Postprint: A Minority Class Sample Generation Mechanism with Hyper-rectangle Constraints

Authors: He Zuowei, Tao Jiaqing, Leng Qiangkui, Zhai Junchang, Meng Xiangfu

Date: 2022-05-18T16:08:25+00:00

Abstract

Synthetic Minority Over-sampling Technique (SMOTE) is one of the effective methods for addressing class imbalance problems. However, SMOTE' s linear interpolation mechanism restricts synthetic samples to the line connecting original samples, resulting in a lack of diversity in new samples, and this line may generate noisy samples when it passes through majority class regions. To address the aforementioned issues, a minority class sample generation mechanism with hyper-rectangle constraints is proposed. This mechanism employs hyper-rectangles as the generation region for new samples to replace linear interpolation, thereby increasing the difference between synthetic samples and original samples. Moreover, by detecting whether majority class samples exist within the hyper-rectangle, it determines whether to adjust this hyper-rectangle, thus preventing newly synthesized samples from falling into majority class regions. The proposed mechanism is used to replace linear interpolation and is integrated into three oversampling methods: SMOTE, Borderline-SMOTE, and ADASYN. Experimental evaluations are then conducted on 11 standard datasets from KEEL. The results demonstrate that, compared with the original methods, the integrated methods can help classifiers achieve higher F1-scores and comparable G-mean values. This indicates that the hyper-rectangle generation mechanism can significantly improve the classifier' s ability to recognize minority class samples while also taking majority class samples into account.

Full Text

Preamble

Title: Generation Mechanism for Minority Samples with Hypercuboid Constraints

Authors: He Zuowei[1], Tao Jiaqing[1], Leng Qiangkui[2]†, Zhai Junchang[1], Meng Xiangfu[2]
[1]College of Information Science & Technology, Bohai University, Jinzhou, Liaoning 121013, China
[2]School of Electronics & Information Engineering, Liaoning Technical University, Huludao, Liaoning 125105, China

Abstract: Synthetic Minority Oversampling Technique (SMOTE) is one of the effective methods to solve the class-imbalanced problem. However, the linear interpolation mechanism of SMOTE restricts the synthesized samples to the connecting line of the original samples, resulting in a lack of diversity for new samples, and may generate noisy samples when this line passes through the majority class region. In response to these issues, this paper proposes a generation mechanism for minority samples with hypercuboid constraints. This mechanism constructs a hypercuboid as the generation region of new samples instead of linear interpolation, thereby increasing the variability between the synthesized samples and the original samples. Then, it detects whether there are majority samples in the hypercuboid to determine whether to adjust the hypercuboid, which aims at preventing the new samples from falling into the region of the majority class. This paper integrated the proposed mechanism into three oversampling methods—SMOTE, Borderline-SMOTE, and ADASYN—by using it to replace linear interpolation, and then experimentally evaluated the integrated methods on 11 benchmark datasets from KEEL. The results showed that compared to the original methods, the integrated methods could help the classifier to obtain higher F1 and comparable G-mean. It verifies that the hypercuboid generation mechanism can significantly improve the classifier' s ability to recognize minority samples, and meanwhile the majority samples are also taken into account.

Keywords: imbalanced classification; oversampling technique; SMOTE; generation mechanism; hypercuboid constraints

0 Introduction

The classification of imbalanced data represents a significant challenge in the fields of machine learning and data mining [**?**, **?**]. In binary classification problems, data imbalance means that the number of minority class samples is far smaller than that of majority class samples [**?**, **?**]. This inter-class imbalance causes standard classifiers to become biased, pushing the decision boundary toward the minority class and resulting in some minority class samples being misclassified [**?**]. However, in important application domains such as medical diagnosis [**?**], software defect prediction [**?**], and malignant tumor grading [**?**], the minority class typically contains more critical information [**?**]. Therefore, improving classification performance for minority class samples is a key issue in imbalanced learning.

Current methods for addressing data imbalance can be divided into two categories [?]: algorithm-level methods and data-level methods. Algorithm-level methods emphasize minority class samples by modifying the classifier itself [?, ?]. Data-level methods preprocess the input samples before classifier intervention to reduce the impact of data imbalance [?, ?]. Data-level methods mainly include undersampling and oversampling techniques. Undersampling achieves balance by removing some majority class samples but may lose important distribution information [?, ?]. Oversampling balances the dataset by increasing minority class samples, with the most classic method being the Synthetic Minority Oversampling Technique (SMOTE) proposed by Chawla et al. [?]. SMOTE generates new minority class samples through linear interpolation between original minority class samples, which can improve the classifier' s generalization ability on the test set.

