Knowledge-Based Systems xxx (xxxx) xxx

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Gaussian prior based adaptive synthetic sampling with non-linear sample space for imbalanced learning☆,☆☆

Tianlun Zhang [a], Yang Li [a], Xizhao Wang [b,*]

[a] College of Information Science and Technology, Dalian Maritime University, Dalian 116026, China
[b] College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China

ARTICLE INFO

Article history:
Received 30 June 2019
Received in revised form 26 September 2019
Accepted 11 November 2019
Available online xxxx

Keywords:
Imbalanced learning
Error bound model
Adaptive method
Classification algorithm
Gaussian mixture model

ABSTRACT

In the presence of skewed category distribution, most learning algorithms fail to provide favorable performance on the representation about data characteristics. Thus learning from imbalanced data is a crucial challenge in the field of data engineering and knowledge discovery. In this work, we proposed an imbalanced learning method to generate minority samples for the compensation of class distribution skews. Different from existing synthetic over-sampling techniques, the data generation is conducted within the hyperplane rather than on the hyperline, thus the proposed method breaks down the ties imposed by the linear interpolation. In addition, this proposed method minimizes the sampling uncertain and risk by integrating a prior knowledge about the minority class instances. Moreover, a multi-objective optimization combined with error bound model develops this proposed method into an adaptive imbalanced learning. Extensive experiments have been performed on imbalanced issues, and the experimental results demonstrate that this method can improve the performance of different classification algorithms.

1. Introduction

With the continuous increase of data availability in academia and industry fields, the algorithms of knowledge discovery and analysis have played essential roles in wide-ranging applications and investigations [1,2]. However, the imbalance problems arise frequently in practical applications, such as fraud detection [3] and fault diagnosis [4]. In such imbalanced data, the minority category is not rare in its own right, however, the size of minority class instances is heavily outnumbered by that of majority class. As a result, the data complexity tends to be amplified by the imbalance class distribution.

When suffering from the complex imbalanced data, most standard learning algorithms are prone to the concept of majority class, and the distributive characteristics of the minority class are poorly learned due to the relatively under-represented data. Consequently, the inductive bias learned from imbalanced data fails to properly represent the minority class concept and leads to the unfavorable classification performance on all classes. Furthermore, the approaches to the construction of classification learning from imbalanced data are desired [5]. In several major conferences and workshops [6–8], a great influx of attention and high activity of advancement have been devoted to the field of imbalanced learning. These valuable milestones facilitated the development of imbalanced learning approaches.

In this community of imbalanced learning, data pre-processing methods have emerged as the popular technique, which aims at adjusting the balance degree of the skewed class distribution. The commonly deployed pre-processing technique is the re-sampling that can be categorized into two conventional methods, i.e., under-sampling and over-sampling [9]. The under-sampling mechanics reduce the population of majority class samples while keeping the original size of the minority class [10]. On the other hand, the over-sampling approaches augment the training dataset by adding a set of minority data, which can be the existing samples in original minority class, the synthetic samples generated by linear interpolation [11,12] or the samples labeled by active learning [13,14]. Motivated by these two re-sampling methods, some joint sampling strategies were proposed to simultaneously alter the sizes of minority and majority data. Typically, the combination is that the under-sampling of majority class is fused by the over-sampling with replacement or synthetic samples of minority class [11]. In addition, a more effective joint strategy employs

☆ No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.knosys.2019.105231.

☆☆ This work is supported by the National Natural Science Foundation of China (No.61976141, No.61732011, No.61772344, No.61811530324.)

* Corresponding author.
E-mail address: xizhaowang@ieee.org (X. Wang).

https://doi.org/10.1016/j.knosys.2019.105231
0950-7051/

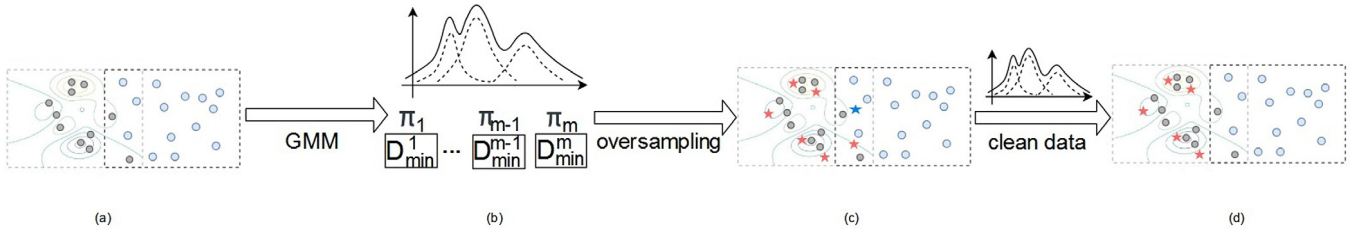

Fig. 1. Illustration of the proposed method. (a) is the imbalanced data, the contours denote the distribution about minority class. (b) is the GMM, $D^i_{\min}$ is the ith subset of minority dataset which is divided into m groups. π_i denotes the importance degree that $D^i_{\min}$ contributes to the minority class concept. (c) is the result of over-sampling, the red stars are the synthetic minority samples, and blue circle is the synthetic data with wrong label, (d) is the result of data clean. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

ensemble learning to fuse different but related supervised learnings on sampling results, which are obtained by operating various sampling methods on one imbalanced dataset [15,16].

Studies have shown that the sampling methods do indeed aid in the compensation of skewed class distribution, and make classification algorithms pay more attention on minority concepts [2]. These academic achievements have provided an immense opportunity to make these methods play an important role in a broad range of applications. One of pioneering works was conducted by Solberg et al. [17], they addressed the imbalanced problem in oil slick detection via a joint sampling strategy. Recently, Yang et al. [18] found that the high-impact bug reports are rare cases in bug triage systems, and they discussed the effect of different sampling methods in the modification of class distribution. The similar methods were also applied into the recognition of good successor positions which have low proportion in chess game records [19]. Besides, imbalanced learning also has yielded a large number of successful consequences on other domains, such as gene recognition [20] and tool condition monitoring [21]. These consequences imply that a balanced dataset can provide improved overall classification result compared to the imbalanced version.

Despite impressive consequences and wide applications, there still exist several crucial challenges in developing a robust sampling methodology. Essentially, on one hand, an inherent problem is that most sampling methods can result in higher data complexity, such as the data with overlapping, missing and redundant information. On the other hand, a common problem is that the adjusted dataset tends to have a tied distribution, which can cause the learned rules to become too specific in certain subconcepts of the original data. Consequently, sampling methods have potential risk leading to the immense hindering effects on classification learning [2]. The reason causing these limitations mainly roots in the sampling uncertain, which will be further analyzed in the following sections. Generally, to deal with these problems, much expert experience have to be devoted into the parameter setting to control the sampling process. However, for various imbalanced issues, it is difficult to find a shared empirical agreement between the setting of parameters and the favorable classification performance, thus it is necessary to develop a self-adaptive method that can update parameters according to the imbalanced issue at hand. Based on these aforementioned analyses, our focus is to conduct some further investigations on sampling method for imbalanced learning. Fig. 1 shows the entire imbalanced learning proposed in this paper. More concretely, our work can be concluded as follows.

(1) We improve the synthetic over-sampling method by avoiding tied regions in the distribution of adjusted dataset. To this end, the new synthetic samples are generated in a rational feature space with the same dimension as the minority data. Thus the existing synthetic over-sampling approaches are the special cases of this proposed method in terms of data generation.

(2) We proposed an adaptive Gaussian mixture model (GMM) based data sampling and cleaning techniques. The prior information about minority data is provided by GMM, which greatly improves the robustness to outliers and noise. Meanwhile, this learned prior knowledge is used as the guideline to eliminate the unexpected synthetic samples, such as the data at borderline or with incorrect annotation, which have low agreement with the distributive characteristics of minority class, and depreciate the classification performance.

(3) We formulate a multi-objective optimization to solve the unknown hyper-parameters in this imbalanced learning, where a novel fitness evaluation is proposed to assess the quality of re-balanced dataset. This is the first endeavor of such technique incorporating an evaluation measure into the sampling approach, and it is interesting to show the relationship between generalization ability and data class distribution. In a broad experimental sweep, our method can effectively improve the performance of various learning algorithms on extensive imbalanced issues.

2. Preliminaries

In this section, we will provide the foundation for the following discussion. Firstly, we present the popular sampling methods and data clean techniques in the field of imbalanced learning. Then we recall the basic concepts related to extreme learning machine (ELM) [22,23] used as the classification mechanism in our method. To ease the following presentation, some notations are established here. Given a training dataset D with N samples (i.e., $|D| = N$), we have definition as: $D = \{(x_i, t_i)\}$, $i = 1, \ldots, N$, where $x_i = (x_{i1} \cdots, x_{in})^T$ is the ith sample with n-dimensional feature, $t_i \in \{0, 1\}^{C\times 1}$ is a one-hot vector that denotes the category identity label of x_i, without losing generality, we consider the binary-class problem, namely $C = 2$. In this case, the set of minority class samples is defined as $D_{\min}$, and the majority class set is D_{maj}, so that $D_{\min} \cap D_{maj} = \Phi$, $D_{\min} \cup D_{maj} = D$ and $|D_{\min}| \ll |D_{maj}|$.

