J Inf Process Syst, Vol.17, No.1, pp.89~106, February 2021
https://doi.org/10.3745/JIPS.01.0065

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Re-SSS: Rebalancing Imbalanced Data Using Safe Sample Screening

Hongbo Shi[*], Xin Chen[*], and Min Guo[*]

Abstract

Different samples can have different effects on learning support vector machine (SVM) classifiers. To rebalance an imbalanced dataset, it is reasonable to reduce non-informative samples and add informative samples for learning classifiers. Safe sample screening can identify a part of non-informative samples and retain informative samples. This study developed a resampling algorithm for Rebalancing imbalanced data using Safe Sample Screening (Re-SSS), which is composed of selecting Informative Samples (Re-SSS-IS) and rebalancing via a Weighted SMOTE (Re-SSS-WSMOTE). The Re-SSS-IS selects informative samples from the majority class, and determines a suitable regularization parameter for SVM, while the Re-SSS-WSMOTE generates informative minority samples. Both Re-SSS-IS and Re-SSS-WSMOTE are based on safe sampling screening. The experimental results show that Re-SSS can effectively improve the classification performance of imbalanced classification problems.

Keywords

Imbalanced Data, Safe Sample Screening, Re-SSS-IS, Re-SSS-WSMOTE

1. Introduction

Rare events (such as cancer in cancer detection [1], financial fraud in financial fraud detection [2], and intrusion events in intrusion detection [3]) are usually difficult to detect owing to their relative scarcity; however, detecting rare events is more critical than detecting ordinary events in many practical problems. Detecting rare events is, in essence, the process of identifying samples in a minority class from an imbalanced dataset. Researchers in the field of imbalanced classification problems have long been focusing on improving the recognition rate of minority samples.

Currently, two main strategies are used to address classification problems with imbalanced data. The first strategy is to change the distribution of various classes in a dataset, and the second is to design or modify learning algorithms to reduce the negative effect of class imbalance. The first strategy can be further classified into undersampling [4-6], oversampling [7-11], and hybrid sampling methods [12-14], which are used to change the class distribution. Undersampling rebalances an imbalanced dataset by removing part of the samples from the majority class. Oversampling selects some samples from the minority class or generates new samples based on the existing minority samples, and then, adds the selected or generated samples into the minority class, thereby obtaining a balanced dataset. Hybrid

Manuscript received December 26, 2019; first revision May 12, 2020; second revision June 5, 2020; accepted June 7, 2020.

Corresponding Author: Hongbo Shi (shb710@163.com)

[*] School of Information, Shanxi University of Finance and Economics, Taiyuan, China (shb710@163.com, cx462240557@163.com, guomin9617@163.com)

sampling transforms an imbalanced dataset into a balanced dataset by adding minority samples via oversampling and reducing the majority samples via undersampling. The second strategy encompasses common methods, such as cost-sensitive techniques [15-17] and ensemble classifiers [18-20]. Cost-sensitive methods assign higher costs to the minority samples so that the learned classifiers can identify more minority samples. The ensemble classifiers divide the majority samples into several subsets, where each subset has a similar size to the minority class. Once several balanced datasets were generated, each can be used to learn a classifier that will be later combined into an ensemble classifier.

This study focuses on the first strategy for handling imbalanced classification problems. The basic objective of this strategy is to change the distribution of datasets and balance the sample size of various classes. This strategy has been featured in recent researches [21-23], which placed more emphasis on the availability of each sample, expecting to incorporate as many informative samples for the classifiers as possible in the balanced dataset. In fact, different classification models have different preferences for samples. For example, classification models based on classification decision boundary are more dependent on the samples near the decision boundary, whereas classification models based on data distribution are more dependent on the overall and local distribution of the samples. Therefore, to obtain informative samples for learning a given classifier, the nature of classification models should be considered before selecting a method for changing the distribution of data.

Support vector machine (SVM) is a classification model based on classification decision boundary, and the learned decision hyperplane is only related to support vectors located near the decision hyperplane. Thus, it is reasonable for decision boundary-based classifiers to employ SVM as a preprocessing method to tackle the problem of imbalanced data. Farquad and Bose [24] employed a trained SVM to preprocess imbalanced data. As a result, more minority samples were correctly predicted without compromising the accuracy of the system. Lin [25] set a regularization parameter for SVM by employing a classification performance. They then used an SVM with the selected regularization parameter as a preprocessor for an imbalanced dataset for further modeling. After the original dataset was balanced using the SVM, the classification ability for the minority samples was improved. Wang [12] learned an SVM decision hyperplane, and resampled an imbalanced dataset in the light of the distance between majority samples and the SVM hyperplane to balance the dataset. Based on the initial hyperplane of SVM, Guo et al. [26] selected key samples of the majority class and learned a final SVM classifier using the key samples of the majority class and all the minority samples.

The regularization parameter in SVM is widely known to be crucial in learning classification hyperplanes. Different regularization parameters produce different classification hyperplanes for a given dataset. Although several studies [12,24,26] realized the importance of regularization parameters for SVM, these methods learned SVM classifiers by setting regularization parameter without specifying an explicit method for its selection. The SVM regularization parameter in [25] was selected using the enumeration method.

Safe sample screening [27] constructs a series of safe sample screening rules using a regularization path algorithm [28]. For each regularization parameter, safe sample screening can identify a part of non-informative samples, and screen them out prior to the training phase without affecting the performance of the classifiers. Safe sample screening has two notable features. First, it can distinguish part of non-informative samples from a given dataset. Second, it can obtain a series of screened datasets corresponding to multiple regularization parameters.

These two features inspired us to employ safe sample screening for handling imbalanced data. As safe sample screening does not consider the characteristics of imbalanced data, we need to solve some problems to apply safe sample screening to imbalanced data. The challenges include the selection of a suitable regularization parameter for obtaining an informative screened dataset from a series of screened datasets and the utilization of a series of screened datasets to generate informative minority samples for oversampling.

In this study, we developed a resampling algorithm, called Re-SSS, for imbalanced datasets based on safe sample screening. The Re-SSS algorithm is composed of Re-SSS-IS and Re-SSS-WSMOTE. The Re-SSS-IS selects a suitable regularization parameter for an imbalanced dataset and employs the screened dataset, corresponding to the suitable regularization parameter, to obtain informative samples from the majority class. The Re-SSS-WSMOTE sets the weight for each sample in the minority class based on a series of screened datasets, then generates informative minority samples based on the weighted minority samples, and finally adds the synthetic samples into the dataset. This study is based on our previous work [29,30]. In [29], the authors applied safe double screening (including sample screening and feature screening) to the higher dimensional imbalanced data, while both [30] and this study merely adopted safe sample screening. Undersampling methods in [29,30] discarded a part of samples in the majority according to the classification performance of learned SVM classifiers, which is time consuming. To improve efficiency, this study set the number of retained minority samples as the criteria of discarding samples. Moreover, this study developed a new oversampling algorithm Re-SSS-WSMOTE, while both [29] and [30] directly used SMOTE.

The main contributions of this study are as follows:

1. A resampling algorithm based on safe sample screening is developed. In this algorithm, the informative samples for the SVM classifier in the majority class are retained and the synthetic minority samples are generated using a series of screened datasets to obtain a balanced dataset.
2. A feasible method of selecting the regularization parameter of the SVM classifier for imbalanced data is developed. This method employs the number of retained samples in the minority class after safe sample screening to select the regularization parameter of the SVM classifier.
3. Experiments are conducted to verify the effectiveness of the developed resampling algorithm and the method of selecting the best regularization parameter.

The rest of this paper is organized as follows. Section 2 introduces the related work of method proposed in this paper with an emphasis on SVM and safe sample screening. Section 3 introduce the developed resampling algorithm in detail. Section 4 presents the experimental datasets and gives the experimental results and analysis. Section 5 draws our conclusion and future work outlook.