In recent years, many SMOTE-based methods have been proposed, focusing on either inter-class imbalance or intra-class imbalance issues [?]. For inter-class imbalance, Han et al. [?] argued that minority class samples located at class boundaries are more easily misclassified and proposed Borderline-SMOTE, which performs synthetic oversampling only on boundary minority class samples. He et al. [?] proposed ADASYN, an adaptive synthetic oversampling technique that determines the synthesis weight for minority class samples based on the proportion of majority class samples in their neighborhood. However, both Borderline-SMOTE and ADASYN are heavily influenced by the neighbor parameter K; when K takes different values, the distribution of newly synthesized samples shows significant differences. Yan et al. [?] proposed CMOTE, an oversampling technique based on a constructive covering algorithm that selects root samples according to coverage density. However, the setting of two threshold parameters P and D remains an issue requiring discussion. Wang et al. [?] proposed AdaN_{SMOTE}, which adaptively determines the neighbor value for minority class samples based on precision degradation and adjusts neighbor size according to noise and other factors. The new samples synthesized by this method can preserve the distinct clustering characteristics of minority class samples while effectively avoiding the influence of noise, small disjuncts, and complex shapes. Li et al. [?] guided the synthesis of boundary samples by integrating support degree SD and influencing factor posFac, which not only avoids the blindness of sample selection in SMOTE but also comprehensively considers the overall sample distribution. However, the SDRSMOTE algorithm still requires further optimization to improve its operational efficiency.

For intra-class imbalance issues (where minority class samples exhibit multi-cluster distribution [?]), Sheng et al. [?] improved density peak clustering using Box-Cox transformation and criteria, and combined it with the SMOTE algorithm. This method can effectively eliminate various types of noise data, and the obtained clusters are not limited by spatial shape, avoiding subjective interference from manual parameter input. Bunkhumpornpat et al. [?] divided the minority class into multiple arbitrarily shaped sub-clusters and then synthesized new samples between randomly selected minority class samples and sub-cluster centers. However, this method can easily cause inter-class data overlap and cannot effectively identify boundary samples with higher oversampling weights. Nekooeimehr et al. [?] proposed A-SUMO, an adaptive semi-unsupervised weighted oversampling method. After using hierarchical clustering algorithms, it adaptively determines the oversampling size for each sub-cluster. Additionally, A-SUMO achieved good results in identifying boundary samples. However, it should be noted that this method only considers distance factors during clustering, ignoring sample distribution information, resulting in weak anti-noise interference capability. Douzas et al. [?] proposed a heuristic oversampling method based on K-Means and SMOTE, which estimates sampling weights according to the size and density of each cluster. However, the K-Means clustering algorithm cannot find irregularly shaped clusters, and this method does not provide a feasible strategy for determining the optimal number of clusters. Tao et al. [?] used density peak clustering algorithms to improve K-Means' shortcomings in handling intra-class imbalance issues. Based on Euclidean distance and density distribution, the synthesis weights for minority class samples are adaptively calculated, with boundary and low-density samples receiving higher sampling opportunities. Although this method can effectively avoid synthesizing noise data, the setting of the safe distance threshold depends on a parameter γ to be tuned, and its reasonable value range can currently only be obtained through experiments.

In fact, every SMOTE-based method can be decomposed into two mechanisms: a data selection mechanism and a data generation mechanism. The aforementioned methods all improve the data selection mechanism while adopting the same linear interpolation as SMOTE for generating new samples. However, this linear interpolation approach limits the quality of synthesized samples and is also a primary reason why some oversampling methods cannot overcome intra-class imbalance issues [?]. Literature [?] also points out that synthesized new samples should have the ability to expand the minority class region to emphasize the importance of the minority class in the overall data distribution. Particularly, when minority class samples are multi-cluster distributed, linear interpolation performs synthesis operations between clusters, causing new samples to fall into the majority class region and form noise, further aggravating overlap between the two classes [?].