2.1. Review of sampling methods for imbalanced learning

Typically, the most intuitive solution for imbalanced issues is to adjust the imbalance ratio (IR) between minority and majority classes [2]. Based on this intention, various sampling approaches have been developed to modify the distributive characteristics of imbalanced data, such as random under-sampling (RUS [24]), random over-sampling (ROS [25]) and synthetic minority oversampling technique (SMOTE [11]). In particular, as shown in Fig. 2(b-1), RUS randomly selects a subset E_{maj} from D_{maj}, and then removes E_{maj} from the original dataset D so that $|D| = |D_{\min}| + |D_{maj}| - |E_{maj}|$. On the contrary, ROS aims at augmenting the size of minority class, to this end, a randomly subset $E_{\min}$ in $D_{\min}$ is

Please cite this article as: T. Zhang, Y. Li and X. Wang, Gaussian prior based adaptive synthetic sampling with non-linear sample space for imbalanced learning, Knowledge-Based Systems (2019) 105231, https://doi.org/10.1016/j.knosys.2019.105231.

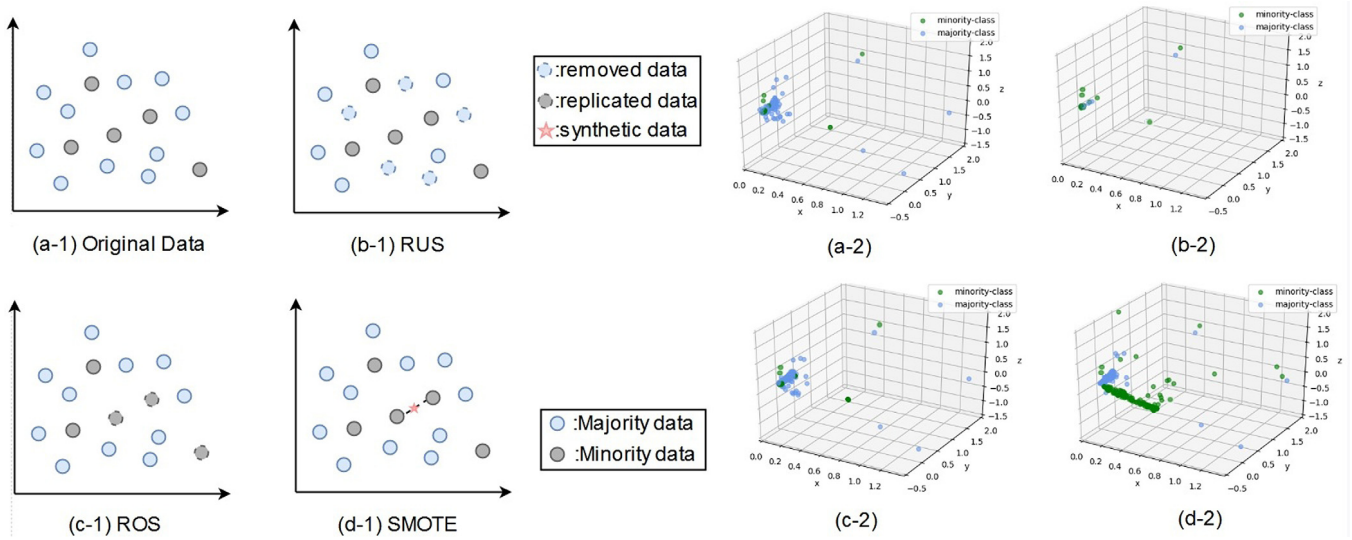

Fig. 2. (a-1), (b-1), (c-1), (d-1) are the illustrations of different sampling methods. The removed data are randomly selected majority samples, the replicated data are randomly selected minority samples, the synthetic data are new minority samples generated on the line linking two near neighbors in minority class. One can refer to these case studies, i.e., (a-2), (b-2), (c-2) and (d-2), for the sampling results.

replicated and added into D, as shown in Fig. 2(c-1). In this way, the size of $D_{\min}$ is increased by $|E_{\min}|$. In the procedure of SMOTE, the size of $D_{\min}$ is augmented by artificial data instead of existing minority data. As illustrated in Fig. 2(d-1), a synthetic data (the star point) can be randomly generated on the line connecting one specified minority sample with its nearest neighbor. By doing so, the category distribution balance of D is adjusted accordingly.

Many studies [26,27] have justified that sampling methods are useful for the improvement of classification on imbalanced data. However, the inherent uncertain caused by random sampling makes the drawbacks of these methods be relatively obvious. In the case of RUS, the important and insignificant instances of majority class can be removed with the same probability, thus some important concepts about the majority class will be missed when removing samples from D_{maj}. In regards to ROS, the noise and normal minority samples have the same opportunity to augment the size of minority class, thus this method has a weak ability of anti-noise; besides, multiple copies of the same samples increase the redundant data, and tend to cause the tied distribution in certain regions of $D_{\min}$. SMOTE also has some inherent limitations which will be analyzed in the following section. In particular, these drawbacks often result in problematic consequences which potentially hinder the imbalanced learning [28,29].

Thus, to overcome these limitations, some improved sampling approaches have been proposed, especially, the data clean techniques. Eliminating Tomek links [30] is an effective data clean technique to remove the overlapping caused by sampling methods. A Tomek link is a sample pair (x_i, x_j), in which $x_i \in D_{\min}$, $x_j \in D_{maj}$, and x_j is the nearest neighbor of x_i, vice versa, as shown in Fig. 3(c). If two samples form a Tomek link, either one in this sample pair is noise or both are close to the border. Thus we can get a clear decision boundary by removing all Tomek links. Some similar work in this area includes the integration method of condensed nearest neighbor rule and Tomek Links [31], the neighbor data clean rule based on edited nearest neighbor (ENN) [32], the joint method SMOTE+ENN and the SMOTE+Tomek links [31].

2.2. Review of extreme learning machine

Extreme learning machine (ELM) is an efficient algorithm to train single hidden layer feed-forward networks (SLFNs), as shown in Fig. 4. Mathematically, a SLFN with M latent neurons can be formulated as

$$f_\theta(x_i) = \sum_{j=1}^{M} \beta_j \sigma(w_j x_i + b_j), \qquad \sigma(w_j x_i + b_j) = \frac{1}{1 + \exp[-(w_j x_i + b_j)]}, \tag{1}$$

in which $w_j = (w_{j1}, w_{j2}, \ldots w_{jn})$ is a weight vector connecting the observation vector x_i with the jth latent neuron, which is connected to the output layer via $\beta_j = (\beta_{j1}, \beta_{j2}, \ldots, \beta_{jC})^T$. b_j is the threshold associating with the jth hidden node. σ is a nonlinear activation, here the sigmoid function. $\theta = (w, b, \beta)$ is the parameter set where the hidden weights and thresholds (w, b) are randomly generated, and $\beta = [\beta_1, \beta_2, \ldots \beta_M]^T$ is the output weight matrix. When given stochastic latent parameters, the compact format of hidden output H is

$$H = \begin{bmatrix} \sigma(w_1 x_1 + b_1) & \cdots & \sigma(w_M x_1 + b_M) \\ \sigma(w_1 x_2 + b_1) & \cdots & \sigma(w_M x_2 + b_M) \\ \vdots & \ddots & \vdots \\ \sigma(w_1 x_N + b_1) & \cdots & \sigma(w_M x_N + b_M) \end{bmatrix}_{N \times M} \tag{2}$$

and the actual output matrix $Y = [y_1, y_2, \ldots y_N]^T \in \Re^{N \times C}$ is

$$Y = H\beta. \tag{3}$$

As such, the training objective is to minimize the cumulative training error which can be described as

$$Minimize: \sum_{i=1}^{N} \|y_i - t_i\|^2. \tag{4}$$

The output weight matrix can be solved through the smallest norm least-squares solution of above mentioned equation,

$$\hat{\beta} = H^{\dagger} T. \tag{5}$$

Here $T = [t_1, t_2, \ldots t_N]^T$ is the target output matrix, $H^{\dagger}$ is the Moore–Penrose generalized inverse of H. Through Eqs. (1) and (2), one can note that the hidden features are determined by the stochastic model parameters (w, b) and the input data, meanwhile, β is related with the hidden features H and the expected

Please cite this article as: T. Zhang, Y. Li and X. Wang, Gaussian prior based adaptive synthetic sampling with non-linear sample space for imbalanced learning, Knowledge-Based Systems (2019) 105231, https://doi.org/10.1016/j.knosys.2019.105231.