2. Related Work

2.1 SVM

Considering a training dataset $\boldsymbol{D} = \{(\boldsymbol{x}_i, y_i) | \boldsymbol{x}_i \in \boldsymbol{R}^d,\ y_i \in \{-1, +1\}, i = 1,2, \dots, n\}$ with d features and n samples, the classification decision function learned by SVM can be expressed as

$$f(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{w}, \tag{1}$$

where $\boldsymbol{w} = (w_1, w_2, \ldots, w_d, w_{d+1})$ is composed of the weight vectors $(w_1, w_2, \ldots, w_d)$ for the learned decision hyperplane and model bias w_{d+1}. To obtain an SVM classifier with fault tolerance capability, a soft margin SVM can be built by solving the following optimization problem.

$$\min_{\boldsymbol{w}} P_\lambda(\boldsymbol{w}) = \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \sum_{i=1}^{n} \ell\big(y_i f(\boldsymbol{x}_i)\big), \tag{2}$$

where $\ell(\bullet)$ is a loss function, and λ is a regularization parameter for controlling the trade-off between the regularization term and the loss term. If a hinge loss function is adopted and a slack variable ξ_i is introduced, (2) can be rewritten as

$$\begin{gathered}\min_{\boldsymbol{w}} P_\lambda(\boldsymbol{w}) = \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \sum_{i=1}^{n} \xi_i, \\ \text{s.t. } y_i f(\boldsymbol{x}_i) \geq 1 - \xi_i, \quad \xi_i \geq 0, i = 1,2,\ldots,n.\end{gathered} \tag{3}$$

The dual problem in (3) can be written as

$$\max_{\boldsymbol{\alpha}} Q_\lambda(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2\lambda}\sum_{i,j=1}^{n} y_i y_j \alpha_i \alpha_j \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle \;\; \text{s.t. } 0 \leq \alpha_i \leq 1, i = 1,\ldots,n. \tag{4}$$

If an n-dimensional vector of Lagrange multipliers $\boldsymbol{\alpha}^* = (\alpha_1^*, \ldots, \alpha_i^*, \ldots, \alpha_n^*)$ is the solution of (4), the classification decision function $f(\boldsymbol{x})$ in (1) can be rewritten as

$$f(\boldsymbol{x}) = \frac{1}{\lambda}\sum_{i=1}^{n} \alpha_i^* y_i \boldsymbol{x}_i^T \boldsymbol{x}, \tag{5}$$

where α_i^* is the Lagrange multiplier corresponding to $(\boldsymbol{x}_i, y_i) \in \boldsymbol{D}$. Based on the optimality conditions of (3) or (4), the samples in $\boldsymbol{D}$ can be categorized into three types: safe sample ($\boldsymbol{SS}$), boundary sample ($\boldsymbol{BS}$), and noise sample ($\boldsymbol{NS}$).

$$\begin{gathered}(\boldsymbol{x}_i, y_i) \in \boldsymbol{SS}\text{:} \;\; y_i f(\boldsymbol{x}_i) > 1 \Rightarrow \alpha_i^* = 0, \\ (\boldsymbol{x}_i, y_i) \in \boldsymbol{BS}\text{:} \;\; y_i f(\boldsymbol{x}_i) = 1 \Rightarrow \alpha_i^* \in (0,1), \\ (\boldsymbol{x}_i, y_i) \in \boldsymbol{NS}\text{:} \;\; y_i f(\boldsymbol{x}_i) < 1 \Rightarrow \alpha_i^* = 1,\end{gathered} \tag{6}$$

The safe sample, located outside the classification margin, is far from the classification hyperplane. For any sample $(\boldsymbol{x}_i, y_i) \in \boldsymbol{SS}$, $\alpha_i^* y_i \boldsymbol{x}_i^T = \boldsymbol{0}$ holds as $\alpha_i^* = 0$. Therefore, the safe sample has no influence on the determination of the decision function $f(\boldsymbol{x})$ in (5). Even if these samples are removed from the training data set, the classification hyperplane would not be changed.

The boundary sample lies on the boundary of the margin, and is near to the classification hyperplane. As α_i^* corresponding to $(\boldsymbol{x}_i, y_i) \in \boldsymbol{BS}$ is a nonzero value, $\alpha_i y_i \boldsymbol{x}_i^T \neq \boldsymbol{0}$ always holds, except for $\boldsymbol{x}_i = \boldsymbol{0}$. This means that the boundary sample is involved in the calculation of the decision function $f(\boldsymbol{x})$, thereby affecting the choice of SVM classification hyperplane.

For the noise sample $(\boldsymbol{x}_i, y_i)$, $\alpha_i y_i \boldsymbol{x}_i^T \neq \boldsymbol{0}$ holds in most cases, except for $\boldsymbol{x}_i = \boldsymbol{0}$. Thus, the noise sample will also affect the decision function $f(\boldsymbol{x})$ in (5). The location of noise sample is related to slack variable ξ_i. If $\xi_i > 1$, the noise sample $(\boldsymbol{x}_i, y_i)$ is located at the other side of its true class about the classification hyperplane; if $0 \leq \xi_i \leq 1$, the noise sample $(\boldsymbol{x}_i, y_i)$ is located between the classification hyperplane and the margin boundary close to the true class. To simplify the problem, all the samples with $\alpha_i = 1$ are known as noise samples.

By analyzing the three types of samples, it was found that the boundary sample should be given more attention for learning the SVM classifier. For imbalanced datasets, the imbalanced ratio (the most common class imbalance metric) is the proportion of the samples between various classes. However, the classification performance of SVM is more dependent on the boundary sample, rather than on all the samples. Therefore, to learn an SVM classifier on an imbalanced dataset, we should focus on the proportion of boundary samples between different classes, rather than the proportion of all the samples between different classes.

In addition, λ in (2) is generally understood as a trade-off parameter for balancing the generalization and fitting performances of SVM. For imbalanced data, the value of λ can influence the position of the classification hyperplane. In general, when the λ value is smaller, the hyperplane moves more toward the majority class; thus, more minority samples can be correctly classified [24]. For example, as shown in Fig. 1, compared with the decision hyperplane (solid line) learned by SVM with $\lambda = 1$, the decision hyperplane (dashed lines) learned by SVM with $\lambda = 0.0001$ moves toward the majority samples (red dots), and three more samples (green stars) of the minority class are correctly predicted. Therefore, to improve the recognition rate of the minority samples, it is necessary to set an appropriate value of λ. However, if the value of λ is selected through enumeration method for learning the best SVM model, it would be time consuming.

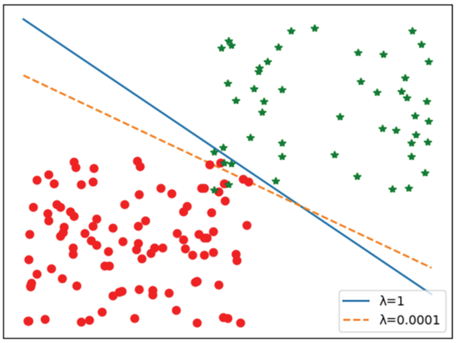


Fig. 1. Comparison of decision hyperplanes constructed using $\lambda = 1$ and $\lambda = 0.0001$ (★: minority samples, •: majority samples).

2.2 Safe Sample Screening

Safe sample screening [27,31] is based on the SVM and regularization path algorithms. Given a dataset, safe sample screening can rapidly screen out parts of the safe samples via safe sample screening rules, and generate a series of screened subsets using a regularization path algorithm.

