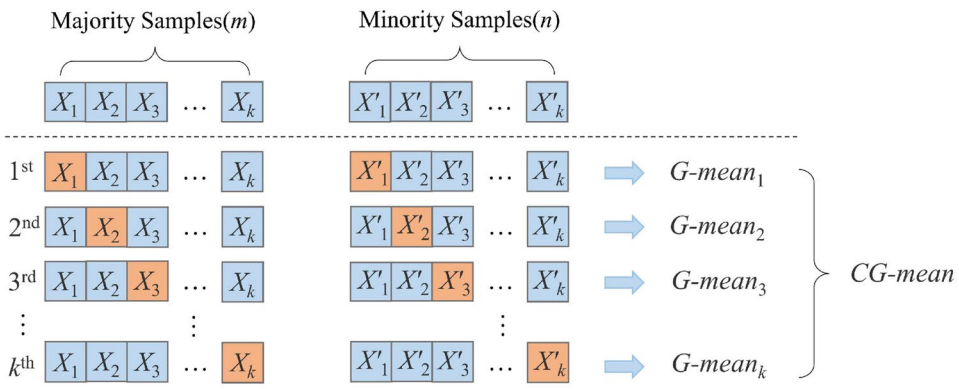


Fig. 6 Schematic diagram of the proposed k-fold cross-validation for imbalance classification

Figure 7 shows the process of finding the optimal state on a specific oversampling method using the k-fold icross-validation. In the $jth(1 \le j \le s)$ sampling iteration process, D_j is generated from the original date set D, and each $D_j(1 \le j \le s)$ contains a same number of samples, which is equal to the number of the minority samples in D. In the k-fold icross-validation, $CG\text{-}mean_j$ is generated on the data sets including D and those $D_i(i \le j)$. The optimal value $CG\text{-}mean^*$ is corresponding to the $CG\text{-}mean_j$ who has the highest value. The pseudo codes of k-fold icross-validation is provided in Algorithm 1. The proposed framework can not only give sufficient attention to the minority samples but can also consider the majority samples by ensuring that the balance rate in the validation set is equal to that in the training set, i.e., that the data distributions of the validation set and the training set are as consistent as possible.

4 Experiment

In this section, five widely used or state-of-the-art sampling methods, SMOTE, ADASYN, Borderline-SMOTE, MDO, and SVMSMOTE, are used to validate the effectiveness of the proposed framework on eight real-world data sets. Five classifiers are selected based on "scikit-learn" package: GBDT, SVM, LR, DT and LGB. The parameter k is set to the default value of 5 in these sampling methods, and the other parameters are set to the default values. The parameter k is set to 5 in k-fold icross-validation framework. The information on the imbalances of these data sets is shown in Table 1, and the balance rate is between 1/6 and 1/27. Each data set is divided into a training set and a test set in a 3:1 ratio. During this process, the majority class and minority class are divided separately to make the balance rates equal.

In the case of imbalanced learning conditions, since the classification accuracy of traditional classification is biased towards majority, the classification accuracy can not fully evaluate the observed learning algorithm. In our experiment, a set of ROC-based evaluation metrics are used as follows: *accuracy*, *precision*, *recall*, *F1-measure*, *G-mean*. These metrics are listed in the Eqs. (4)–(8). The minority class is set as the positive class, and the majority class is set as the negative class. Among them, *accuracy* represents the correct ratio of positive and negative predictions among all prediction results. *Precision* is the ratio of correct predictions among all predicted positive samples. It reflects how many of the predicted positive samples are true positive samples. *Recall* is the ratio of correctly predicted instances in all real positive classes, that is, how many positive class instances are correctly classified. *F1-measure* is the harmonic average of *precision* and *recall*. It is closer to the smaller one, which means that *F1-measure*