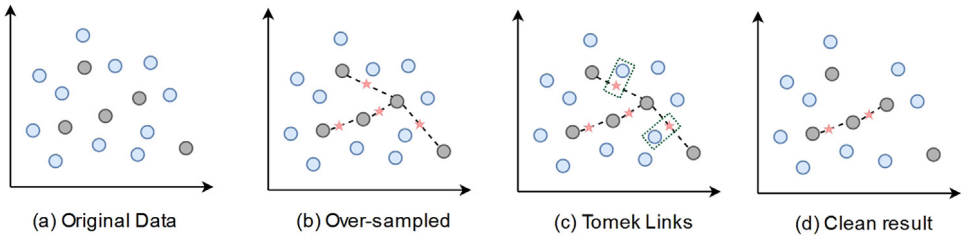

Fig. 3. Clean technique with Tomek Links. (a) is the original distribution of a dataset. (b) is the augmented data by linear interpolation, and the red stars are the synthetic samples. The green frames in (c) are Tomek Links which are removed for clear borderline as shown in (d). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

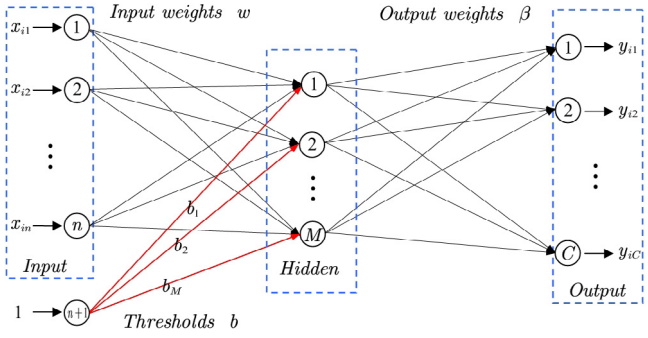

Fig. 4. An illustration of single hidden layer feed-forward network. This network has three layers, i.e., input, hidden and output layers with $n+1$, M and C nodes, respectively. The value of the $(n+1)$th node in input layer is fixed as one, and is connected with the thresholds. The arrows indicate the direction of data propagation.

output T. Thus, if the hidden parameters (w, b) are fixed, β only depends on the training data.

3. Discussion of the proposed method

In this section, we propose a novel over-sampling approach to create synthetic samples for minority class. The overall framework of this proposed imbalanced learning includes data generation and clean technique, in which a reasonable samples distribution, guided sampling method and self-adaptive manner are utilized as the basic elements to ensure the robustness of the proposed method in different imbalanced issues.

3.1. Sample generation and clean for imbalanced learning

To adjust the proportion of balance between $D_{\min}$ and D_{maj}, the synthetic over-sampling mechanism naturally follows from its description by appending an additional minority class set E_{syn} into $D_{\min}$. Consequently, the adjusted D with the increasing size of $D_{\min}$ is expected to have enhanced concept representation about minority class. However, the region of the minority class will be tied if the additional samples are created by linear interpolation (as illustrated by Fig. 2 (d-2)). As a result, the inductive bias becomes too specific for the tied region; in essence, overfitting. Reasonable data distribution thus is a fundamental issue for over-sampling methods. Depending upon the characteristic of sample space, we employ a new data generation approach to create synthetic examples in the feature space rather than the data space. Concretely, the feature space corresponding to a minority class sample under consideration can be defined as:

$$\Omega = \{x_{syn} | 0 \le |x_{syn} - x_\tau| \le R\}, \tag{6}$$

where $x_\tau \in D_{\min}$ is considered as the sampling kernel, $R = |\hat{x}_\tau - x_\tau|$, $\hat{x}_\tau \in D_{\min}$ is the nearest neighbor of x_τ. In our work, the ith dimension of one synthetic sample x_{syn} can be derived in what follows

$$x_{syn,i} = x_{\tau i} + \alpha_i |\hat{x}_{\tau i} - x_{\tau i}|, \tag{7}$$

in which $\alpha_i \in [-1, 1]$ is a random value that determines the position of $x_{syn,i}$ relative to $x_{\tau i}$. For clear explanation, we illustrate a special case with $n = 3$ (see Fig. 5), here Ω is a sphere centered at the sampling kernel $x_\tau = (x_{\tau 1}, x_{\tau 2}, x_{\tau 3})$, the direction $(-, +, -)$ and the step (R_x, R_y, R_z) jointly contribute to the synthetic sample $x_{syn} = (\hat{x}_{\tau 1} - R_x, \hat{x}_{\tau 2} + R_y, \hat{x}_{\tau 3} - R_z)$. Under the control of these random values, the new sample x_{syn} can be generated at random point in Ω. The random data generation plays an important role to avoid the tied distribution in $D_{\min}$. Consequently, this method can effectively force the decision regions of the minority category to be more general.

In the data generation mentioned above, the feature space has been extended from line segment to hyperspace, however, the selection of sampling kernel x_τ is a crucial remaining issue. In most of over-sampling approaches, the sampling kernels are randomly selected from $D_{\min}$, in practice, this blind sampling mechanic often accompanies with uncertain and risk, because the noise data can be sampled with the same opportunity as the normal instances of minority class. Thus it is necessary to develop a guided sampling method for the data creation. To this end, a prior based sample selection is proposed in the following discussion.

Generally, $D_{\min}$ contains disjunct clusters, as shown in Fig. 6(a), each cluster is composed of several minority class samples. Focusing our discussion on the prior, we define these minority class sub-concepts by using the weighted mixture of multivariate Gaussian model, i.e.,

$$G(x|\mu, \Sigma, \pi) = \sum_{i=1}^{m} \pi_i g(x|\mu_i, \Sigma_i). \tag{8}$$

Here, GMM assumes that $D_{\min}$ has m clusters, the ith cluster can be represented by a Gaussian model as follows

$$g(x|\mu_i, \Sigma_i) = \frac{1}{\sqrt{2\pi|\Sigma_i|}} \exp[(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i)], \tag{9}$$

in which μ_i indicates the mean, the covariance matrix Σ_i captures the variance of each dimension, as well as the covariance between any two dimensions of samples. The component weight π_i denotes the importance degree that the ith cluster contributes to the overall distributive characteristics of $D_{\min}$, such that $1 = \sum_{i=1}^{m} \pi_i$. We use $\pi = (\pi_1, \pi_2, \ldots \pi_m)$ to guide the selection of minority class clusters. In this way, the dominant clusters of minority class are more likely selected than the outliers. When given a selected cluster, a membership degree set $\lambda = (\lambda_1, \lambda_2, \ldots \lambda_{|D_{\min}|})$ can be derived according to the GMM, the element λ_i in λ denotes the likelihood that minority sample x_i belongs to this cluster. We use λ as the sampling factor for a guided sampling process in the selected cluster. By doing so, the instances conflicting with this

Please cite this article as: T. Zhang, Y. Li and X. Wang, Gaussian prior based adaptive synthetic sampling with non-linear sample space for imbalanced learning, Knowledge-Based Systems (2019) 105231, https://doi.org/10.1016/j.knosys.2019.105231.

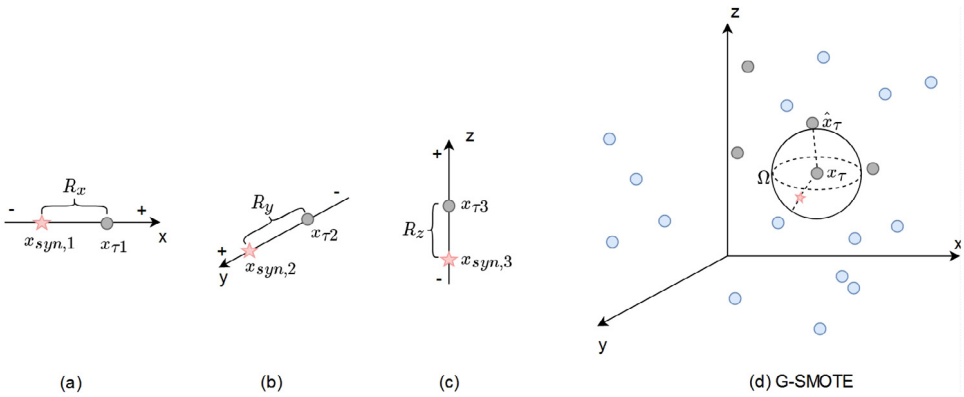

Fig. 5. The illustration of G-SMOTE. In this case, all samples have 3 dimensions. $x_\tau = (x_{\tau 1}, x_{\tau 2}, x_{\tau 3})$ and $\hat{x}_\tau = (\hat{x}_{\tau 1}, \hat{x}_{\tau 2}, \hat{x}_{\tau 3})$ are sampling kernel and its nearest neighbor, respectively. (a), (b) and (c) shows the positions of the synthetic sample in different dimensions, i.e., $x_{syn,1}$, $x_{syn,2}$ and $x_{syn,3}$. + and − denote the directions. And the distance in different dimensions can be represented as $R_x = |\alpha_1 \times |\hat{x}_{\tau 1} - x_{\tau 1}||$, $R_y = |\alpha_2 \times |\hat{x}_{\tau 2} - x_{\tau 2}||$, $R_z = |\alpha_3 \times |\hat{x}_{\tau 3} - x_{\tau 3}||$.

cluster will have lower chance to be selected as the sampling kernels. Afterwards, the synthetic examples can be generated in these Ω of selected sampling kernels, as discussed by Eqs. (6) and (7). In this case, the examples in E_{syn} are rarely attributed to noise or outliers, and can represent the main concept of minority category.