2.2.1 Safe sample screening rules

Safe sample screening rules can be used to identify non-informative samples. To construct safe sample screening rules, we adopted the objective function of SVM from (2) for safe sample screening. However, owing to the non-differentiability of the hinge loss function at the inflection point, the smooth hinge loss function from (7) was adopted as the loss function of safe sample screening, to ensure that it is differentiable everywhere within the range of values.

$$\ell(y_i f(\boldsymbol{x}_i)) = \begin{cases} 0 & , \quad y_i f(\boldsymbol{x}_i) > 1 \\ \frac{1}{2\gamma}[1 - y_i f(\boldsymbol{x}_i)]^2, & 1-\gamma \le y_i f(\boldsymbol{x}_i) \le 1 \\ 1 - y_i f(\boldsymbol{x}_i) - \frac{\gamma}{2}, & y_i f(\boldsymbol{x}_i) < 1-\gamma \end{cases}, \tag{7}$$

where $\gamma > 0$ is a tuning parameter. The dual problem of safe sample screening can be written as

$$\max_{\boldsymbol{\alpha}} D_\lambda(\boldsymbol{\alpha}) = -\frac{\lambda}{2}\sum_{j=1}^{d}\left(\sum_{i=1}^{n}\frac{1}{\lambda n}x_{ij}\alpha_i y_i\right)^2 - \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\gamma}{2}\alpha_i^2 - y_i\alpha_i\right). \tag{8}$$

Let us assume that $\boldsymbol{w}^* = (w_1^*, \dots, w_d^*, w_{d+1}^*)$ and $\boldsymbol{\alpha}^* = (\alpha_1^*, \dots, \alpha_i^*, \dots, \alpha_n^*)$ represent the optimal solution of the primal and dual problems for safe sample screening, respectively. In the case of smooth hinge loss, according to the Karush-Kuhn-Tucker (KKT) optimality conditions, we can obtain

$$y_i \boldsymbol{x}_i^T \boldsymbol{w}^* = \begin{cases} [1, \infty) & , \quad \alpha_i^* = 0 \\ (1-\gamma, 1) & , \quad \alpha_i^* \in (0,1). \\ (-\infty, 1-\gamma], & \alpha_i^* = 1 \end{cases} \tag{9}$$

Similar to SVM, the samples with $\alpha_i^* = 0$, $\alpha_i^* \in (0,1)$, and $\alpha_i^* = 1$ are called safe samples, boundary samples, and noise samples, respectively. Safe sample screening is aimed at removing parts of the safe and noise samples and retaining all of the boundary samples. To identify safe and noise samples, a solution space $\Theta_{\boldsymbol{w}^*}$ containing the optimal solution $\boldsymbol{w}^*$ is first constructed by employing the feasible solutions of the primal and dual problems. Specifically, for any given feasible solutions $\widehat{\boldsymbol{w}} \in \mathrm{dom}P_\lambda$ and $\widehat{\boldsymbol{\alpha}} \in \mathrm{dom}D_\lambda$,

$$\boldsymbol{w}^* \in \Theta_{\boldsymbol{w}^*} = \left\{\boldsymbol{w} \middle| \|\widehat{\boldsymbol{w}} - \boldsymbol{w}\| \le \sqrt{2[P_\lambda(\widehat{\boldsymbol{w}}) - D_\lambda(\widehat{\boldsymbol{\alpha}})]/\lambda}\right\}. \tag{10}$$

A pair of lower and upper bounds of $y_i \boldsymbol{x}_i^T \boldsymbol{w}^*$ is given as

$$\mathrm{LB}(y_i \boldsymbol{x}_i^T \boldsymbol{w}^*) = y_i \boldsymbol{x}_i^T \widehat{\boldsymbol{w}} - \|y_i \boldsymbol{x}_i\| \sqrt{2[P_\lambda(\widehat{\boldsymbol{w}}) - D_\lambda(\widehat{\boldsymbol{\alpha}})]/\lambda}, \tag{11}$$

$$\mathrm{UB}(y_i \boldsymbol{x}_i^T \boldsymbol{w}^*) = y_i \boldsymbol{x}_i^T \widehat{\boldsymbol{w}} + \|y_i \boldsymbol{x}_i\| \sqrt{2[P_\lambda(\widehat{\boldsymbol{w}}) - D_\lambda(\widehat{\boldsymbol{\alpha}})]/\lambda}. \tag{12}$$

According to (9), (11), and (12), safe sample screening rules can be represented as follows.

Screening rule 1: If $\mathrm{LB}(y_i \boldsymbol{x}_i^T \boldsymbol{w}^*) \ge 1$, then $(\boldsymbol{x}_i, y_i) \in \boldsymbol{SS}$ and $(\boldsymbol{x}_i, y_i)$ can be discarded.
Screening rule 2: If $\mathrm{UB}(y_i \boldsymbol{x}_i^T \boldsymbol{w}^*) \le 1-\gamma$, then $(\boldsymbol{x}_i, y_i) \in \boldsymbol{NS}$ and $(\boldsymbol{x}_i, y_i)$ can be discarded.

Using the above mentioned two rules, the safe and noise samples, which were identified, will be discarded, and the remaining samples will be retained. The advantage of this method is that it can reduce the sample size by employing the relationship between the optimal and feasible solutions, without directly solving the optimization problems.

2.2.2 Regularization path solving strategy

To set a λ value in (2), the authors [28] proposed an SVM regularization path algorithm that can quickly solve all feasible λ values and the corresponding SVM on a given sample set $\boldsymbol{D}$. The initial λ value can be obtained from the original dataset. When the boundary samples of the interval change, each λ_m is solved from its previous λ_{m-1}, and the iteration is continued until there are no samples in the interval or λ is reduced to 0. Owing to the piecewise linearity of the SVM regularization path on regularization parameters, a complete regularization path can be obtained by solving the inflection points of the regularization parameters.

In a given dataset, it is not necessary to solve all the values of λ. In [27], the regularization parameter values in a given range were solved. Safe sample screening only constructs safe sample screening rules corresponding to the regularization parameter values in this range. As the convergence of SVM tends to be faster for larger regularization parameters, the regularization path is computed from larger λ to smaller λ using the warm-start method [27]. In the solving process, the previous optimal solution at λ_{m-1} is used as the initial starting point of the next optimization problem for λ_m. The upper bound λ_{max} and lower bound λ_{min} of the range are as follows:

$$\lambda_{max} = \max_{1 \le j \le d} \left| \frac{1}{n} \sum_{i=1}^{n} x_{ij} y_i \right|; \quad \lambda_{min} = 10^{-4} \lambda_{max}. \tag{13}$$

Given that $\lambda_m \in [\lambda_{min}, \lambda_{max}]$, the values of the upper and lower bounds for $y_i \boldsymbol{x}_i^T \boldsymbol{w}^*$ of each sample can be determined by (11) and (12), respectively. The samples that meet the screening conditions of screening rules 1 and 2 will be removed, and the retained samples are the result of the safe sample screening described in [27].

3. Proposed Algorithm

We developed the Re-SSS algorithm, comprising Re-SSS-IS and Re-SSS-WSMOTE, to change the distribution of imbalanced data based on safe sample screening. The former is used to select a suitable regularization parameter for imbalanced data and obtain informative samples of the majority class, and the latter is used to generate informative minority samples. Notably, both Re-SSS-IS and Re-SSS-WSMOTE can be performed as a part of the Re-SSS, or separately.

3.1 Re-SSS-IS

Safe sample screening can generate a series of screened datasets, each of which contains fewer samples than the original dataset. To find the informative majority samples from an imbalanced dataset using the safe sample screening, two problems should be solved: the first is setting up the range of the regularization parameter values, and the second is selecting the suitable regularization parameter and its corresponding screened dataset.

To solve the first problem, we first analyzed the range of the λ values (see (13)) in [27] and adjusted it for imbalanced data. To simplify the discussion, we assumed that $\left| \frac{1}{n} \sum_{i=1}^{n} x_{ij} y_i \right|$ in (13) reaches the maximum at the j-th attribute and that $x_{ij} = 1$ $(i = 1, \dots, n)$; thus, λ_{max} in (13) can be rewritten as

$$\lambda_{max} = \frac{n_- - n_+}{n}, \tag{14}$$

where n_+ and n_- represent the number of minority and majority samples in the original dataset, respectively. For imbalanced datasets, $(n_- - n_+)$ is usually very large. However, [29] found that, if λ is very large, there will be little or no boundary samples. Thus, the safe samples in the minority class are retained, which is not beneficial for identifying samples in the minority class. To avoid this case, we added a hyperparameter $c < 1$ into (14) (the value of c is given in Section 4.3, which was determined in our experiments), and obtained the maximum value of the regularization parameter λ as

$$\lambda_{max} = c\frac{n_{-} - n_{+}}{n}, \tag{15}$$

A smaller λ_{max} retains more boundary and safe samples in the minority class. In addition, λ_{min} was assigned in the same way as (13); thus, the range of regularization parameter values was $[\lambda_{min}, \lambda_{max}]$.