To address the problems existing in the linear interpolation generation mechanism and make new samples more random and diverse, this paper proposes a data generation mechanism with hypercuboid constraints (referred to as the hypercuboid generation mechanism). This mechanism first constructs a hypercuboid using the line connecting a minority class root sample and its selected neighbor as the diagonal; new samples will be generated within this hypercuboid. However, before generation, it is necessary to detect whether majority class samples exist within the hypercuboid. If they do, the hypercuboid is adjusted. Finally, new minority class samples are generated in safe regions without majority class samples. The hypercuboid generation mechanism is an independent module that can replace linear interpolation and be integrated into most

SMOTE-based methods. Next, this paper will first explain the proposed hypercuboid generation mechanism, then embed it into three oversampling methods –SMOTE, Borderline-SMOTE, and ADASYN–and conduct experimental comparisons with the original methods to evaluate the effectiveness of this mechanism.

1.1 SMOTE and Linear Interpolation

SMOTE iteratively searches and selects one sample from the minority class as a root sample, calculates the Euclidean distances from the root sample to other minority class samples, and obtains the k nearest minority class neighbors of the root sample. Then, between the root sample and a randomly selected neighbor, SMOTE uses linear interpolation to synthesize new minority class samples.

Given a minority class sample set X in d-dimensional Euclidean space $\mathbb{R}^d$, assume $x_i \in X$ is the currently selected root sample, and when $k = 5$, the obtained neighbor set is $S_{nn} = \{x_{nn_1}, x_{nn_2}, x_{nn_3}, x_{nn_4}, x_{nn_5}\}$. According to SMOTE' s linear interpolation principle, if x_{nn_2} is randomly selected (Fig. 1(a)), the new sample x_{syn} will be synthesized on the line connecting x_i and x_{nn_2}, that is:

$$x_{syn} = x_i + \varepsilon \times (x_{nn_2} - x_i)$$

where ε is a random number between (0, 1). Intuitively, x_{syn} is restricted to a line segment, and literature [?] also points out that this linear interpolation affects the quality of synthesized new samples. Additionally, if the selected neighbor sample is x_{nn_5} (Fig. 1(b)), the line connecting x_i and x_{nn_5} will pass through the majority class region, and the new sample x_{syn} will be synthesized among majority class samples, resulting in noise generation.

1.2 Generation within Hypercuboid

To address the aforementioned problems of linear interpolation, this paper proposes a hypercuboid generation mechanism to expand the distribution range of minority class samples. Given $x_i \in X$ in $\mathbb{R}^d$, if its neighbor x_{nn_2} is randomly selected (Fig. 2(a)), the new sample x_{syn} will be synthesized within the hypercuboid determined by x_i and x_{nn_2}, that is:

$$x_{syn} = x_i + A \times (x_{nn_2} - x_i)$$

where A is a d-order diagonal matrix, $A = \text{diag}(\alpha_1, \alpha_2, \cdots, \alpha_d)$, and $\alpha_i (i = 1, 2, \cdots, d)$ are random numbers between (0, 1). If the minority class samples are expanded by dimension, $x_i = \{x_i^1, x_i^2, \cdots, x_i^d\}^T$ and $x_{nn_2} = \{x_{nn_2}^1, x_{nn_2}^2, \cdots, x_{nn_2}^d\}^T$ will be represented as:

$$x_{syn} = \begin{pmatrix} x_i^1 + \alpha_1 \times (x_{nn_2}^1 - x_i^1) \\ x_i^2 + \alpha_2 \times (x_{nn_2}^2 - x_i^2) \\ \vdots \\ x_i^d + \alpha_d \times (x_{nn_2}^d - x_i^d) \end{pmatrix}$$

Through Equation (3), it can be seen that compared with linear interpolation, generation within the hypercuboid increases the randomness and distribution range of new samples. However, it is worth noting that x_{syn} still has a certain probability of being synthesized on the line connecting x_i and x_{nn_2}, at which point the hypercuboid generation mechanism degenerates into linear interpolation.

This degeneration probability can be estimated. Assuming α_i contains r decimal places, the probability that the synthesized sample lies on the line connecting x_i and x_{nn_2} is $P = 10^{-r(d-1)}$. From this, it can be obtained that when high-dimensional data is applied, the probability of the hypercuboid generation mechanism degenerating into linear interpolation is very low. For example, when $d = 2$ and $r = 2$, $P = 0.01$; when $d = 2$ and $r = 4$, $P = 0.0001$.