After the data generation mentioned above, we can get the set of synthetic samples. However, the synthetic sampling is conducted without consideration to neighboring examples of opposite classes. Thus the new samples of minority class often cause its decision boundary to spread into the other class regions. For clear category borderline, a data cleaning technique based on GMM prior is proposed to deal with the overlapping between classes. To this end, the $G(x|\mu, \Sigma, \pi)$ learned in data generation is used to filter synthetic samples. For one minority class cluster, we measure the importance degree of all synthetic samples by deriving the membership degree set $\lambda_{syn} = (\lambda_{syn,1}, \lambda_{syn,2}, \ldots \lambda_{syn,|E_{syn}|})$. The lower $\lambda_{syn,i}$ indicates that the ith synthetic sample has poor agreement with the statistical characteristics of this cluster. Thus, in each cluster, the synthetic samples with greater importance degree are selected to augment $D_{\min}$. By doing so, the remaining synthetic samples effectively avoid the occurrence of overlapping, while enhancing the representation with respect to different sub-concepts of minority class.

Algorithm 1 presents the entire steps about the synthetic over-sampling based on GMM prior (G-SMOTE), the parameters (μ, Σ, π) are learned by Expectation–Maximization algorithm.

3.2. Self-adaptive framework for synthetic over-sampling technique

In the previous section, we discussed an improved synthetic over-sampling approach based on GMM prior. Although theory of this approach is appealing, technically speaking, several hyper-parameters in Algorithm 1 are usually unknown and hard to choose in practice. To be more actionable, this over-sampling approach is formulated as an optimization problem with respect to the unknown inputs of Algorithm 1. In the following discussion, we use a population-based evolutionary algorithm, namely differential evolution (DE) [33], to iteratively solve the multi-objective optimization problem. By doing so, this over-sampling approach can deal with different imbalanced issues in a self-adaptive way (as illustrated in Fig. 7). The adaptive method is summarized in Algorithm 2, in which some crucial steps are discussed in what follows.

Chromosome Encoding: In DE, a chromosome is the candidate solution consisting of variables to be optimized, in this case, the sampling ratio ($n1$), the size of synthetic samples in each Ω ($n2$), the ratio of remaining samples ($n3$) and the size of Gaussian components (m). $n1$, $n2$ and $n3$ jointly control the balance degree of the adjusted dataset. m is used to fit the number of minority class clusters that is unknown in advance. As shown in Fig. 6, some sub-concepts of minority class will be under-represented if m is lower than the practical number of minority class clusters, on the contrary, the sampling result will be over-represented with respect to some specific clusters. In this step, these four parameters are directly encoded as genes with numerical type for one chromosome ($n1, n2, n3, m$). And a set of chromosomes are simultaneously initialized to represent different cases of parameters setting. The final evolved chromosome is the most appropriate setting for over-sampling process.

Differential Evolution Operators: The optimization of chromosomes is driven by a sequence of three evolution operators, namely, mutation, crossover and selection. For each chromosome, the mutation operator determines the updating direction and step. Then the crossover operator decides whether the new genes in updated chromosomes are retained or not. Afterwards, the selection operator chooses the set of genes with better fitness degree to the over-sampling task.

Fitness Evaluation: As mentioned above, a better fitness degree is the goal of evolution process. In regard to our work, the evolution of ($n1, n2, n3, m$) is towards an augmented dataset with better representation about minority class. To quantify this goal, we derive the relationship between ($n1, n2, n3, m$) and the generalization capacity of classification learning. To this end, we use the sampling result to train a classification model, here the ELM mentioned in Section 2. When model parameters θ is learned through $D \cup E_{syn}$, the generalization capacity of model f_θ can be represented by the localized generalization error model (LGEM) [34,35]

$$R_{SM}(q) = \int_{S_q} (f_\theta(x_u) - F(x_u))^2 p(x_u) dx_u, \tag{10}$$

in which $F(x_u)$ is the input–output rule, and $p(x_u)$ is the probability density function. For all training samples in D, S_q is the q-union set that can be defined as $S_q = S_q(x_1) \cup S_q(x_2) \cup \cdots S_q(x_N)$, in which $S_q(x_i)$ refers the q-neighborhood with respect to the ith training sample x_i, when given an input perturbation Δx_i, $S_q(x_i)$ can be defined as $\{x_u | x_{uj} = x_{ij} + \Delta x_{ij}, 0 < |\Delta x_{ij}| < q, j = 1, \ldots, n\}$ where the jth element in Δx_i is a small random value from the uniform distribution with zero mean and variance $\sigma^2_{\Delta x_{ij}}$. Each sample x_u in S_q is unseen, the localized generalization error of f_θ can be interpreted as the expectation of classification loss over these unseen samples. Therefore, in our work, we except to develop the relationship between ($n1, n2, n3, m$) and the localized

Please cite this article as: T. Zhang, Y. Li and X. Wang, Gaussian prior based adaptive synthetic sampling with non-linear sample space for imbalanced learning, Knowledge-Based Systems (2019) 105231, https://doi.org/10.1016/j.knosys.2019.105231.

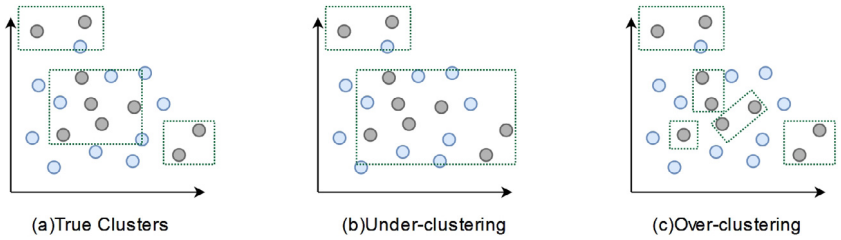

Fig. 6. The groups in minority class dataset. Each green frame denotes a subset in minority data. (a) shows the true but unknown division with respect to the minority class data. (b) shows a fail case with inadequate representation about minority sub-concepts. (c) shows the redundant groups in one minority sub-concept. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
12 imbalanced datasets from KEEL.

ID	Dataset	IR	Attributes	Train	Test
KEEL_1	ecoli1	3.36	7	268	68
KEEL_2	glass0	2.06	9	171	43
KEEL_3	glass1	1.82	9	171	43
KEEL_4	haberman	2.78	3	244	62
KEEL_5	new-thyroid1	5.14	5	172	43
KEEL_6	new-thyroid2	5.14	5	172	43
KEEL_7	segment0	6.02	19	1846	462
KEEL_8	vehicle0	3.25	18	676	170
KEEL_9	vehicle1	2.9	18	676	170
KEEL_10	vehicle2	2.88	18	676	170
KEEL_11	wisconsin	1.86	9	546	137
KEEL_12	yeast-2_vs_4	9.08	8	411	103

generalization error, then, by using this relation, $(n1, n2, n3, m)$ can be evolved towards the lower localized generalization error. However, $F(x_u)$ and $p(x_u)$ are unknown in practice, we thus use the error bound model of LGEM as the fitness function. According to Hoeffdings inequality, the upper bound of Eq. (10) can be derived as

$$R_{SM}(q) \leq [\sqrt{E_{S_q}((\Delta y)^2)} + \sqrt{R_{emp}} + A]^2 + \varepsilon. \tag{11}$$

Here, for a given training sample x_i, such that $\Delta y = f_\theta(x_u) - f_\theta(x_i)$, $\varepsilon = B\sqrt{\ln \eta/(-2N)}$, R_{emp} is the training error, A, B and η respectively are difference between the highest and lowest values in expected outputs, the possible maximum value of training loss and the confidence of the bound. $\frac{q^2}{3}\sum_{i=1}^{M}\sum_{j=1}^{M}\beta_i\beta_j\sum_{k=1}^{n}w_{ik}w_{jk}$ is the stochastic sensitivity measure $E_{S_q}((\Delta y)^2)$ (refer to appendix for the derivation). On the right of the Eq. (11), these items are determined by the model parameter θ. Furthermore, when given (w, b), β only depends on the training data. Thus, during the whole evolution process, (w, b) are fixed, when given a candidate solution, we use the corresponding balanced data to learn β through Eq. (5), the trained model is employed to derive the fitness degree, in this case, the upper bound of LGEM as described by Eq. (11).

By doing so, for a given task at hand, GMM can appropriately fit minority class sub-concepts by adaptively adjusting the size of Gaussian components. Then, based on GMM prior, the guided sampling and data clean can be performed with appropriate parameters ξ_{best} solved by Algorithm 2.

Algorithm 1: G-SMOTE

Input:
$n1$: sampling ratio
$n2$: the number of samples generated in each Ω
$n3$: the ratio of remaining samples
m: the number of Gaussians
D_{min}: the set of minority class cases
Output:
E_{syn}: the synthetic minority class samples.