For the second problem, our solution was to find a classification hyperplane with maximum margin, which could correctly predict as many informative minority samples as possible. For a given regularization parameter λ, the sets of safe, boundary, and noise samples in the minority class were denoted as $\boldsymbol{SS}_{\lambda}^{+}$, $\boldsymbol{BS}_{\lambda}^{+}$, and $\boldsymbol{NS}_{\lambda}^{+}$, respectively. As the noise samples may have a negative effect on the classifier, we expected that only the safe and boundary samples were correctly identified. Hence, we wanted to find a suitable regularization parameter λ^{*}, which has the maximum number of safe and boundary samples. This solution can be expressed as

$$\lambda^{*} = \underset{\lambda}{\operatorname{argmax}} |\boldsymbol{BS}_{\lambda}^{+} \cup \boldsymbol{SS}_{\lambda}^{+}|, \tag{16}$$

First, utilizing the regularization path algorithm, the Re-SSS-IS algorithm quickly obtained a series of feasible λ values, with their corresponding screened datasets. Then, the screened dataset with the largest $|\boldsymbol{BS}_{\lambda}^{+} \cup \boldsymbol{SS}_{\lambda}^{+}|$ was selected, and the corresponding λ was set as the suitable regularization parameter λ^{*}. Lastly, the majority samples in $\boldsymbol{BS}_{\lambda^{*}}$ corresponding to λ^{*} were taken as the set of informative majority samples $\boldsymbol{BS}_{\lambda^{*}}^{-}$.

Re-SSS-IS Algorithm

```
Input: datasets $\boldsymbol{D}$
Output: selected regularization parameter $\lambda^{*}$, informative majority samples $\boldsymbol{BS}_{\lambda^{*}}^{-}$
($\boldsymbol{NS}_{\lambda_1}, \boldsymbol{BS}_{\lambda_1}, \boldsymbol{SS}_{\lambda_1}$),…, ($\boldsymbol{NS}_{\lambda_T}, \boldsymbol{BS}_{\lambda_T}, \boldsymbol{SS}_{\lambda_T}$) ←perform modified safe sample screening on $\boldsymbol{D}$
$\lambda^{*} \leftarrow \text{argmax}_{\lambda_m} |\boldsymbol{BS}_{\lambda_m}^{+} \cup \boldsymbol{SS}_{\lambda_m}^{+}|$
$\boldsymbol{BS}_{\lambda^{*}}^{-} \leftarrow$ majority samples in $\boldsymbol{BS}_{\lambda^{*}}$
```

Modified Safe Sample Screening Algorithm

```
Input: dataset $\boldsymbol{D}$, number of screening $T$, tuning factor $c$, increment $\Delta\lambda$ of $\lambda$
Output: screened datasets ($\boldsymbol{NS}_{\lambda_1}, \boldsymbol{BS}_{\lambda_1}, \boldsymbol{SS}_{\lambda_1}$), ($\boldsymbol{NS}_{\lambda_2}, \boldsymbol{BS}_{\lambda_2}, \boldsymbol{SS}_{\lambda_2}$),…, ($\boldsymbol{NS}_{\lambda_T}, \boldsymbol{BS}_{\lambda_T}, \boldsymbol{SS}_{\lambda_T}$)
$\lambda_{\max} = c\frac{n_{-} - n_{+}}{n}$
$\lambda_m = \lambda_{\max}, m = 1$
while $m < T$
    construct the screening rules with $\lambda_m$
    ($\boldsymbol{NS}_{\lambda_m}, \boldsymbol{BS}_{\lambda_m}, \boldsymbol{SS}_{\lambda_m}$)←using the screening rules screening $\boldsymbol{D}$
    $m = m + 1, \lambda_m = \lambda_{max} * \Delta\lambda$
return ($\boldsymbol{NS}_{\lambda_1}, \boldsymbol{BS}_{\lambda_1}, \boldsymbol{SS}_{\lambda_1}$), ($\boldsymbol{NS}_{\lambda_2}, \boldsymbol{BS}_{\lambda_2}, \boldsymbol{SS}_{\lambda_2}$),…, ($\boldsymbol{NS}_{\lambda_T}, \boldsymbol{BS}_{\lambda_T}, \boldsymbol{SS}_{\lambda_T}$)
```

3.2 Re-SSS-WSMOTE

To generate informative minority samples, this study developed a modified SMOTE algorithm, Re-SSS-WSMOTE, for imbalanced data. SMOTE is a popular oversampling method that generates synthetic samples from existing minority samples. However, not all the minority samples are useful for learning an SVM, and the samples far from the decision hyperplane are more likely to have no effect on learning the classifier. In general, if both the sample and its selected similar sample are boundary samples, the sample generated by combining these two_samples is more likely to be a boundary sample; otherwise, the generated sample will be more likely to be a safe sample. Thus, the availability of a synthetic sample for SVM is related to the availability of the two selected original samples.

Next, we need to consider how to determine the availability of each sample. In Section 2.1, we compared two decision hyperplanes learned by the SVM with $\lambda = 0.0001$ and $\lambda = 1$, and found that the decision hyperplanes learned by the SVM with different λ values may be different. In fact, support vectors of SVM with different λ values may not be exactly similar, as shown in Fig. 2. We can see from Fig. 2 that points 1, 2, 3, and 4 are the support vectors of the SVM with $\lambda = 1$, and points 2, 3, and 4 are the support vectors of the SVM with $\lambda = 0.0001$. Points 2, 3, and 4 are the common support vectors of the SVM for the two different λ values, which means that these points were more likely located closest to the classification hyperplane.

Based on the above analysis, we used the weight value to represent the availability of each sample. The weight value of each sample was calculated based on the screened datasets corresponding to the different λ values. $\boldsymbol{BS}_{\lambda_1}, \boldsymbol{BS}_{\lambda_2}, \dots, \boldsymbol{BS}_{\lambda_T}$ are the boundary sample sets with different λ values; $\boldsymbol{BS} = \boldsymbol{BS}_{\lambda_1} \cup \boldsymbol{BS}_{\lambda_2} \cup \dots \cup \boldsymbol{BS}_{\lambda_T}$ denotes the set of boundary samples for T regularization parameters; and $\boldsymbol{BS}^+ = \{(\boldsymbol{x}_i, y_i) | (\boldsymbol{x}_i, y_i) \in \boldsymbol{BS}, y_i = ` + 1'\}$ is the set of the minority boundary samples. As some samples in the original minority class might not exist in $\boldsymbol{BS}^+$, we adopted a Laplace correction to adjust the weight values to prevent these samples from being selected. The weight value of each minority sample was set as

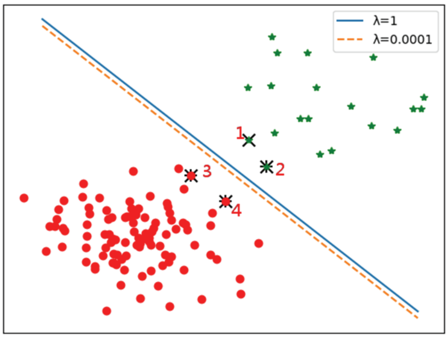


Fig. 2. Comparison of support vectors with $\lambda = 1$ and $\lambda = 0.0001$ (★: minority samples, •: majority samples, +: support vectors of SVM with $\lambda = 0.0001$, x : support vectors of SVM with $\lambda = 1$, ※: common support vectors of SVM with $\lambda = 0.0001$ and $\lambda = 1$).