1.3 Noise Prevention Strategy

As shown in Fig. 2(b), when the selected neighbor sample is x_{nn_5}, the hypercuboid determined by x_i and x_{nn_5} overlaps with the majority class region. If new samples are synthesized within this hypercuboid, they will fall among majority class samples and form noise. To avoid synthesizing noise, this paper adds a detection and correction strategy to the hypercuboid generation mechanism. First, it calculates and detects majority class samples falling within the hypercuboid, obtaining $T = \{y_j\}$. Then, it finds the majority class sample y_p in T that is closest to x_i. Finally, it executes the correction strategy, adjusting the initial hypercuboid determined by x_i and x_{nn_5} to a new hypercuboid determined by x_i and y_p, which ultimately serves as the generation region for new samples.

The formal description of this detection and correction strategy is given below. Given the majority class sample set Y, the strategy first detects whether $y_j \in Y$ is located within the initial hypercuboid $G(x_i, x_{nn_5})$. For the t-th dimension of y_j, the judgment criterion is as shown in Equation (5):

$$\min(x_i^t, x_{nn_5}^t) \leq y_j^t \leq \max(x_i^t, x_{nn_5}^t)$$

If every dimension of y_j satisfies Equation (5), it indicates that y_j is located within the initial hypercuboid, and y_j is placed into set T. The above detection steps must traverse every sample in Y. After traversal, if $T \neq \emptyset$, the sample y_p closest to x_i is found from T using Equation (6):

$$y_p = \arg\min_{y_j \in T} \|x_i - y_j\|$$

Then, the correction strategy is used to reconstruct the hypercuboid based on x_i and y_p, and new samples are generated within the corrected hypercuboid. It should be noted that the correction strategy only needs to be executed once to ensure that the corrected hypercuboid does not contain majority class samples. This is because if there exists $y_q (q \neq p)$ falling within the corrected hypercuboid, then Equation (7) holds:

$$\|x_i - y_q\| < \|x_i - y_p\|$$

which clearly contradicts Equation (6).

It should be noted that if the hypercuboid generation mechanism is not treated as an independent module for replacing linear interpolation in SMOTE-based oversampling algorithms, the correction lookup process in Algorithm 1 can be further optimized. We can pre-calculate information about majority class samples contained in hypercuboids formed by any two minority class samples in the training set, and then use this information during each new sample synthesis. For example, if the hypercuboid formed by minority class samples x_i and x_{nn_2} contains majority class samples $\{y_1, y_3, y_7, y_9\}$, it is represented as $G(x_i, x_{nn_2}) \leftarrow \{y_1, y_3, y_7, y_9\}$, indicating that the hypercuboid formed by minority class samples x_i and x_{nn_2} contains majority class samples y_1, y_3, y_7, y_9. Before oversampling, all $G(x_i, x_{nn})$ are calculated, so this information can be directly used when synthesizing new samples, which will greatly shorten the algorithm' s running time.

Notably, based on $G(x_i, x_{nn})$, it is unnecessary to traverse the entire majority class sample set Y. Correspondingly, steps b)~l) in Algorithm 1 can be simplified to a single step: $T \leftarrow G(x_i, x_{nn})$. At this point, the algorithm' s input needs to include a new parameter G, and the algorithm' s time complexity will decrease from $O(|Y|)$ to $O(1)$.

1.4 Algorithm Description

The operational steps of the hypercuboid generation mechanism are shown in Algorithm 1. Steps 4-8 detect whether a majority class sample y_j is located within the initial hypercuboid constructed by x_i and x_{nn}. If so, y_j is placed into set T. Steps 13-15 find the sample y_p in T (when T is not empty) that is closest to x_i. Steps 16-18 synthesize new minority class samples. In specific details, $|T|$ represents the cardinality of set T, and flag serves as a switch for whether y_j is stored in T.

The time complexity of this algorithm can be estimated as $O(|Y|)$, which is higher than the $O(1)$ of linear interpolation. However, since this mechanism needs to be embedded into synthetic oversampling algorithms, and the oversampling process belongs to the data preprocessing stage, which is independent of the classifier, it does not affect the classifier' s training time.