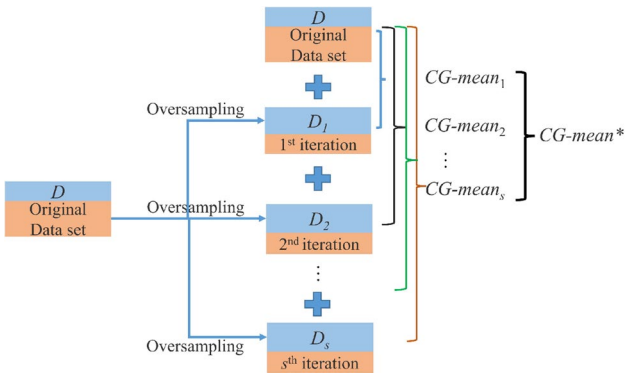


Fig. 7 Schematic diagram of searching for the optimal state on a certain sampling algorithm

Table 1 The information on the imbalances of the data sets

Datasets	#Sample	#Majority	#Minority	#Attribute
breastcancer	699	630	69	10
ecoli	336	316	20	8
htru2	17,898	12,189	1234	8
fourclass	575	416	15	2
userknowledge	403	353	50	6
vowel	528	480	48	11
BreastTissue	106	69	10	9
letter	20,000	14,459	541	16

Table 2 Comparison results of the optimal state and the balanced state under the five classifiers

Clf	Methods	Accuracy	Precision	Recall	F1	Gmean	AUC	CG-mean
SVM	Smote	**0.9177**	**0.6035**	0.8920	**0.6988**	**0.8994**	**0.9039**	**0.9100**
		0.9099	0.5857	**0.8930**	0.6868	0.8948	0.8998	0.8894
	ADASYN	**0.8748**	**0.6021**	**0.7600**	**0.6562**	**0.8153**	**0.8223**	**0.9213**
		0.8735	0.5941	0.7597	0.6510	0.8145	0.8214	0.9117
	BorderlineSmote	**0.8977**	**0.5640**	**0.8248**	**0.6357**	**0.8567**	**0.8627**	**0.8866**
		0.8845	0.5156	**0.8248**	0.6036	0.8490	0.8553	0.8589
	MDO	**0.9115**	**0.6010**	**0.7052**	**0.6147**	**0.7971**	**0.8148**	**0.9208**
		0.9069	0.5912	**0.7052**	0.6061	0.7950	0.8124	0.9180
	SVMSMOTE	**0.9119**	**0.5704**	0.8748	**0.6696**	**0.8882**	**0.8932**	**0.9082**
		0.8670	0.4340	**0.8849**	0.5625	0.8668	0.8730	0.8634
GBDT	Smote	0.9604	**0.7841**	0.7911	**0.7776**	0.8588	0.8826	**0.9751**
		0.9614	0.7668	**0.8007**	0.7716	**0.8630**	**0.8872**	0.9703
	ADASYN	**0.9698**	**0.8277**	0.7908	**0.8018**	0.8709	0.8873	**0.9795**
		0.9692	0.8235	**0.7917**	0.7988	**0.8712**	**0.8875**	0.9738
	BorderlineSmote	**0.9475**	0.6574	0.7927	0.7080	0.8573	0.8759	**0.9672**
		0.9475	**0.6742**	**0.8013**	**0.7163**	**0.8654**	**0.8799**	0.9587
	MDO	**0.9604**	**0.7532**	**0.6583**	**0.6931**	**0.7826**	**0.8204**	**0.9737**
		0.9578	0.7407	**0.6583**	0.6843	0.7815	0.8190	0.9706
	SVMSMOTE	**0.9519**	**0.7504**	0.8410	**0.7601**	0.8937	0.9008	**0.9668**
		0.9440	0.7111	**0.8583**	0.7380	**0.8995**	**0.9044**	0.9622
LR	Smote	**0.8627**	**0.4446**	0.8518	**0.5457**	0.8549	0.8580	**0.8879**
		0.8415	0.3823	**0.8929**	0.5101	**0.8598**	**0.8631**	0.8818
	ADASYN	**0.8831**	**0.5365**	0.8488	**0.5974**	**0.8640**	0.8682	**0.9186**
		0.8628	0.4827	**0.8744**	0.5602	0.8640	**0.8685**	0.9114
	BorderlineSmote	**0.8544**	**0.4624**	0.8005	**0.5262**	0.8241	**0.8308**	**0.8715**
		0.8226	0.3480	**0.8279**	0.4593	**0.8190**	0.8250	0.8554
	MDO	**0.8476**	**0.4567**	0.8650	**0.5370**	**0.8524**	**0.8562**	**0.8574**
		0.8296	0.3953	**0.8665**	0.4999	0.8428	0.8467	0.8471
	SVMSMOTE	**0.8412**	**0.4474**	**0.9085**	**0.5417**	**0.8681**	**0.8719**	**0.8517**
		0.7848	0.3046	0.9029	0.4278	0.8330	0.8391	0.8182
DT	Smote	0.9546	0.7338	**0.7826**	0.7515	**0.8625**	**0.8753**	**0.9710**
		0.9586	**0.7680**	0.7698	**0.7627**	0.8572	0.8716	0.9703
	ADASYN	0.9584	0.7542	**0.7814**	**0.7560**	0.8563	**0.8767**	**0.9702**
		0.9602	**0.7584**	0.7738	0.7545	**0.8568**	0.8740	0.9651
	BorderlineSmote	**0.9569**	**0.6959**	**0.8212**	**0.7446**	**0.8847**	**0.8940**	**0.9610**
		0.9531	0.6684	0.7935	0.7160	0.8641	0.8796	0.9600
	MDO	0.9398	0.6112	0.7592	0.6708	0.8407	0.8562	**0.9710**
		0.9499	**0.6558**	**0.7681**	**0.7041**	**0.8521**	**0.8658**	**0.9710**
	SVMSMOTE	**0.9573**	**0.7860**	0.7857	**0.7689**	**0.8695**	**0.8781**	**0.9692**
		0.9506	0.7230	**0.7891**	0.7377	0.8678	0.8760	0.9612
LGB	Smote	0.9684	**0.8495**	**0.7739**	**0.8028**	**0.8578**	**0.8790**	**0.9797**
		0.9693	0.8473	0.7680	0.7989	0.8496	0.8768	0.9807
	ADASYN	**0.9706**	0.8610	**0.7627**	**0.8036**	**0.8478**	**0.8752**	**0.9814**
		0.9606	**0.8611**	0.6999	0.7618	0.8108	0.8434	0.9775
	BorderlineSmote	0.9586	**0.7276**	0.7912	0.7506	0.8623	0.8817	**0.9743**
		0.9593	0.7169	**0.8261**	**0.7600**	**0.8795**	**0.8978**	0.9734
	MDO	**0.9680**	**0.8054**	**0.7973**	**0.7997**	**0.8704**	**0.8893**	**0.9769**
		0.9651	0.7984	0.7584	0.7754	0.8440	0.8703	0.9737
	SVMSMOTE	**0.9660**	**0.8121**	0.8276	0.8121	0.8942	0.9023	**0.9792**
		0.9640	0.8258	**0.8392**	**0.8179**	**0.8992**	**0.9064**	0.9764