$$We(\boldsymbol{x}_i, y_i) = \begin{cases} \frac{k_i+1}{\sum_{i=1}^{|\boldsymbol{BS}^+|} k_i + |\boldsymbol{BS}^+|}, \text{if } (\boldsymbol{x}_i, y_i) \in \boldsymbol{BS}^+, \\ \frac{1}{\sum_{i=1}^{|\boldsymbol{BS}^+|} k_i + |\boldsymbol{BS}^+|}, \text{otherwise} \end{cases}, \tag{17}$$

where k_i denotes the number of boundary sample sets containing $(\boldsymbol{x}_i, y_i)$, namely

$$k_i = \sum_{j=1}^{T} I_{\boldsymbol{BS}_{\lambda_j}}(\boldsymbol{x}_i, y_i), \tag{18}$$

$$I_{\boldsymbol{BS}_{\lambda_j}}(\boldsymbol{x}_i, y_i) = \begin{cases} 1, \text{if}(\boldsymbol{x}_i, y_i) \in \boldsymbol{BS}_{\lambda_j}, \\ 0, \text{if}(\boldsymbol{x}_i, y_i) \notin \boldsymbol{BS}_{\lambda_j}, \end{cases} \tag{19}$$

In summary, the Re-SSS-WSMOTE first obtained a series of screened datasets via safe sample screening, and employed the boundary sample sets $\boldsymbol{BS}_{\lambda_1}, \boldsymbol{BS}_{\lambda_2}, \ldots, \boldsymbol{BS}_{\lambda_T}$ to calculate the weight of each sample according to (17). Then, a minority sample was randomly selected according to the weight of the sample, and its similar sample was selected from its k-nearest neighbors according to the weight of the sample. Finally, the linear interpolation method was applied to the two selected samples to generate a synthetic sample. The Re-SSS-WSMOTE algorithm is shown as follows.

Re-SSS-WSMOTE Algorithm

```
Input: dataset $\boldsymbol{D}$, informative majority samples $\boldsymbol{BS}_{\lambda^*}^{-}$, number of nearest neighbors $k$
Output: balanced dataset $\boldsymbol{D}^*$
$(\boldsymbol{NS}_{\lambda_1}, \boldsymbol{BS}_{\lambda_1}, \boldsymbol{SS}_{\lambda_1})$,…, $(\boldsymbol{NS}_{\lambda_T}, \boldsymbol{BS}_{\lambda_T}, \boldsymbol{SS}_{\lambda_T})$ ←perform modified safe sample screening on $\boldsymbol{D}$
$\boldsymbol{D}^+ \leftarrow \{(\boldsymbol{x}_i, y_i) | (\boldsymbol{x}_i, y_i) \in \boldsymbol{D}, y_i = `+1'\}$
$t = |\boldsymbol{BS}_{\lambda^*}^{-}| - |\boldsymbol{D}^+|$
$\boldsymbol{BS}^+ = \{(\boldsymbol{x}_i, y_i) | (\boldsymbol{x}_i, y_i) \in \boldsymbol{BS}_{\lambda_1} \cup \boldsymbol{BS}_{\lambda_2} \cup \ldots \cup \boldsymbol{BS}_{\lambda_T}, y_i = `+1'\}$
calculate $We$ for each minority sample using (17)
for $s$ ←1 to $t$
    a) $(\boldsymbol{x}_{s1}, y_{s1})$ ←select a sample from $\boldsymbol{D}^+$ based on $We$
    b) $(\boldsymbol{x}_{s2}, y_{s2})$ ←select a sample from the $k$ nearest neighbors of $(\boldsymbol{x}_{s1}, y_{s1})$ in $\boldsymbol{D}^+$ based on $We$
    c) $\boldsymbol{x}_s = \boldsymbol{x}_1 + \text{rand}(0,1)(\boldsymbol{x}_{s2} - \boldsymbol{x}_{s1}), y_s = `+1'$
    d) add $(\boldsymbol{x}_s, y_s)$ to generate dataset $\boldsymbol{D}^g$
end for
$\boldsymbol{D}^* \leftarrow \boldsymbol{BS}_{\lambda^*}^{-} \cup \boldsymbol{D}^g \cup \boldsymbol{D}^+$
```

Note that, if only Re-SSS-WSMOTE is used, we will use the original majority sample set $\boldsymbol{D}^-$, instead of the informative majority sample set $\boldsymbol{BS}_{\lambda^*}^{-}$, as the majority sample set in Re-SSS-WSMOTE.

4. Experiments and Analysis

4.1 Datasets

To investigate the effectiveness of the Re-SSS algorithm, we chose 35 datasets from the UCI Repository (http://archive.ics.uci.edu/ml/index.php), LIBSVM datasets (https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/), and KEEL dataset repository (http://www.keel.es/). Except for the original two-class datasets available, we also chose a part of the multiclass datasets and transformed them into two-class imbalanced datasets. For example, for movement-1 with multiple classes, the samples with “1”, “2”, and “3” class labels in the original dataset were combined as the minority class, and the samples with the other class labels were combined as the majority class. In addition, we applied min-max normalization to each dataset for learning the SVM classifiers. Table 1 shows a detailed description of each dataset. The third and fourth columns show the class constituents of the minority and majority classes in each dataset, respectively. The number of minority samples, number of majority samples, number of features, and imbalance ratio of each dataset are listed from the fifth to eighth columns, respectively. The last column presents the source of each dataset.

Table 1. Description of the chosen imbalanced datasets

No	Dataset	Minority class	Majority class	# of minority samples	# of majority samples	# of features	Imbalance ratio	Source of dataset
1	blood transfusion service center	1	0	178	570	4	3.202	UCI
2	breasttissue-1	car	others	21	85	9	4.048	UCI
3	breasttissue-2	adi	others	22	84	9	3.818	UCI
4	diabetes	-1	1	268	500	8	1.866	LIBSVM
5	german	1	-1	300	700	24	2.333	LIBSVM
6	glass-1	1	others	70	144	9	2.057	UCI
7	glass-2	2	others	76	138	9	1.816	UCI
8	glass-3	5,6,7	1,2,3	51	163	9	3.196	UCI
9	glass-4	3,5,6	1,2,7	39	175	9	4.487	UCI
10	heartspectf	0	1	55	212	44	3.855	UCI
11	ionosphere	b	g	126	225	34	1.786	UCI
12	movement-1	1,2,3	others	72	288	89	4	UCI
13	movement-2	4,5,6	others	72	288	89	4	UCI
14	movement-3	7,8,9	others	72	288	89	4	UCI
15	movement-4	10,11,12	others	72	288	89	4	UCI
16	movement-5	13,14,15	others	72	288	89	4	UCI
17	new-thyroid	others	normal	65	150	5	2.308	KEEL
18	pageblocks	others	1	56	492	10	8.786	KEEL
19	pima	positive	negative	268	500	8	1.866	KEEL
20	segment-1	1,2	3,4,5,6,7	660	1650	19	2.5	UCI
21	segment-2	3,4	1,2,5,6,7	660	1650	19	2.5	UCI
22	segment-3	5,6	1,2,3,4,7	660	1650	19	2.5	UCI
23	segment-4	7	1,2,3,4,5,6	330	1980	19	6	UCI
24	shuttle	others	1	469	1706	9	3.638	KEEL
25	svmguide3	1	-1	296	947	22	3.199	LIBSVM
26	svmguide4-1	-1,1	2,-2,3,-3	218	394	10	1.807	LIBSVM
27	svmguide4-2	-2,2	1,-1,3,-3	226	386	10	1.708	LIBSVM
28	svmguide4-3	-3,3	1,-1,2,-2	168	444	10	2.643	LIBSVM
29	vehicle-1	1	2,3,4	212	634	18	2.991	LIBSVM
30	vehicle-2	2	1,3,4	217	629	18	2.899	LIBSVM
31	vehicle-3	3	1,2,4	218	628	18	2.881	LIBSVM
32	vehicle-4	4	1,2,3	199	647	18	3.251	LIBSVM
33	vertebral	NO	AB	100	210	6	2.1	UCI
34	winequality-white	others	5,6	1243	3655	11	2.94	UCI
35	wisconsin	positive	negative	239	444	9	1.858	KEEL

4.2 Performance Evaluation Metrics

For imbalanced classification problem, the suitable metrics should not be dominated by the majority samples. In [32], the impact of class imbalance on classification performance metrics has been systematically studied. The results have shown that the metrics with no bias due to imbalance, *recall, specificity*, *geometric mean* (*G-Mean*), and *area under curve* (*AUC*), are the best performance metrics. As the *specificity* takes into account only the results on the majority class, we did not use it as a performance metric. The *F-score* is also commonly used for imbalanced data. Thus, we compared the different methods by using the four metrics: *recall, F-score, G-Mean*, and *AUC*.

