sensors

MDPI

Article

Evolutionary Mahalanobis Distance-Based Oversampling for Multi-Class Imbalanced Data Classification

Leehter Yao * and Tung-Bin Lin

Department of Electrical Engineering, National Taipei University of Technology, Taipei 10618, Taiwan; dark5027@gmail.com

* Correspondence: ltyao@ntut.edu.tw

Abstract: The number of sensing data are often imbalanced across data classes, for which oversampling on the minority class is an effective remedy. In this paper, an effective oversampling method called evolutionary Mahalanobis distance oversampling (EMDO) is proposed for multi-class imbalanced data classification. EMDO utilizes a set of ellipsoids to approximate the decision regions of the minority class. Furthermore, multi-objective particle swarm optimization (MOPSO) is integrated with the Gustafson–Kessel algorithm in EMDO to learn the size, center, and orientation of every ellipsoid. Synthetic minority samples are generated based on Mahalanobis distance within every ellipsoid. The number of synthetic minority samples generated by EMDO in every ellipsoid is determined based on the density of minority samples in every ellipsoid. The results of computer simulations conducted herein indicate that EMDO outperforms most of the widely used oversampling schemes.

Keywords: oversampling; mahalanobis distance; MOPSO; classification; minority class; ellipsoid

Citation: Yao, L.; Lin, T.-B. Evolutionary Mahalanobis Distance-Based Oversampling for Multi-Class Imbalanced Data Classification. *Sensors* **2021**, *21*, 6616. https://doi.org/10.3390/s21196616

Academic Editor: Andrzej Stateczny

Received: 23 July 2021
Accepted: 29 September 2021
Published: 4 October 2021

1. Introduction

With advancements in sensor technology and the Internet of things (IoT), vast quantities of sensing data have been collected and analyzed for different applications. Cost-effective sensors are widely used in our everyday lives to collect various types of data for further online or offline analyses and applications. The classification of real-world sensing data is a highly important research topic in the field of data mining and machine learning. However, the data sets collected using sensors or other sensing techniques usually have a skewed class distribution because the number of data points vary greatly between classes. Such data are called imbalanced data. The data utilized in applications, such as anomaly detection in high-speed trains [1–3], fault diagnosis of motors [4–6], fault detection and diagnosis in manufacturing processes [7–9], and medical diagnosis [10–12], are usually imbalanced. In imbalanced data sets, at least one class of data has significantly more data points compared with other classes. Learning on imbalanced data results in poor performance, and this problem has thus attracted considerable research attention in recent years. It is mainly because the performance of many conventional learning algorithms is degraded on the skewed class distribution of imbalanced data sets [13].

Balanced class distribution [14] or equal weighting of classification errors for every class [13] is generally assumed in most conventional machine learning algorithms. For instance, 95% and 5% of imbalanced data sets can comprise majority and minority class samples, respectively. With the equal weighting of the classification errors, traditional classification approaches tend to overlook several or most of the minority class samples in the attempt to minimize the overall classification error. Consequently, although the overall classification error rate is low, the classification error rate for the minority class is high. Minority class samples are important in classification in certain applications, such as medical diagnosis, anomaly detection, and fault detection and diagnosis. Majority class samples usually represent normal conditions, and minority class samples represent abnormal conditions, which can be key in such applications. The learning approaches for

imbalanced data are designed to increase learning accuracy with respect to minority classes without trading off learning accuracy with respect to majority classes.

The learning approaches for imbalanced data can generally be categorized into three types: cost-sensitive learning, data-level learning, and ensemble learning, which are comprehensively reviewed in [15,16]. Cost-sensitive learning assigns higher misclassification costs to minority class samples than to majority class samples. Studies have proposed various learning approaches that adjust the misclassification cost using kernel functions; these approaches involve the radial basis function [17], matrix-based kernel regression [18], support vector machine [19,20], and deep learning [21,22].