In every two rows, the upper row represents the state obtained using i-cross validation, and the lower row represents the balanced state

Table 3 Compare results on average in optimal and balanced states

	Smote		ADASYN		BorderlineSmote		MDO		SVMSMOTE	
	Optimal	Balanced	Optimal	Balanced	Optimal	Balanced	Optimal	Balanced	Optimal	Balanced
Accuracy	**0.9328**	0.9281	**0.9313**	0.9253	**0.9230**	0.9134	**0.9254**	0.9218	**0.9257**	0.9020
Precision	**0.6831**	0.6700	**0.7163**	0.7040	**0.6214**	0.5846	**0.6455**	0.6363	**0.6733**	0.5997
Recall	0.8183	**0.8249**	**0.7888**	0.7799	0.8061	**0.8147**	**0.7570**	0.7513	0.8475	**0.8549**
F1	**0.7153**	0.7060	**0.7230**	0.7053	**0.6730**	0.6511	**0.6630**	0.6540	**0.7105**	0.6568
Gmean	**0.8667**	0.8649	**0.8508**	0.8435	**0.8570**	0.8554	**0.8286**	0.8231	**0.8827**	0.8733
Auc	**0.8797**	0.8797	**0.8659**	0.8590	**0.8690**	0.8675	**0.8474**	0.8428	**0.8893**	0.8798
CG-mean	**0.9448**	0.9385	**0.9542**	0.9479	**0.9321**	0.9213	**0.9400**	0.9361	**0.9350**	0.9163