Algorithm 1: Hypercuboid Data Generation Mechanism
Input: Minority class root sample x_i, neighbor x_{nn}, majority class sample set Y
Output: A synthesized minority class sample x_{syn}

```
a) Initialize $T = \emptyset$
b) For $j = 1$ to $|Y|$
c)     flag = 1
d)     For $t = 1$ to $d$
e)         If $y_j^t > \max(x_i^t, x_{nn}^t)$ or $y_j^t < \min(x_i^t, x_{nn}^t)$
f)             flag = 0
g)             Break
h)         End If
i)     End For
j)     If flag == 1
k)         $T \leftarrow T \cup \{y_j\}$
l)     End If
m) End For
n) If $T \neq \emptyset$
o)     $y_p = \arg\min_{y_j \in T} \|x_i - y_j\|$
p) End If
q) For $t = 1$ to $d$
```

r) $x_{syn}^{t} = x_{i}^{t} + \text{random}(0,1) \times (x_{nn}^{t} - x_{i}^{t})$

s) End For

2 Experimental Results and Analysis

The proposed hypercuboid generation mechanism is an independent module that can be embedded into SMOTE-based algorithms to replace linear interpolation and improve the quality of synthesized data. This paper embeds the proposed mechanism into three oversampling algorithms—SMOTE, Borderline-SMOTE (abbreviated as BLSMOTE), and ADASYN—and refers to the embedded algorithms as HC-SMOTE, HC-BLSMOTE, and HC-ADASYN. The effectiveness of this mechanism is then evaluated through experiments on both artificial synthetic datasets and standard benchmark datasets.

2.1 Artificial Synthetic Dataset Experiments

The artificial synthetic datasets are shown in Fig. 3, where minority class samples are represented by red stars and majority class samples by gray circles. Figs. 3(a)(c)(e) show the results of using original SMOTE, BLSMOTE, and ADASYN for oversampling minority class samples, with newly synthesized samples represented by triangles. Figs. 3(b)(d)(f) show the results of using HC-SMOTE, HC-BLSMOTE, and HC-ADASYN for oversampling, with newly synthesized samples represented by diamonds.

From Fig. 3, it can be observed that SMOTE, BLSMOTE, and ADASYN synthesize minority class samples using linear interpolation, and the new samples are all located on the lines connecting original minority class samples, showing an obvious linear distribution. After embedding the hypercuboid generation mechanism, HC-SMOTE, HC-BLSMOTE, and HC-ADASYN synthesize more uniformly distributed minority class samples and expand the distribution range of the minority class. Additionally, Figs. 3(a)(c) show cases where synthesized samples cross the majority class region; these new samples become noise and degrade classifier performance. However, after using the noise prevention strategy in the proposed mechanism, this situation no longer occurs, as shown in Figs. 3(b)(d).

2.2 Standard Dataset Experiments

To ensure objectivity, 11 standard datasets were selected from the KEEL imbalanced database [?] for experiments. The dataset descriptions are shown in Table 1. Each dataset has already been divided into training and test sets using 5-fold cross-validation, and experimental results will report the average of 5 experiments. Experimental parameters are set to default values: SMOTE, BLSMOTE, and ADASYN use neighbor parameters of 5, 5, and 7 respectively when synthesizing samples, and BLSMOTE uses a neighbor parameter of 7 when determining boundary samples. Classifiers used are C4.5 [?] and AdaBoost [?].

Evaluation metrics include F1 and G-mean. F1 is the harmonic mean of Precision and Recall, reflecting the classifier' s ability to classify minority class samples. G-mean is the geometric mean of Sensitivity and Specificity, reflecting the classifier' s ability to balance both classes. These metrics are calculated based on the confusion matrix (Table 2) as follows:

$$\text{Precision} = \frac{TP}{TP+FP}, \quad \text{Recall} = \text{Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{Specificity} = \frac{TN}{TN+FP}, \quad F1 = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}$$

$$G\text{-mean} = \sqrt{\text{Sensitivity} \times \text{Specificity}}$$

Table 3 shows the comparative experimental results between HC-SMOTE and original SMOTE. C4.5 achieved higher F1 on 9 datasets and higher G-mean on 5 datasets, indicating that after HC-SMOTE oversampling, C4.5' s ability to recognize minority class samples was significantly improved, though it was still insufficient in balancing the majority class. AdaBoost achieved higher F1 on all 11 datasets and better G-mean on 8 datasets, demonstrating that HC-SMOTE had a positive impact on AdaBoost.