Train Gaussian mixture model $\Theta = (\mu, \Sigma, \pi)$ with m components on D_{min} via EM, the main steps at the eth iteration are:
Expectation-Step:$Q(\Theta, \Theta^e) = \mathbb{E}[lnP(X, \delta|\Theta)|X, \Theta^e]$
Maximization-Step: $\Theta^{e+1} = argminQ(\Theta; \Theta^e)$ $//X \in \Re^{N\times n}$ is the feature matrix, δ is the missing variable.
Sample *kernels* from D_{min} based on Θ, and $|kernels| = |D_{min}| \times n1$
E_{syn}=Array[$n2 \times |kernels|$][n]
$index = 0$
for $i \leftarrow 1$ **to** $|kernels|$ **do**
  Compute the nearest neighbor $\hat{x}_i$ for *kernels*[i]
  for $j \leftarrow 1$ **to** $(n2)$ **do**
    R=distance between *kernels*[i] and $\hat{x}_i$
    Randomly choose a point x_{syn} from Ω, and $x_{syn} \notin E_{syn}$
    E_{syn}[*index*]=x_{syn}
    $index++$
  end
end
Get the top-$(n3 \times |E_{syn}|)$ instances with the high probability from E_{syn}

4. Performance evaluation and analysis

In this section, comprehensive experiments have been conducted on typical imbalanced datasets from the popular Knowledge Extraction based on Evolutionary Learning (KEEL) data repository [36]. Table 1 presents the details of these datasets. In regards to evolution metrics, typically, there are four outcomes for a set of instances, i.e. true positive (TP), false negative (FN), false positive (FP) and true negative (TN). These relations among the four outcomes are summarized in Fig. 8, in this work, the minority class is positive, and the majority class is negative. The accuracy representing the performance on all classes can be defined as

$$acc = \frac{TP + TN}{TP + TN + FP + FN}. \tag{12}$$

Besides, an additional measure, namely F-score, is used to evaluate the classification performance on positive class, and can be defined as

$$F - score = \frac{2\,\mathrm{Pr}\,ecision \times \mathrm{Re}call}{\mathrm{Pr}\,ecision + \mathrm{Re}call}, \tag{13}$$

in which the Precision and Recall can be calculated as

$$\begin{aligned} \mathrm{Pr}\,ecision &= \frac{TP}{TP + FP}; \\ \mathrm{Re}call &= \frac{TP}{TP + FN}. \end{aligned} \tag{14}$$

4.1. Comparison with different sampling methods

In this experimental part, the GA-SMOTE is compared with several popular imbalanced learning methods, namely ROS, RUS,

Algorithm 2: GA-SMOTE

Input:
G_e: Upper bound of generations
P_z: Size of population
M_u: Mutation factor
C_r: Crossover probability
D_{min}: Minority class cases set
$range = (\xi_{min}, \xi_{max})$: Range between lower bound ξ_{min} and upper bound ξ_{max}
Output: ξ_{best}

Step 1) Initialization:
Generate an initial population $\{\xi_1^0, ..., \xi_{P_z}^0\}$.
$\xi_i^0 = \xi_{min} + rand(0, 1) * (\xi_{max} - \xi_{min})$, $i = 1, 2, ..., P_z$.
Evaluate each initial candidate solution ξ_i^0 via the fitness functions $fitness(\xi_i^g)$:
$(n1, n2, n3, m) = \xi_i^g$, $g \in \{0, 1, ..., G_e\}$
E_{syn}= G-SMOTE($n1, n2, n3, m, D_{min}$)
$D_{aug} = D \cup E_{syn}$
Train ELM on D_{aug} and return the upper bound of LGEM with respect to ξ_i^g.
Step 2) Evolution:
for $g = 1, \cdots, G_e$ **do**
  for $i = 1, \cdots, P_z$ **do**
    Step 2.1) Mutation:
    Randomly choice two indexes $r1$ and $r2$ from $\{1, \cdots, P_z\}$;
    $h_i^g = \xi_i^{g-1} + M_u \cdot (\xi_{r1}^{g-1} - \xi_{r2}^{g-1})$;
    Step 2.2) Crossover:
    for $j = 1, \cdots, 4$ **do**
      $v_i^g(j) = \begin{cases} h_i^g(j), & \text{if } rand(0, 1) \le C_r \\ \xi_i^{g-1}(j), & \text{otherwise} \end{cases}$
    end
    Step 2.3) Selection
    if $fitness(v_i^g) < fitness(\xi_i^{g-1})$ **then**
      $\xi_i^g = v_i^g$
    else
      $\xi_i^g = \xi_i^{g-1}$
    end
  end
end
$\xi_{best} = argmin(fitness(\xi_i^{G_e}))$, $i = 1, ..., P_z$

SMOTE, SMOTE-ENN and SMOTE-TL. The classifier is the ELM as used in Algorithm 2. During the iteration of DE, the hidden weights and thresholds of ELM are fixed after stochastic initialization. Each imbalanced dataset is re-balanced by all methods mentioned above, and the balanced datasets are used to train ELMs with the same (w, b), then the results on test data are reported in Table 2. According to these consequences, one can note that all the performance on different imbalanced data can be significantly improved by the proposed method (average increment on accuracy and F-score are 3.33% and 26.84%, respectively), these boosted results indicate that the GA-SMOTE can provide favorable accuracies across overall categories, the reason is that the bias on majority class is adjusted by the re-balanced dataset with reasonable synthetic samples. Meanwhile, compared with other imbalanced techniques, our method gets higher accuracy and F-score in most cases. These improvement can demonstrate that GA-SMOTE has better robustness, which benefits from GMM prior and reasonable sampling space which greatly avoid interference of outliers, and enhance the controllability in sampling process and sample synthesis. Besides, compared with other data clean techniques (e.g., SMOTE-ENN and SMOTE-TL), GA-SMOTE has better performance, because the data clean in the proposed method removes the unsatisfactory synthetic samples without losing the majority class information, and the proposed method can adaptively work for better generalization.

4.2. Extended experiment on different classification algorithms

To further demonstrate the effectiveness of GA-SMOTE, these balanced datasets mentioned in previous section are used to train different classifiers, i.e. support vector machine (SVM), k-nearest neighbors (KNN), naive Bayesian (NB), random forest (RF) and classification and regression tree (CART). Tables 3–7 report the results on test datasets.

By these results, we can observe that the balanced datasets processed by our method also improve the performance of various classification algorithms, particularly for the sensitive approach to skewed category distribution, such as SVM, its results tend to behave great difference between high accuracy and low F-score (at least 5.3% difference), which indicates high FN rate on minority class in tandem with high FP rate on majority class. This mainly is due to the fact that majority class dominates the learning algorithm. In contrast, the compensation yielded by our method enhances the representation of few-shot instances (average improvement of F-score is 19.58% for all classifiers). This confirms that the generalization ability is dependent on the data the classifiers were trained on. Compared with other imbalanced learnings, the proposed method achieves better results in general, especially the predication accuracy of few-shot samples is boosted, while the FP rate of majority case is weakened. These results indicate that the synthetic samples are more similar to existing minority class samples due to the guidance of GMM prior, and the extended minority data take more effective attention of classification learning.

5. Conclusion

In this paper, we propose the GA-SMOTE which is a novel over-sampling mechanic for imbalanced learning. GA-SMOTE improves the over-sampling robustness from three aspects. On one hand, instead the synthetic instances are generated in high dimensional feature space rather than a simple linear space. On the other hand, the GMM is employed to distinguish the outliers from minority class instances and filter out the synthetic instances with low agreement to the minority class concept. Last and more importantly, an adaptive optimization method is proposed to optimize these parameters in sampling process. By doing so, synthetic samples can be created in an effectiveness and efficiency way. Comprehensive experiments prove that this proposed framework provides a more robust way to generate minority class instances, and boost the performance of different classification algorithms on imbalanced issues.

Appendix

Given a training sample x, the derivation about stochastic sensitivity measure [37] can be described in what follows.

According to the Taylors series expansion: $\frac{1}{1+x} = \sum_{t=0}^{\infty}(-1)^t x^t$, $-1 < x < 1$, the first item in Eq. (1) can be rewritten as

$$f_\theta(x) = \sum_{j=1}^{M} \beta_j \sum_{t=0}^{\infty} (-1)^t (\exp(-(w_j \times x + b_j)))^t, \quad 0 < \exp(-(w_j \times x + b_j)) < 1. \tag{15}$$

Please cite this article as: T. Zhang, Y. Li and X. Wang, Gaussian prior based adaptive synthetic sampling with non-linear sample space for imbalanced learning, Knowledge-Based Systems (2019) 105231, https://doi.org/10.1016/j.knosys.2019.105231.

Table 2
Accuracy/*F*-score of ELM.