will only become higher when *precision* and *recall* reach a high value simultaneously. *F1-measure* is designed to measure minority class performance, while *G-mean* measures the performance of two classes. When *G-mean* is high, the overall performance of the classifier is robust. Another commonly used metric is the area under the receiver operating characteristic curve (ROC), i.e., *AUC*. ROC is generated by plotting the proportion of true positives and the proportion of false positives.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{4}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{5}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{6}$$

$$\text{F1} - \text{measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \tag{7}$$

$$\text{G} - \text{mean} = \sqrt{\frac{TP}{TP + FN} \times \frac{TN}{TN + FP}} \tag{8}$$

Some comparison results between the optimal state and the balanced state are provided in Tables 2 and 3. The average results of five classifiers under eight data sets are shown in Table 2. For each algorithm, the upper row represents the optimal state, and the lower row represents the balanced state. In particular, *CG-mean* is the average validation result for imbalanced classification using icross-validation. Overall, the bold face appeared on the optimal state found using the proposed cross-validation method in most cases. It indicates that, the optimal state found using the proposed cross-validation method is better than the balanced state in most cases. This trend is more obvious in Table 3. Table 3 shows the average results of those experimental results in Table 2 under five classifiers. In the comparison of various algorithms, it can be observed that the optimal state performs better than the balanced state in most cases. In some cases, the advantage of the optimal state obtained by using the *k*-fold icross-validation is considerable, such as that the average *precision* under SVMSMOTE exceeds the balanced state by 7.36% in the optimal state. Due to space limitations, we put other experimental results in the supplementary materials.

5 Conclusion

This paper analyzes the relationship between balanced rate and classification performance in the oversampling process from a novel perspective that sampling may cause the loss of the distribution while the minority class is enhanced using the proposed "distribution difference" measurement. We also present an effective cross-validation framework to find a better state than the balanced state. The experimental results on real-world data sets demonstrate that the proposed cross-validation method can achieve better classification performance that that of balanced state in most cases. In the future, we aim to improve the proposed framework to more efficiently and effectively optimize the balance rate [31].

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s13042-023-01804-x.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 62176033 and 61936001, Key Cooperation Project of Chongqing Municipal Education Commission under Grant No. HZ2021008, and Natural Science Foundation of Chongqing under Grant No. cstc2019jcyj-cxttX0002, National Key Research and Development Program of China under Grant No. 2019QY(Y)0301.