Table 4 shows the comparative experimental results between HC-BLSMOTE and original BLSMOTE. C4.5 achieved higher F1 on 7 datasets and higher G-mean on 9 datasets, while AdaBoost achieved higher F1 and G-mean on 8 datasets. Since BLSMOTE only oversamples boundary minority class samples during the data selection stage, the excellent performance after combining with the proposed mechanism indicates that synthesizing minority class samples within hypercuboid regions at boundaries can greatly improve the quality of newly synthesized samples and help enhance the classifier' s generalization performance.

Table 5 shows the comparative experimental results between HC-ADASYN and original ADASYN. C4.5 and AdaBoost achieved higher F1 on 11 and 10 datasets respectively, but only achieved higher G-mean on 6 and 3 datasets respectively. ADASYN assigns a synthesis weight to each minority class sample, where the weight increases when more majority class samples are in the neighborhood. After embedding the hypercuboid generation mechanism, HC-ADASYN pays more attention to minority class samples with larger weights, but may cause some majority class samples to be neglected.

Fig. 4 shows boxplots of the above experimental results, where red diamond points represent the mean and green dashed lines represent the median. SM, BD, and AD are abbreviations for oversampling methods SMOTE, BLSMOTE, and ADASYN, respectively. C45 and Ada are abbreviations for classifiers C4.5 and AdaBoost, respectively. From subplots 4(a)(c)(e), it can be seen that the improved methods achieved substantial leads in F1, indicating that the proposed mechanism can significantly improve the classifier’ s recognition of the minority class. Meanwhile, the improved HC-SMOTE and HC-BLSMOTE also outperformed the original methods in G-mean. Overall, the performance was best when the hypercuboid generation mechanism was embedded into Borderline-SMOTE.

3 Conclusion

This paper proposes a novel data generation mechanism to improve synthetic oversampling methods. It uses a hypercuboid as the generation region for new samples instead of linear interpolation to increase the diversity between synthesized and original samples. To prevent new samples from falling into the majority class region, a detection and correction strategy is added to the hypercuboid generation mechanism, thereby avoiding noise generation.

Experiments on standard datasets show that when this mechanism is integrated into three oversampling methods—SMOTE, Borderline-SMOTE, and ADASYN —two standard classifiers achieved higher F1 values on most datasets, demonstrating that the hypercuboid generation mechanism can significantly improve the classifier’ s ability to recognize minority class samples. On the G-mean evaluation metric, the integrated methods performed comparably to the original methods, indicating that while focusing on minority class samples, they can also take majority class samples into account.

This work approaches the problem from the perspective of data generation mechanisms, providing a new research direction for oversampling methods in imbalanced learning. However, the proposed hypercuboid generation mechanism is heuristic, and its effectiveness is established through experimental evaluation. Future work will conduct in-depth theoretical research on the impact of data generation mechanisms on the quality of synthesized samples.

References

[13] Elreedy D, Atiya A F. A comprehensive analysis of synthetic minority oversampling technique (SMOTE) for handling class imbalance [J]. Information Sciences, 2019, 505: 32-64.

[14] Zhu Yuanwei, Yan Yuanting, Zhang Yiwen, et al. EHSO: Evolutionary Hybrid Sampling in overlapping scenarios for imbalanced learning [J]. Neurocomputing, 2020, 417: 333-346.

[15] Fernández A, Garcia S, Herrera F, et al. SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary [J]. Journal of Artificial Intelligence Research, 2018, 61: 863-905.

[16] Wu Yifan, Liang Jiye, Wang Junhong. Classification algorithm based on hybrid sampling for unbalanced data [J]. Journal of Frontiers of Computer Science and Technology, 2019, 13 (2): 342-349.

[17] Chawla N V, Bowyer K W, Hall L O, et al. SMOTE: synthetic minority over-sampling technique [J]. Journal of Artificial Intelligence Research, 2002, 16: 321-357.