The *recall* measures the ratio of minority samples correctly classified as the minority class to all the minority samples. The range of the *recall* values is [0,1]. The higher the *recall*, the higher the recognition rate of the minority samples is.

The *F-score* is the harmonic mean of the *precision* and *recall*, namely *F-score*=(2**recall***precision*) /(*recall*+*precision*), where the *precision* measures the ratio of minority samples correctly classified to all the samples classified as the minority class. The *F-score* works well for the recognition rate of the minority samples.

The *G-Mean* is the geometric mean of the *recall* and *specificity* with a range of [0,1]. The *specificity* is the actual proportion of majority samples that are correctly identified. The closer the *G-Mean* value is to 1, the better the classification effect is

$$\textit{G-Mean}= \sqrt{recall * specificity} \tag{20}$$

The *AUC* is the area under the Receiver Operating Characteristic (ROC) curve. The range of the *AUC* values is [0,1], and the *AUC* value less than 0.5 indicates that the result is not as good as random prediction. The *AUC* value can well reflect the classification performance of the model.

4.3 Experimental Results and Analysis

In this section, two experiments were performed. The first experiment involved the regularization parameters of the SVM classifiers on the original imbalanced datasets, and the second presented the results of SVM classifiers on the datasets balanced using different methods for changing the distribution of data. The parameters used in the two experiments are as follows: $T = 100$, $c = 10^{-1.6}$, $\Delta\lambda = 10^{0.04-0.04m}$, and $k = 5$.

4.3.1 Experiments on different regularization parameters of the SVM classifiers on the original imbalanced datasets

The main purpose of this experiment was to examine the superiority of the regularization parameter λ^* obtained using the Re-SSS-IS algorithm. First, we applied the Re-SSS-IS algorithm on each original imbalanced dataset, and obtained a suitable regularization parameter λ^* for each dataset. Then, the SVM classifier with λ^* was built directly on the original imbalanced dataset. For comparison, the SVM classifiers with the other 11 regularization parameters were also built.

The experiments were performed using 5-fold cross-validation. In each fold, the original dataset was split into training and test data. The SVM classifier with the corresponding regularization parameter was built on the training data; the *AUC, F-score, G-Mean,* and *recall* of the SVM classifier on the test data was then recorded and averaged across all the splits.

The experimental results are presented in Table 2. The first row lists all the regularization parameters used for the SVM classifiers in the experiments. The second to twelfth columns present the experimental results of the SVM classifiers with different regularization parameters. Each row shows the average performance metrics of the SVM classifiers with the corresponding regularization parameter on the 35 datasets. For example, the average *AUC* of the SVM classifier with λ^* on the 35 datasets was 0.846. As the value of λ^* obtained using the Re-SSS algorithm was different for each dataset, we did not list the value of λ^* in Table 2. From Table 2, it can be seen that the performance metrics of the SVM classifiers were related to the values of λ. When λ was larger, the SVM classifiers had poorer average performance; when λ was smaller, the SVM classifiers obtained better average performance. This result is consistent with the discussion in Section 2.1. However, the values of λ corresponding to the maximum *AUC*, *F-score*, *G-Mean,* and *recall* were not the smallest, in other words,_it is not true that the smaller the value of λ, the better the performance of the SVM classifier. Moreover, each dataset obtained its maximum metrics under different λ values; hence, an appropriate λ value for a given dataset needs to be selected. Furthermore, the SVM classifier with λ^* obtained using the Re-SSS-IS was close to the maximum average metrics. This result shows that the Re-SSS-IS algorithm can select the appropriate value for λ.

Table 2. Comparison of SVM experimental results with different λ on the original datasets

λ	10	1	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	λ^*
AUC	0.538	0.619	0.733	0.805	0.837	**0.847**	**0.847**	**0.847**	0.843	0.837	0.836	0.846
F-score	0.086	0.296	0.556	0.707	0.759	0.770	0.774	**0.775**	0.768	0.761	0.758	0.770
G-Mean	0.090	0.328	0.610	0.767	0.815	0.828	0.833	**0.836**	0.833	0.827	0.825	0.830
recall	0.077	0.248	0.497	0.658	0.730	0.754	0.760	0.767	**0.768**	0.758	0.755	0.761

4.3.2 Experiments on the SVM classifiers with datasets balanced using different data balancing methods

To verify the effectiveness of the developed Re-SSS algorithm, we compared the Re-SSS algorithm with the other methods for changing the distribution of data. The methods proposed in [12,26] did not explicitly mention the adjustment of regularization parameters in SVMs. In [24,25], the SVMs were used for the preprocessing of samples by adjusting the regularization parameters in a similar manner. Aside from preprocessing of samples, feature selection was also used in [25]. However, there was no feature selection involved in our study. Thus, only the Pre-SVM algorithm in [24] was selected as one of the baseline methods in our study. The five baseline methods are described as follows:

- Undersampling: The original dataset is randomly under-sampled; that is, the samples are randomly extracted from the majority class, and the number of samples extracted is equal to the number of minority samples.
- Oversampling: The original dataset is randomly oversampled, that is, randomly extracting samples from the minority class, and adding the samples to the original dataset to balance the dataset.
- SMOTE: Based on the existing minority samples, an interpolation method is used to generate new minority samples, which are added to the original dataset to balance the dataset.
- BorderLine-SMOTE: The improved version of SMOTE uses an interpolation method to generate new small minority samples according to the existing minority class boundary samples, which are added to the original dataset to balance the dataset.

- Pre-SVM [24]: First, this algorithm builds the SVM models on imbalanced data, and the SVM model with the best prediction accuracy is selected and used for prediction purposes. Then, the actual target values of the training samples are replaced by the prediction of the trained SVM.