ID	ORIGIN	ROS	RUS	SMOTE	SMOTE-ENN	SMOTE-TL	GA-SMOTE
KEEL_1	0.8529/0.6667	0.8676/0.7429	0.8824/0.75	0.8529/0.7059	0.8529/0.7222	0.8676/0.7429	0.8971/0.7742
KEEL_2	0.8701/0.2105	0.7554/0.5066	0.7576/0.5214	0.7554/0.4978	0.7446/0.4825	0.7468/0.4891	0.8939/0.5950
KEEL_3	0.8588/0.6842	0.7941/0.6602	0.7824/0.6542	0.7882/0.6538	0.8/0.6792	0.7941/0.6729	0.8824/0.7561
KEEL_4	0.7765/0.4722	0.7059/0.5763	0.7235/0.5913	0.6941/0.5593	0.7/0.5785	0.6588/0.5538	0.7882/0.6327
KEEL_5	0.9416/0.92	0.9416/0.9184	0.9562/0.94	0.9416/0.9216	0.9197/0.8911	0.927/0.9	0.9489/0.9293
KEEL_6	0.9223/0.5	0.932/0.72	0.9029/0.6429	0.9029/0.6429	0.9029/0.6429	0.9029/0.6429	0.9515/0.7368
KEEL_7	0.7209/0.4	0.6744/0.65	0.6977/0.6667	0.6744/0.65	0.6512/0.6341	0.5581/0.5957	0.7674/0.6667
KEEL_8	0.5116/0.087	0.5116/0.4878	0.4884/0.4762	0.4884/0.4762	0.5116/0.5116	0.4884/0.5	0.5349/0.5238
KEEL_9	0.7419/0.2	0.6774/0.4737	0.5806/0.35	0.4032/0.3509	0.3871/0.3448	0.4032/0.3729	0.7742/0.5882
KEEL_10	0.8837/0.4444	0.907/0.7778	0.8605/0.7	0.907/0.7778	0.907/0.7778	0.907/0.7778	0.9767/0.9333
KEEL_11	0.907/0.6	0.6744/0.4615	0.7209/0.5	0.6512/0.4444	0.6512/0.4444	0.6512/0.4444	0.9535/0.8571
KEEL_12	0.7118/0.1695	0.7176/0.5789	0.7059/0.5763	0.7118/0.5664	0.6941/0.5593	0.6647/0.5366	0.7294/0.5818

Table 3
Accuracy/*F*-score of KNN.

ID	ORIGIN	ROS	RUS	SMOTE	SMOTE-ENN	SMOTE-TL	GA-SMOTE
KEEL_1	1.0/0.9231	0.9535/0.875	0.907/0.75	0.9767/0.9333	0.9535/0.8571	0.9302/0.8	1.0/0.9767
KEEL_2	0.8434/0.8293	0.9059/0.8298	0.8765/0.7921	0.8882/0.8	0.8765/0.7879	0.8824/0.7959	0.9235/0.9176
KEEL_3	0.9524/0.8421	0.9417/0.7692	0.9515/0.8	0.9126/0.6897	0.8932/0.6452	0.8835/0.625	0.9903/0.9709
KEEL_4	0.4348/0.3377	0.6647/0.5043	0.6941/0.5806	0.7/0.5565	0.6882/0.5691	0.6824/0.5781	0.6941/0.7
KEEL_5	1.0/0.9231	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0	1.0/0.9767
KEEL_6	0.9701/0.9848	0.9827/0.9429	0.8506/0.6567	0.9827/0.9429	0.9848/0.9496	0.9848/0.9496	0.9913/0.9957
KEEL_7	0.5/0.5	0.6744/0.5625	0.5581/0.4242	0.6279/0.5	0.6512/0.5455	0.6744/0.6316	0.6744/0.6744
KEEL_8	0.6897/0.6897	0.7674/0.7222	0.7442/0.6667	0.7907/0.7273	0.7209/0.6471	0.6977/0.6667	0.7907/0.7907
KEEL_9	0.75/0.7381	0.8765/0.7961	0.8235/0.7115	0.8765/0.7921	0.8588/0.7692	0.8765/0.8	0.8706/0.8706
KEEL_10	0.5/0.32	0.6452/0.45	0.629/0.4651	0.4677/0.4	0.5161/0.4828	0.4839/0.4667	0.6774/0.7258
KEEL_11	0.9895/0.9895	0.9854/0.9792	0.9854/0.9792	0.9708/0.9592	0.9781/0.9691	0.9781/0.9691	0.9927/0.9927
KEEL_12	0.7742/0.7586	0.8676/0.7429	0.8676/0.7273	0.8824/0.75	0.8529/0.7059	0.8676/0.7429	0.8971/0.8971

Table 4
Accuracy/*F*-score of SVM.

ID	ORIGIN	ROS	RUS	SMOTE	SMOTE-ENN	SMOTE-TL	GA-SMOTE
KEEL_1	0.4444/0.25	0.8837/0.4444	0.7674/0.5833	0.8837/0.4444	0.8837/0.4444	0.8837/0.4444	0.8837/0.8605
KEEL_2	0.7647/0.0	0.7647/0.0	0.7765/0.0952	0.7647/0.0	0.7647/0.0	0.7706/0.0488	0.7647/0.7647
KEEL_3	0.4286/0.0	0.9612/0.8333	0.9709/0.8571	0.9515/0.8	0.9515/0.8	0.9515/0.8	0.9223/0.8932
KEEL_4	0.7412/0.0	0.7412/0.0	0.3118/0.4236	0.7412/0.0	0.2765/0.4171	0.2706/0.4151	0.7412/0.7412
KEEL_5	0.8333/0.4444	0.907/0.6	0.6744/0.5	0.9302/0.7273	0.9535/0.8333	0.9535/0.8333	0.9535/0.8837
KEEL_6	0.68/0.5957	0.9307/0.68	0.9589/0.8319	0.9524/0.8	0.9502/0.789	0.9502/0.7928	0.9307/0.9177
KEEL_7	0.6047/0.5517	0.6279/0.619	0.6279/0.619	0.6279/0.619	0.6279/0.619	0.4884/0.56	0.6047/0.6977
KEEL_8	0.8125/0.3158	0.6744/0.6667	0.6279/0.6364	0.6744/0.6667	0.7442/0.7179	0.6279/0.6364	0.8605/0.6977
KEEL_9	0.7412/0.0	0.7412/0.0	0.7824/0.2745	0.7588/0.1277	0.7588/0.1277	0.7529/0.087	0.7412/0.7412
KEEL_10	0.3125/0.0	0.6452/0.0833	0.5484/0.4815	0.5806/0.2778	0.5323/0.5085	0.5323/0.5085	0.6452/0.7258
KEEL_11	0.94/0.9495	0.9635/0.9495	0.9562/0.94	0.9562/0.94	0.9562/0.94	0.9562/0.94	0.9562/0.9635
KEEL_12	0.8/0.6667	0.8971/0.8	0.8971/0.8	0.8971/0.8	0.8824/0.7778	0.8971/0.8	0.8971/0.8676

Table 5
Accuracy/*F*-score of CART.

ID	ORIGIN	ROS	RUS	SMOTE	SMOTE-ENN	SMOTE-TL	GA-SMOTE
KEEL_1	0.9333/0.8	0.9767/0.9231	0.9535/0.8571	1.0/1.0	1.0/1.0	1.0/1.0	0.9767/0.9302
KEEL_2	0.8571/0.8736	0.9294/0.85	0.9235/0.8539	0.8765/0.7273	0.9176/0.8372	0.9/0.809	0.9294/0.9353
KEEL_3	0.8696/0.8	0.9709/0.8571	0.9126/0.6897	0.9417/0.75	0.9417/0.7692	0.932/0.7407	0.9709/0.9612
KEEL_4	0.5055/0.4889	0.7412/0.4884	0.7588/0.6496	0.7706/0.6139	0.7353/0.5946	0.7412/0.6071	0.7353/0.7294
KEEL_5	1.0/1.0	1.0/1.0	0.9535/0.875	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0
KEEL_6	0.9848/0.9774	0.9957/0.9851	0.9784/0.9296	0.9935/0.9778	0.9913/0.9706	0.987/0.9559	0.9957/0.9935
KEEL_7	0.5806/0.5161	0.6744/0.5333	0.6977/0.6486	0.6279/0.5	0.6977/0.6286	0.6512/0.5946	0.6977/0.6512
KEEL_8	0.6429/0.75	0.907/0.8571	0.8372/0.7742	0.7907/0.7097	0.907/0.8667	0.8605/0.8125	0.7674/0.814
KEEL_9	0.9176/0.8941	0.9588/0.9157	0.9353/0.8791	0.9765/0.9535	0.9529/0.9048	0.9529/0.9091	0.9588/0.9471
KEEL_10	0.5641/0.3226	0.6935/0.4242	0.7097/0.5714	0.6452/0.3529	0.5968/0.5283	0.4516/0.3462	0.7258/0.6613
KEEL_11	0.9574/0.8791	0.9708/0.9565	0.9708/0.9574	0.9562/0.9333	0.9416/0.9184	0.9708/0.9574	0.9708/0.9197
KEEL_12	0.6667/0.7097	0.8529/0.6875	0.8382/0.6857	0.9118/0.8125	0.8676/0.7273	0.8676/0.7273	0.8529/0.8676

Please cite this article as: T. Zhang, Y. Li and X. Wang, Gaussian prior based adaptive synthetic sampling with non-linear sample space for imbalanced learning, Knowledge-Based Systems (2019) 105231, https://doi.org/10.1016/j.knosys.2019.105231.