[18] Blu T, Thévenaz P, Unser M. Linear interpolation revitalized [J]. IEEE Trans on Image Processing, 2004, 13 (5): 710-719.

[19] Tao Xinmin, Zheng Yujia, Tao Weichen, et al. SVDD-based weighted oversampling technique for imbalanced and overlapped dataset learning [J]. Information Sciences, 2022, 588: 13-51.

[20] Han Hui, Wang Wenyuan, Mao Binghuan. Borderline-SMOTE: a new oversampling method in imbalanced data sets learning [C]// International Conference on Intelligent Computing. Berlin: Springer Press, 2005: 878-887.

[21] He Haibo, Bai Yang, Garcia E A, et al. ADASYN: Adaptive synthetic sampling approach for imbalanced learning [C]// 2008 International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence). New York: IEEE Press, 2008: 1322-1328.

[22] Yan Yuanting, Zhu Yuanwei, Wu Zengbao, et al. Constructive covering algorithm-based SMOTE over-sampling method [J]. Journal of Frontiers of Computer Science and Technology, 2020, 14 (6): 975-984.

[23] Wang Fang, Wu Wentong, Zhang Lili, et al. Research on neighborhood adaptive SMOTE algorithm [J]. Application Research of Computers, 2021, 38 (06): 1673-1677.

[24] Li Kewen, Lin Yalin, Yang Yaozhong. An improved SDRSMOTE algorithm based on Euclidean distance [J]. Computer Engineering & Science, 2019, 41 (11): 2063-2070.

[25] Leevy J L, Khoshgoftaar T M, Bauder R A, et al. A survey on addressing high-class imbalance in big data [J]. Journal of Big Data, 2018, 5 (1): 1-30.

[26] Sheng Kai, Liu Zhong, Zhou Dechao, et al. IDP-SMOTE resampling algorithm for imbalanced classification [J]. Application Research of Computers, 2019, 36 (01): 115-118.

[27] Bunkhumpornpat C, Sinapiromsaran K, Lursinsap C. DBSMOTE: density-based synthetic minority over-sampling technique [J]. Applied Intelligence, 2012, 36 (3): 664-684.

[28] Nekooeimehr I, Lai-Yuen S K. Adaptive semi-unsupervised weighted oversampling (A-SUWO) for imbalanced datasets [J]. Expert Systems with Applications, 2016, 46: 405-416.

[29] Douzas G, Bacao F, Last F. Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE [J]. Information Sciences, 2018, 465: 1-20.

[30] Tao Xinmin, Li Qing, Guo Wenjie, et al. Adaptive weighted oversampling for imbalanced datasets based on density peaks clustering with heuristic filtering [J]. Information Sciences, 2020, 519: 43-73.

[31] Li Yihong, Wang Yunpeng, Li Tao, et al. SP-SMOTE: A novel space partitioning based synthetic minority oversampling technique [J]. Knowledge-Based Systems, 2021, 228: 107269.

[32] Douzas G, Bacao F. Geometric SMOTE a geometrically enhanced drop-in replacement for SMOTE [J]. Information Sciences, 2019, 501: 118-137.

[33] Zhu Tuanfei, Lin Yaping, Liu Yonghe. Improving interpolation-based oversampling for imbalanced data learning [J]. Knowledge-Based Systems, 2020, 187: 104826.

[34] Raghuwanshi B S, Shukla S. SMOTE based class-specific extreme learning machine for imbalanced learning [J]. Knowledge-Based Systems, 2020, 187: 104814.

[35] Moreno-torres J G, Sáez J A, Herrera F. Study on the impact of partition-induced dataset shift on k-fold cross-validation [J]. IEEE Trans on Neural Networks and Learning Systems, 2012, 23 (8): 1304-1312.

[36] Elyan E, Moreno-garcia C F, Jayne C. CDSMOTE: class decomposition and synthetic minority class oversampling technique for imbalanced-data classification [J]. Neural Computing and Applications, 2021, 33 (7): 2395-2406.

[37] Niu Kun, Zhang Zaimei, Liu Yan, et al. Resampling ensemble model based on data distribution for imbalanced credit risk evaluation in P2P lending [J]. Information Sciences, 2020, 536: 120-138.

Note: Figure translations are in progress. See original paper for figures.

Source: ChinaXiv —Machine translation. Verify with original.