Table 3. AUC comparison of the experimental results of the Re-SSS

	Undersampling		**Oversampling**		**SMOTE**		**Borderline-SMOTE**		**Pre-SVM**	**Re-SSS**
	λ=0.1	**λ=1**	**λ=0.1**	**λ=1**	**λ=0.1**	**λ=1**	**λ=0.1**	**λ=1**		
blood transfusion service center	0.682	0.679	0.683	0.679	**0.688**	0.676	0.687	0.676	0.520	0.676
breasttissue-1	0.893	0.776	0.889	**0.904**	0.883	0.887	0.899	0.812	0.859	0.846
breasttissue-2	0.947	0.952	0.939	0.952	0.957	0.958	0.964	0.940	0.913	**0.969**
diabetes	0.755	0.724	0.747	0.742	0.741	0.723	**0.762**	0.744	0.715	0.606
german	**0.725**	0.709	0.712	0.709	0.719	0.717	0.717	0.710	0.679	0.579
glass-1	0.733	0.674	0.743	0.684	0.761	0.688	0.733	0.688	0.763	**0.790**
glass-2	0.665	0.589	0.659	0.572	0.675	0.604	**0.679**	0.619	0.577	0.678
glass-3	0.917	0.858	0.916	0.877	0.909	0.858	0.910	0.917	0.878	**0.928**
glass-4	0.625	0.590	0.660	0.604	0.648	0.631	0.649	0.640	0.557	**0.758**
heartspectf	0.770	0.611	**0.786**	0.705	0.779	0.696	0.783	0.703	0.612	0.694
Ionosphere	**0.899**	0.864	0.893	0.872	0.892	0.869	0.890	0.872	0.888	0.808
movement-1	0.822	0.717	0.913	0.813	0.906	0.835	0.874	0.764	0.894	**0.934**
movement-2	0.923	0.896	0.949	0.918	**0.954**	0.934	0.949	0.892	0.953	0.927
movement-3	0.764	0.711	0.853	0.764	0.858	0.770	0.733	0.617	0.823	**0.915**
movement-4	0.758	0.683	0.811	0.737	0.817	0.760	0.824	0.736	0.671	**0.846**
movement-5	0.765	0.713	0.822	0.753	0.798	0.761	0.796	0.754	0.660	**0.855**
new-thyroid	0.815	0.692	0.831	0.715	0.846	0.715	0.860	0.715	0.815	**0.951**
pageblocks	0.852	0.819	0.860	0.836	0.886	0.845	**0.889**	0.819	0.824	0.884
pima	0.755	0.724	0.747	0.742	0.741	0.723	**0.762**	0.744	0.715	0.602
segment-1	0.964	0.888	0.985	0.944	0.984	0.942	0.963	0.856	**0.996**	0.945
segment-2	0.847	0.774	0.871	0.810	0.870	0.817	0.864	0.785	0.877	**0.905**
segment-3	0.868	0.833	0.875	0.855	0.876	0.852	0.862	0.821	0.880	**0.884**
segment-4	**0.995**	**0.995**	**0.995**	**0.995**	**0.995**	**0.995**	**0.995**	0.992	**0.995**	0.980
shuttle	0.949	0.945	**0.960**	0.933	0.958	0.936	0.945	0.912	**0.966**	0.960
svmguide3	0.691	0.598	**0.722**	0.623	0.720	0.649	0.712	0.645	0.635	0.688
svmguide4-1	0.891	0.775	0.899	0.835	0.901	0.829	0.895	0.821	0.939	**0.943**
svmguide4-2	0.789	0.605	0.810	0.680	0.805	0.694	0.821	0.639	0.809	**0.893**
svmguide4-3	0.689	0.631	0.698	0.648	0.707	0.645	0.687	0.654	0.674	**0.774**
vehicle-1	0.733	0.650	0.770	0.676	0.772	0.677	0.774	0.683	0.637	**0.809**
vehicle-2	0.704	0.645	0.787	0.660	0.788	0.665	0.789	0.683	0.680	**0.815**
vehicle-3	0.928	0.762	0.946	0.861	0.941	0.860	0.945	0.899	0.937	**0.951**
vehicle-4	0.95	0.788	0.964	0.852	0.967	0.850	0.963	0.849	**0.971**	0.952
vertebral	0.802	0.754	0.807	0.748	0.821	0.766	0.806	0.755	**0.845**	0.821
wine quality-white	0.689	0.684	0.688	0.684	0.685	0.683	0.680	0.678	0.567	**0.690**
wisconsin	0.969	0.970	0.973	**0.975**	0.970	0.972	0.973	0.974	0.964	0.967

First, four baseline methods were used to rebalance the datasets, and then, the SVM classifiers with the frequently-used regularization parameters (namely 0.1, 1) were performed on the balanced datasets using the four baseline methods. The Pre-SVM and Re-SSS adaptively chose the regularization parameters.

The experimental performance evaluation metrics used were similar to those of the first experiment. The experimental *AUC* results are shown in Table 3, where the first row presents the method used. It can be seen that the Re-SSS method performed optimally on 18 datasets, followed by Borderline-SMOTE (λ = 0.1) and Pre-SVM, which performed optimally on 5 datasets. From Table 3, it is clear that the Re-SSS surpassed the other methods in *AUC*. In addition, the average value for each oversampling method (Oversampling, SMOTE, and Borderline-SMOTE) was better than that for the undersampling.

As the experimental results of the other three metrics were similar to those of the *AUC*, we have not included them here. The comprehensive experimental results are presented in Table 4.

Table 4. Comparison of the experimental results of Re-SSS

		Undersampling		**Oversampling**		**SMOTE**		**Borderline-SMOTE**		**Pre-SVM**	**Re-SSS**
		λ=0.1	**λ=1**	**λ=0.1**	**λ=1**	**λ=0.1**	**λ=1**	**λ=0.1**	**λ=1**		
AUC	Number	3	1	3	3	3	1	5	0	5	18
	Average	0.815	0.751	0.833	0.782	0.835	0.785	0.830	0.772	0.791	0.835
F-score	Number	3	1	3	2	4	1	2	0	10	15
	Average	0.702	0.617	0.733	0.660	0.736	0.665	0.721	0.638	0.681	0.740
G-Mean	Number	4	1	2	3	4	1	4	0	5	18
	Average	0.806	0.730	0.826	0.768	0.829	0.771	0.823	0.735	0.743	0.827
Recall	Number	3	7	3	5	3	5	11	10	1	17
	Average	0.836	0.788	0.842	0.808	0.843	0.811	0.857	0.812	0.628	0.855

In Table 4, the first row presents the method used, and the row named "number" presents the number of datasets for which the SVM classifier with its corresponding regularization parameter obtained optimal performance for a certain evaluation metric. Note that the sum of ten numbers in each row may be greater than 35 as the SVM classifiers with different λ values may have the same results. The row named "average" presents the average obtained by the SVM classifiers with the corresponding regularization parameters on 35 datasets for a certain evaluation metric. It can be seen from Table 4 that, in most cases, the result of $\lambda = 0.1$ is better than that of $\lambda = 1$ when using the same method. With the decrease in the value of λ, the experimental performance of the dataset was more favorable for minority classes. For the four different metrics, the Re-SSS was superior to the other methods in terms of the number of datasets, in which it showed optimal performance and had the highest average value. This verifies the feasibility of the Re-SSS algorithm developed in this study for handling the imbalanced classification problem. In addition, it can be seen from Table 4 that the oversampling method performed slightly better than the undersampling method.

5. Conclusion and Future Work

We developed a resampling Re-SSS algorithm, made up of Re-SSS-IS and Re-SSS-WSMOTE based on safe sample screening, to exploit the informative samples learned by the SVM classifier on an imbalanced data set. The Re-SSS-IS algorithm can select suitable regularization parameters and obtain informative majority samples; the Re-SSS-WSMOTE algorithm is used to generate informative minority samples for the SVM classifier. Then, two experiments were conducted to verify the effectiveness of the algorithm. Compared with the other methods, the proposed resampling method showed better performance. The proposed Re-SSS algorithm can not only discard parts of non-informative samples, but also add useful informative ones. Our future work will focus on developing an effective method for selecting hyperparameter c in the Re-SSS algorithm and exploring how to extend the Re-SSS algorithm to address multiclass imbalanced problems.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (No. 61801279), the Key Research and Development Project of Shanxi Province (No. 201903D121160), and the Natural Science Foundation of Shanxi Province (No. 201801D121115 and 201901D111318).

References

[1] K. Kourou, T. P. Exarchos, K. P. Exarchos, M. V. Karamouzis, and D. I. Fotiadis, “Machine learning applications in cancer prognosis and prediction,” *Computational and Structural Biotechnology Journal*, vol. 13, pp. 8-17, 2015.

[2] D. Sanchez, M. A. Vila, L. Cerda, and J. M. Serrano, “Association rules applied to credit card fraud detection,” *Expert Systems with Applications*, vol. 36, no. 2, pp. 3630-3640, 2009.

[3] R. A. R. Ashfaq, X. Z. Wang, J. Z. Huang, H. Abbas, and Y. L. He, “Fuzziness based semi-supervised learning approach for intrusion detection system,” *Information Sciences*, vol. 378, no. 1, pp. 484-497, 2017.

[4] X. Y. Liu, J. Wu, and Z. H. Zhou, “Exploratory undersampling for class-imbalance learning,” *IEEE Transactions On Systems Man And Cybernetics, Part B (Cybernetics)*, vol. 39, no. 2, pp. 539-550, 2009.