Table 6
Accuracy/*F*-score of NB.

ID	ORIGIN	ROS	RUS	SMOTE	SMOTE-ENN	SMOTE-TL	GA-SMOTE
KEEL_1	0.875/0.9333	0.9767/0.9333	0.9767/0.9333	0.9767/0.9333	0.9767/0.9333	0.9767/0.9333	0.9535/0.9767
KEEL_2	0.5197/0.5323	0.6471/0.5522	0.6471/0.5385	0.6353/0.5303	0.6412/0.5344	0.6353/0.5303	0.6412/0.6588
KEEL_3	0.1538/0.1942	0.165/0.1887	0.9223/0.7143	0.1942/0.1942	0.233/0.202	0.1942/0.1942	0.8932/0.1942
KEEL_4	0.5217/0.531	0.6765/0.5455	0.6353/0.5231	0.6529/0.5354	0.6412/0.5271	0.6353/0.5373	0.6765/0.6882
KEEL_5	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0
KEEL_6	0.7052/0.6337	0.8268/0.6154	0.8355/0.6275	0.8312/0.6214	0.8312/0.6214	0.8312/0.6214	0.8896/0.8398
KEEL_7	0.55/0.6383	0.5116/0.5532	0.5581/0.6122	0.5814/0.625	0.5349/0.6	0.5116/0.5532	0.5814/0.6047
KEEL_8	0.6364/0.6364	0.6047/0.6222	0.6279/0.6364	0.6279/0.6364	0.6279/0.6364	0.6047/0.6222	0.6279/0.6279
KEEL_9	0.5546/0.561	0.8059/0.6796	0.7824/0.6337	0.7706/0.6355	0.8059/0.6733	0.7706/0.6422	0.6882/0.7882
KEEL_10	0.5517/0.4167	0.8226/0.5926	0.8065/0.5714	0.7258/0.5405	0.7258/0.5405	0.7258/0.5641	0.7903/0.7742
KEEL_11	0.9592/0.9592	0.9708/0.9592	0.9708/0.9592	0.9635/0.9495	0.9635/0.9495	0.9635/0.9495	0.9708/0.9708
KEEL_12	0.7429/0.4167	0.4118/0.4286	0.3676/0.411	0.4559/0.4478	0.4559/0.4478	0.4559/0.4478	0.8676/0.3824

Table 7
Accuracy/*F*-score of RF.

ID	ORIGIN	ROS	RUS	SMOTE	SMOTE-ENN	SMOTE-TL	GA-SMOTE
KEEL_1	1.0/1.0	0.9767/0.9231	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0
KEEL_2	0.8571/0.875	0.9588/0.9136	0.9294/0.8696	0.9412/0.878	0.9588/0.9176	0.9294/0.8605	0.9353/0.9412
KEEL_3	0.8182/0.6667	0.9515/0.7619	0.932/0.72	0.9612/0.8333	0.9515/0.8	0.9515/0.8	0.9612/0.9417
KEEL_4	0.5063/0.4571	0.7765/0.5778	0.8235/0.7059	0.7941/0.6237	0.7882/0.6842	0.7706/0.6667	0.7706/0.7765
KEEL_5	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0	1.0/1.0
KEEL_6	1.0/0.9924	1.0/0.9978	0.9935/0.9778	0.9978/0.9925	0.9978/0.9925	0.9957/0.9851	1.0/1.0
KEEL_7	0.5333/0.6667	0.7442/0.6452	0.7209/0.6471	0.7907/0.7097	0.6977/0.5806	0.6512/0.5946	0.6744/0.7674
KEEL_8	0.7857/0.7407	0.8605/0.7857	0.9302/0.8966	0.8372/0.7407	0.8605/0.8	0.8605/0.8125	0.8605/0.8372
KEEL_9	0.9655/0.9268	0.9765/0.9535	0.9765/0.9545	0.9941/0.9885	0.9824/0.9655	0.9882/0.9773	0.9824/0.9647
KEEL_10	0.3571/0.0833	0.6935/0.3448	0.7097/0.55	0.5645/0.3415	0.5968/0.5098	0.6129/0.5385	0.7097/0.6452
KEEL_11	0.9677/0.9574	0.9781/0.9684	0.9635/0.9474	0.9635/0.9474	0.9708/0.9583	0.9708/0.9574	0.9781/0.9708
KEEL_12	0.6875/0.6667	0.8676/0.7273	0.8824/0.7647	0.8676/0.7273	0.8824/0.7647	0.8676/0.7273	0.8529/0.8529

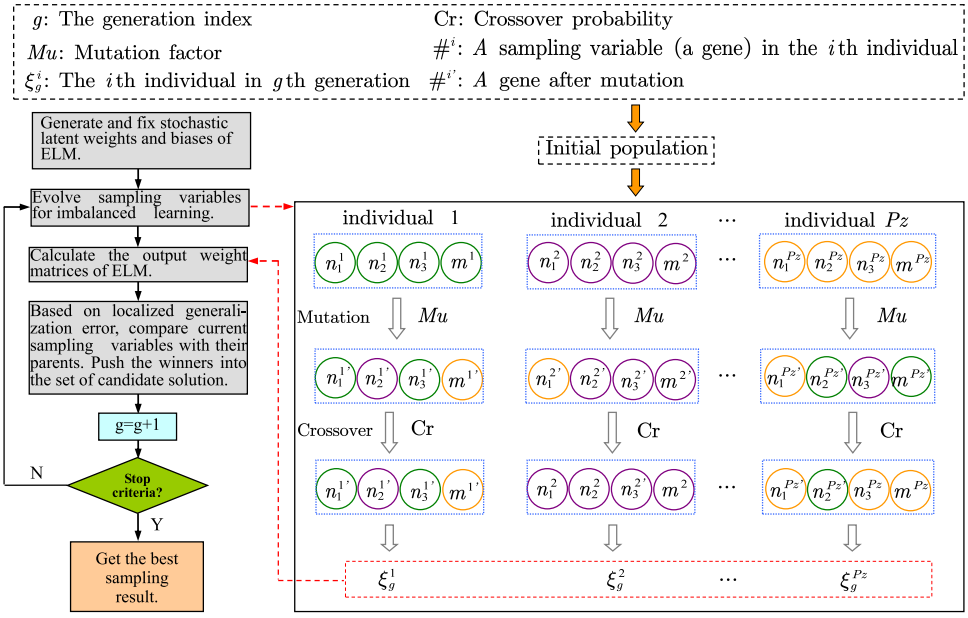

Fig. 7. Evolutionary procedure of imbalanced learning. The right panel shows the mutation and crossover in differential evolution. P_z is the size of population. The red dashed framework is a candidate solution set. The left part shows the iterative optimization of differential evolution. The algorithm has two stop criteria, one is the upper bound of iteration, the other is convergent result. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The terms with order greater than 1 are ignored, then we have $f_\theta(x) \approx \sum_{j=1}^{M} \beta_j(1 - \exp(-(w_j \times x) + b_j))$.

Let $S_j = \sum_{i=1}^{n}(w_{ji}x_i + b_j)$ and $S_j^* = \sum_{i=1}^{n}(w_{ji}(x_i + \Delta x_i) + b_j)$, then $E_{S_q}((\Delta y)^2)$

$$= E_{S_q}((\sum_{j=1}^{M} \beta_j(1 - exp(-S_j^*)) - \sum_{j=1}^{M} \beta_j(1 - exp(-S_j)))^2)$$

$$= E_{S_q}((\sum_{j=1}^{M} \beta_j(exp(-S_j) - exp(-S_j^*)))^2)$$

Let $V_j = exp(-S_j) - exp(-S_j^*)$, then $E_{S_q}((\Delta y)^2) = \sum_{j=1}^{M}\sum_{i=1}^{M} \beta_i\beta_j E_{S_q}(V_iV_j)$, we have $E_{S_q}(V_iV_j)$

$$= E_{S_q}(exp(-S_i - S_j)) - E_{S_q}(exp(-S_i - S_j^*))$$

$$-E_{S_q}(exp(-S_i^* - S_j)) + E_{S_q}(exp(-S_i^* - S_j^*)).$$

Please cite this article as: T. Zhang, Y. Li and X. Wang, Gaussian prior based adaptive synthetic sampling with non-linear sample space for imbalanced learning, Knowledge-Based Systems (2019) 105231, https://doi.org/10.1016/j.knosys.2019.105231.

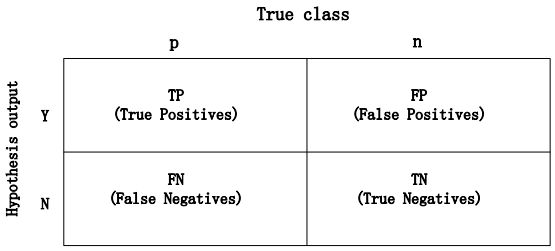

Fig. 8. Confusion matrix about four outcomes.