Declarations

Conflict of interest All authors have no conflict of interest to declare

References

1. Chen B, Xia S, Chen Z, Wang B, Wang G (2021) Rsmote: a self-adaptive robust smote for imbalanced problems with label noise. Inf Sci 553:397–428
2. Douzas G, Bacao F, Last F (2018) Improving imbalanced learning through a heuristic oversampling method based on k-means and smote. Inf Sci 465:1–20
3. Alam TM, Shaukat K, Mahboob H, Sarwar MU, Iqbal F, Nasir A, Hameed IA, Luo S (2021) A machine learning approach for identification of malignant mesothelioma etiological factors in an imbalanced dataset. Comput J 65(7):1740–1751
4. Krawczyk B (2016) Learning from imbalanced data: open challenges and future directions. Prog Artif Intell 5(4):221–232
5. López V, Fernández A, Moreno-Torres JG, Herrera F (2012) Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. open problems on intrinsic data characteristics. Expert Syst Appl 39(7):6585–6608
6. Petrides G, Moldovan D, Coenen L, Guns T, Verbeke W (2022) Cost-sensitive learning for profit-driven credit scoring. J Oper Res Soc 73(2):338–350
7. Datta S, Nag S, Das S (2019) Boosting with lexicographic programming: addressing class imbalance without cost tuning. IEEE Trans Knowl Data Eng 32(5):883–897
8. Datta S, Das S (2018) Multiobjective support vector machines: handling class imbalance with pareto optimality. IEEE Trans Neural Netw Learn Syst 30(5):1602–1608
9. Maulidevi NU, Surendro K et al (2022) Smote-lof for noise identification in imbalanced data classification. J King Saud Univ Comput Inf Sci 34(6, Part B):3413–3423
10. Ren J, Wang Y, Cheung Y-M, Gao X-Z, Guo X (2023) Grouping-based oversampling in kernel space for imbalanced data classification. Pattern Recognit 133:108992
11. Sandhan T, Choi JY (2014) Handling imbalanced datasets by partially guided hybrid sampling for pattern recognition. In: 2014 22nd international conference on pattern recognition. IEEE, pp 1449–1453
12. Japkowicz N et al (2000) Learning from imbalanced data sets: a comparison of various strategies. In: AAAI workshop on learning from imbalanced data sets, vol 68. Menlo Park, CA, pp 10–15
13. Bunkhumpornpat C, Sinapiromsaran K, Lursinsap C (2012) Dbsmote: density-based synthetic minority over-sampling technique. Appl Intell 36(3):664–684
14. Zhai J, Qi J, Shen C (2022) Binary imbalanced data classification based on diversity oversampling by generative models. Inf Sci 585:313–343
15. Lunardon N, Menardi G, Torelli N (2014) Rose: a package for binary imbalanced learning. R J 6(1)
16. Barua S, Islam MM, Yao X, Murase K (2012) Mwmote-majority weighted minority oversampling technique for imbalanced data set learning. IEEE Trans Knowl Data Eng 26(2):405–425
17. Arafa A, El-Fishawy N, Badawy M, Radad M (2022) Rn-smote: reduced noise smote based on dbscan for enhancing imbalanced data classification. J King Saud Univ Comput Inf Sci 34(8, Part A):5059–5074
18. Soltanzadeh P, Hashemzadeh M (2021) Rcsmote: range-controlled synthetic minority over-sampling technique for handling the class imbalance problem. Inf Sci 542:92–111
19. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) Smote: synthetic minority over-sampling technique. J Artif Intell Res 16:321–357
20. Rivera WA (2017) Noise reduction a priori synthetic over-sampling for class imbalanced data sets. Inf Sci 408:146–161
21. Sáez JA, Luengo J, Stefanowski J, Herrera F (2015) Smote-ipf: addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering. Inf Sci 291:184–203
22. Han H, Wang W-Y, Mao B-H (2005) Borderline-smote: a new over-sampling method in imbalanced data sets learning. In: International conference on intelligent computing. Springer, pp 878–887
23. Das B, Krishnan NC, Cook DJ (2014) Racog and wracog: two probabilistic oversampling techniques. IEEE Trans Knowl Data Eng 27(1):222–234
24. Abdi L, Hashemi S (2015) To combat multi-class imbalanced problems by means of over-sampling techniques. IEEE Trans Knowl Data Eng 28(1):238–251
25. Xie Z, Jiang L, Ye T, Li X (2015) A synthetic minority oversampling method based on local densities in low-dimensional space for imbalanced learning. In: International conference on database systems for advanced applications. Springer, pp 3–18
26. Zhou H, Dong X, Xia S, Wang G (2021) Weighted oversampling algorithms for imbalanced problems and application in prediction of streamflow. Knowl Based Syst 229:107306
27. He H, Bai Y, Garcia EA, Li S (2008) Adasyn: adaptive synthetic sampling approach for imbalanced learning. In: 2008 IEEE international joint conference on neural networks (IEEE World congress on computational intelligence). IEEE, pp 1322–1328
28. Prati RC, Batista GE, Silva DF (2015) Class imbalance revisited: a new experimental setup to assess the performance of treatment methods. Knowl Inf Syst 45(1):247–270
29. Barella V, Garcia L, de Carvalho A (2018) The influence of sampling on imbalanced data classification. In: 2019 8th Brazilian conference on intelligent systems (BRACIS). IEEE, pp 210–215
30. Thabtah F, Hammoud S, Kamalov F, Gonsalves A (2020) Data imbalance in classification: experimental evaluation. Inf Sci 513:429–441
31. He J, Zhang S, Yang M, Shan Y, Huang T (2020) Bi-directional cascade network for perceptual edge detection. In: 2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)