[5] D. Devi, S. K. Biswas, and B. Purkayastha, “Learning in presence of class imbalance and class overlapping by using one-class SVM and undersampling technique,” *Connection Science*, vol. 31, no. 2, pp. 105-142, 2019.

[6] A. Onan, “Consensus clustering-based undersampling approach to imbalanced learning,” *Scientific Programming*, vol. 2019, article no. 5901087, 2019. https://doi.org/10.1155/2019/5901087

[7] H. Han, W. Y. Wang, and B. H. Mao, “Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning,” in *Advances in Intelligent Computing*. Heidelberg, Germany: Springer, 2005, pp. 878-887.

[8] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: synthetic minority over-sampling technique,” *Journal of Artificial Intelligence Research*, vol.16, pp. 321-357, 2002.

[9] M. Koziarski, B. Krawczyk, and M. Wozniak, “Radial-based oversampling for noisy imbalanced data classification,” *Neurocomputing*, vol. 343, pp. 19-33, 2019.

[10] R. Malhotra and S. Kamal, “An empirical study to investigate oversampling methods for improving software defect prediction using imbalanced data,” *Neurocomputing*, vol. 343, pp. 120-140, 2019.

[11] G. Dimic, D. Rancic, N. Macek, P. Spalevic, and V. Drasute, “Improving the prediction accuracy in blended learning environment using synthetic minority oversampling technique,” *Information Discovery and Delivery*, vol. 47, no. 2, pp. 76-83, 2019.

[12] Q. Wang, “A hybrid sampling SVM approach to imbalanced data classification,” *Abstract and Applied Analysis*, vol. 2014, article no. 973786, 2014. https://doi.org/10.1155/2014/972786

[13] Z. Hu, R. Chiong, I. Pranata, Y. Bao, and Y. Lin, "Malicious web domain identification using online credibility and performance data by considering the class imbalance issue," *Industrial Management & Data Systems*, vol. 119, no. 3, pp. 676-696, 2019.

[14] M. Bach, A. Werner, J. Zywiec, and W. Pluskiewicz, "The study of under- and over-sampling methods' utility in analysis of highly imbalanced data on osteoporosis," *Information Sciences*, vol. 384, pp. 174-190, 2017.

[15] N. Japkowicz and S. Stephen, "The class imbalance problem: a systematic study," *Intelligent Data Analysis*, vol. 6, no. 5, pp. 429-449, 2002.

[16] J. Y. Chen, J. Lalor, W. S. Liu, E. Druhl, E. Granillo, V. G. Vimalananda, and H. Yu, "Detecting hypoglycemia incidents reported in patients' secure messages: using cost-sensitive learning and oversampling to reduce data imbalance," *Journal of Medical Internet Research*, vol. 21, no. 3, article no. e11990, 2019. https://doi.org/10.2196/11990

[17] P. A. Alaba, S. I. Popoola, L. Olatomiwa, M. B. Akanle, O. S. Ohunakin, E. Adetiba, O. D. Alex, A. A. A. Atayero, and W. M. A. W. Daud, "Towards a more efficient and cost-sensitive extreme learning machine: A state-of-the-art review of recent trend," *Neurocomputing*, vol. 350, pp. 70-90, 2019.

[18] Z. Sun, Q. Song, X. Zhu, H. Sun, B. Xu, and Y. Zhou, "A novel ensemble method for classifying imbalanced data," *Pattern Recognition*, vol. 48, no. 5, pp. 1623-1637, 2015.

[19] A. Irtazal, S. M. Adnan, K. T. Ahmed, A. Jaffar, A. Khan, A. Javed, and M. T. Mahmood, "An ensemble based evolutionary approach to the class imbalance problem with applications in CBIR," *Applied Sciences*, vol. 8, no. 4, article no. 495, 2018. https://doi.org/10.3390/app8040495

[20] H. He, W. Zhang, and S. Zhang, "A novel ensemble method for credit scoring: adaption of different imbalance ratios," *Expert Systems with Applications*, vol. 98, pp. 105-117, 2018.

[21] D. C. Li, S. C. Hu, L. S. Lin, and C. W. Yeh, "Detecting representative data and generating synthetic samples to improve learning accuracy with imbalanced data sets," *Plos One*, vol. 12, no. 8, article no. e0181853, 2017. https://doi.org/10.1371/journal.pone.0181853

[22] Y. T. Yan, Z. B. Wu, X. Q. Du, J. Chen, S. Zhao, and Y. P. Zhang, "A three-way decision ensemble method for imbalanced data oversampling," *International Journal of Approximate Reasoning*, vol. 107, pp. 1-16, 2019.

[23] M. A. Naiel, M. O. Ahmad, M. N. S. Swamy, J. Lim, and M. H. Yang, "Online multi-object tracking via robust collaborative model and sample selection," *Computer Vision and Image Understanding*, vol. 154, pp. 94-107, 2017.

[24] M. A. H. Farquad and I. Bose, "Preprocessing unbalanced data using support vector machine," *Decision Support Systems*, vol. 53, no. 1, pp. 226-233, 2012.

[25] S. J. Lin, "Integrated artificial intelligence-based resizing strategy and multiple criteria decision making technique to form a management decision in an imbalanced environment," *International Journal of Machine Learning and Cybernetics*, vol. 8, no. 6, pp. 1981-1992, 2016.

[26] T. Guo, J. Wang, Q. M. Liu, and J. Y. Liang, "Kernel SVM algorithm based on identifying key samples for imbalanced data," *Pattern Recognition and Artificial Intelligence*, vol. 32, no. 6, pp. 569-576, 2019.

[27] A. Shibagaki, M. Karasuyama, K. Hatano, and I. Takeuchi, "Simultaneous safe screening of features and samples in doubly sparse modeling," in *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, New York, NY, 2016, pp. 1577-1586.

[28] T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu, "The entire regularization path for the support vector machine," *Journal of Machine Learning Research*, vol. 5, pp. 1391-1415, 2004.

[29] H. Shi, Q. Gao, S. Ji, and Y. Liu, "A hybrid sampling method based on safe screening for imbalanced datasets with sparse structure," in *Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN)*, Rio de Janeiro, Brazil, 2018, pp. 1-8.

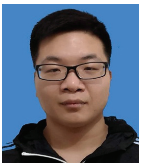
[30] H. Shi, Y. Liu, and S. Ji, "Safe sample screening based sampling method for imbalanced data," *Pattern Recognition and Artificial Intelligence*, vol. 32, no. 6, pp. 545-556, 2019.

[31] K. Ogawa, Y. Suzuki, and I. Takeuchi, "Safe screening of non-support vectors in pathwise SVM computation," in *Proceedings of the 30th International Conference on Machine Learning (ICML)*, Atlanta, GA, 2013, pp. 1382-1390.

[32] A. Luque, A. Carrasco, A. Martin, and A. de las Heras, "The impact of class imbalance in classification performance metrics based on the binary confusion matrix," *Pattern Recognition*, vol. 91, pp. 216-231, 2019.

Hongbo Shi **https://orcid.org/0000-0002-9792-364X**

She is currently a professor in the School of information, Shanxi University of Finance and Economics. She received her Ph.D. degree in School of Computer and Information Technology from Beijing Jiaotong University in 2004. Her main research interests include machine learning, data mining, sparse learning and imbalanced classification.



Xin Chen **https://orcid.org/0000-0002-1949-101X**

He received B.S. degree in Shanxi University of Finance and Economics in 2018. Since September 2018, he has been studying for a master degree in the School of Information from Shanxi University of Finance and Economics. His current research interests include machine learning, data mining and business intelligence.

Min Guo **https://orcid.org/0000-0002-7912-0296**

She received her Ph.D. degree in Shanxi University of Finance and Economics in 2019. She is currently a lecturer in the School of information, Shanxi University of Finance and Economics. Her main research interests include applied statistics, machine learning, data mining, etc.