Based on the central limit theorem, $\exp(S_j)$ and $\exp(S_j^*)$ have a log-normal distribution, thus $E_{S_q}(exp(-S_i^* - S_j^*))$

$$= exp(\frac{Var(S_i^* + S_j^*)}{2} - E(S_i^* + S_j^*))$$
$$\approx 1 + \frac{Var(S_i^* + S_j^*)}{2} - E(S_i^* + S_j^*),$$

in which, the first item on the right hand approximates to one. Then,

$$E_{S_q}(V_iV_j) = \frac{1}{2}(Var(S_i^* + S_j^*) + Var(S_i + S_j)$$
$$-Var(S_i^* + S_j) - Var(S_i + S_j^*)).$$

Because $Var(S_i^* + S_j^*) =$

$$Var(\sum_{k=1}^{n}(w_{ik}(x_k + \Delta x_k) + b_i)$$
$$+ \sum_{k=1}^{n}(w_{jk}(x_k + \Delta x_k) + b_j))$$
$$= \sum_{k=1}^{n}(w_{ik} + w_{jk})^2 Var(x_k) + \frac{q^2}{3}\sum_{k=1}^{n}(w_{ik} + w_{jk})^2$$

Finally, $E_{S_q}((\Delta y)^2) = \frac{q^2}{3}\sum_{i=1}^{M}\sum_{j=1}^{M}\beta_i\beta_j\sum_{k=1}^{n}w_{ik}w_{jk}$.

References

[1] Y. Sun, A.K. Wong, M.S. Kamel, Classification of imbalanced data: a review, Int. J. Pattern Recognit. Artif. Intell. 23 (04) (2009) 687–719.
[2] H. He, E. Garcia, Learning from imbalanced data, IEEE Trans. Knowl. Data Eng. 21 (9) (2009) 1263–1284.
[3] P. Hajek, R. Henriques, Mining corporate annual reports for intelligent detection of financial statement fraud – A comparative study of machine learning methods, Knowl.-Based Syst. 128 (15) (2017) 139–152.
[4] X. Gong, W. Qiao, Imbalance fault detection of direct-drive wind turbines using generator current signals, IEEE Trans. Energy Convers. 27 (2) (2012) 468–476.
[5] N. Japkowicz, S. Stephen, The class imbalance problem: A systematic study, Intel. Data Anal. 6 (5) (2002) 429–449.
[6] Learning from imbalanced data sets, in: N. Japkowicz (Ed.), Proc. Am. Assoc. for Artificial Intelligence (AAAI) Workshop, 2000, Technical Report WS-00-05.
[7] Workshop learning from imbalanced data sets II, in: N.V. Chawla, N. Japkowicz, A. Kolcz (Eds.), Proceedings of the International Conference on Machine Learning, 2003.
[8] N. Chawla, N. Japkowicz, A. Kolcz, Editorial: Special issue on learning from imbalanced data sets, Intel. Data Anal. 6 (1) (2004) 1–6.
[9] N. Japkowicz, The class imbalance problem: significance and strategies, in: Proceedings of the International Conference on Artificial Intelligence, 2000, pp. 111–117.
[10] M. Kubat, S. Matwin, Addressing the curse of imbalanced training sets: one sided selection, in: Proceedings of the International Conference on Machine Learning, 1997, pp. 179–186.
[11] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, J. Artificial Intelligence Res. 16 (1) (2002) 321–357.
[12] H. Han, W. Wang, B. Mao, Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning, in: Proceeding of International Conference on Intelligent Computing, 2005, pp. 878–887.
[13] S. Ertekin, J. Huang, L. Bottou, L. Giles, Learning on the border: active learning in imbalanced data classification, in: Proc. ACM Conf. Information and Knowledge Management, 2007, pp. 127–136.
[14] S. Ertekin, J. Hunag, C. Giles, Active learning for class imbalance problem, in: Proc. Int'L SIGIR Conf. Research and Development in Information Retrieval, 2007, pp. 823–824.
[15] J. Zhai, S. Zhang, C. Wang, The classification of imbalanced large data sets based on MapReduce and ensemble of ELM classifiers, Int. J. Mach. Learn. Cybern. 8 (3) (2017) 1009–1017.
[16] J. Zhai, S. Zhang, M. Zhang, X. Liu, Fuzzy integral-based ELM ensemble for imbalanced big data classification, Soft Comput. 22 (11) (2018) 3519–3531.
[17] A. Solberg, R. Solberg, A large-scale evaluation of features for automatic detection of oil spills in ERS SAR images, in: Proceeding of International Conference on Geoscience and Remote Sensing Symposium, 1996, pp. 1484–1486.
[18] X.L. Yang, D. Lo, X. Xia, Q. Huang, J. Sun, High-impact bug report identification with imbalanced learning strategies, J. Comput. Sci. Tech. 32 (1) (2017) 181–198.
[19] P. Su, X.Z. Wang, Y. Li, Modeling chess strategy by classifier based on imbalance learning and application in computer chinese chess, J. Comput. Res. Dev. 48 (5) (2011) 841–847.
[20] N. García-Pedrajas, J. Pérez-Rodríguez, Class imbalance methods for translation initiation site recognition in DNA sequences, knowl.-Based Syst. 25 (1) (2012) 22–34.
[21] H. Xu, C. Zhang, G. Hong, J. Zhou, Gated recurrent units based neural network for tool condition monitoring, in: Proceeding of International Joint Conference on Neural Networks, 2018.
[22] G. Huang, Q. Zhu, C.K. Siew, Extreme learning machine: Theory and applications, Neurocomputing 70 (1) (2006) 489–501.
[23] X.Z. Wang, T.L. Zhang, R. Wang, Non-iterative deep learning: Incorporating restricted Boltzmann machine into multilayer random weight neural networks, IEEE Trans. Syst. Man Cybern. Syst. 99 (2017) 1–10.
[24] L. X.Y., J. Wu, Z. Zhou, exploratory under sampling for class imbalance learning, in: Proc. Int'l Conf. Data Mining, 2006, pp. 965–969.
[25] G. Liu, Y. Yang, B. Li, Fuzzy rule-based oversampling technique for imbalanced and incomplete data learning, Knowl.-Based Syst. 158 (15) (2018) 154–174.
[26] W. Ng, J. Hu, D. Yeung, Diversified sensitivity-based undersampling for imbalance classification problems, IEEE Trans. Cybern. 45 (11) (2015) 2402–2412.
[27] M. Vijay, X. Nguyen, Identification of the dynamic operating envelope of HCCI engines using class imbalance learning, IEEE Trans. Neural Netw. Learn. Syst. 26 (1) (2015) 98–112.
[28] S. Lin, Integrated artificial intelligence-based resizing strategy and multiple criteria decision making technique to form a management decision in an imbalanced environment, Int. J. Mach. Learn. Cybern. 8 (6) (2017) 1981–1992.
[29] A. Estabrooks, T. Jo, N. Japkowicz, A multiple resampling method for learning from imbalanced data sets, Comput. Intell. 20 (2004) 18–36.
[30] I. Tomek, Two modifications of CNN, IEEE Trans. Syst. Man Cybern. 6 (11) (1976) 769–772.
[31] G.P. Batista, R. Prati, M. Monard, A study of the behavior of several methods for balancing machine learning training data, ACM SIGKDD Explor. Newsl. 6 (1) (2004) 20–29.
[32] J. Laurikkala, Improving identification of difficult small classes by balancing class distribution, in: Proc. Conf. AI in Medicine in Europe: Artificial Intelligence Medicine, 2001, pp. 63–66.
[33] C. Dong, W.W.Y. Ng, X.Z. Wang, P.P.K. Chan, D.S. Yeung, An improved differential evolution and its application to determining feature weights in similarity-based clustering, Neurocomputing 146 (146) (2014) 95–103.
[34] D.S. Yeung, W.W.Y. Ng, D. Wang, E.C.C. Tsang, X.Z. Wang, Localized generalization error model and its application to architecture selection for radial basis function neural network, IEEE Trans. Neural Netw. 18 (5) (2007) 1294–1305.
[35] W.W.Y. Ng, D.S. Yeung, M. Firth, E.C.C. Tsang, X.Z. Wang, Feature selection using localized generalization error for supervised classification problems using RBFNN, Pattern Recognit. 41 (12) (2008) 3706–3719.
[36] KEEL, Available online, https://sci2s.ugr.es/keel/imbalanced.php (3/21/2019 available).
[37] W.W.Y. Ng, D.S. Yeung, D.F. Wang, E.C.C. Tsang, X.Z. Wang, Localized generalization error of Gaussian-based classifiers and visualization of decision boundaries, Soft Comput. 11 (4) (2007) 357–381.

Please cite this article as: T. Zhang, Y. Li and X. Wang, Gaussian prior based adaptive synthetic sampling with non-linear sample space for imbalanced learning, Knowledge-Based Systems (2019) 105231, https://doi.org/10.1016/j.knosys.2019.105231.