Data-level learning approaches essentially rebalance the skewed data distribution of different classes by removing several majority class samples or by adding new minority class samples, and they can generally be divided into undersampling and oversampling learning approaches. The main advantage of data-level learning approaches is that they are independent of classifiers. They can be considered a type of data preprocessing approach. Therefore, data-level learning approaches can be easily integrated with other imbalanced learning approaches. Undersampling involves removing the majority class samples to ensure that the learning results are not overly biased toward the majority class [23–26]. Undersampling reduces both the number of samples in and the computational cost of machine learning. However, it tends to reduce the model's capability to recognize majority classes. By contrast, oversampling involves increasing the number of minority class samples by resampling them or by generating synthetic samples. However, resampling by simply replicating the minority class samples does not improve learning of the decision region of minority classes. The synthetic minority oversampling technique (SMOTE) proposed in [27] selects several samples from a minority class. Searching in the vicinity of the selected samples, it identifies other samples of the minority class to generate new synthetic samples linearly between the two points. SMOTE is the most widely used oversampling technique because of its computational inexpensiveness. However, SMOTE is prone to overgeneralization because it synthesizes new samples through the random selection of minority class samples. Various adaptive sampling methods based on SMOTE have been proposed to overcome its limitation. The adaptive synthetic sampling approach for imbalanced learning algorithm (ADASYN) [28], SMOTEBoost [29], and Borderline SMOTE [30] are effective modified versions of SMOTE. In contrast to SMOTE, other algorithms, such as those in [31–33], have been proposed; these algorithms generate synthetic samples by learning the structure underlying the minority samples.

Ensemble learning for incomplete data integrates traditional machine learning approaches, such as boosting [34], bagging [35], and stacking [36], with other cost-sensitive or data resampling imbalanced learning approaches. In [37], SMOTE was integrated with Adaboost [38] to increase the number of minority samples and to assign higher weights to misclassified minority samples. A similar integration of Adaboost with a novel synthetic sampling method was proposed in [39]. The performance of boosting, bagging, and other hybrid techniques applied to imbalanced data has been compared in [40] and [41].

In the methods proposed in [31–33], synthetic samples are generated based not on individual minority samples as proposed in SMOTE [30] but on the underlying structure of the minority samples. Recently, a similar oversampling approach called Mahalanobis distance-based oversampling (MDO) was proposed in [42]. MDO generates synthetic samples based on the structure of the principal component space of minority samples. The synthetic samples generated by MDO have the same Mahalanobis distance as that of the considered minority sample. Because the class mean of the synthetic samples generated by MDO is the same as that of the minority class samples, the covariance structure of the minority class samples is preserved. In [43], a scheme called adaptive Mahalanobis distance oversampling (AMDO) was proposed. AMDO integrates generalized singular value decomposition with MDO to solve the oversampling problem encountered in mixed-type imbalanced data sets. Either MDO or AMDO can be utilized as a direct learning approach for solving problems with multi-class imbalanced problems.

The oversampling results obtained from MDO or AMDO are equivalent to those obtained by placing minority class samples and generated synthetic samples into the principal component space. The minority class samples and the synthetic samples can be considered to be included in an ellipsoid centered at the class mean. The orientation of the ellipsoid depends on the covariance structure of the minority class samples. The synthetic samples do not change the covariance structure of the minority class because all the synthetic samples are generated within the ellipsoid. However, both MDO and AMDO use only one ellipsoid to include the minority class samples and synthetic samples. If the decision regions of the minority class are separated, the decision region approximated using only one ellipsoid may overlap with the decision regions of other classes. This is especially true for imbalanced multi-class data. Samples from different classes may be included in a single ellipsoid structure depending on the target minority class samples. The synthetic samples generated by MDO or AMDO are randomly assigned in the single ellipsoid only if they have the same Mahalanobis distance as that of the associated minority sample. When synthetic samples are generated within a single ellipsoid, the generated synthetic samples tend to be placed in the cluster of samples belonging to other classes. This reduces the effectiveness of oversampling. Moreover, certain decision regions (e.g., those that are ring- or belt-shaped) are difficult to approximate with only one ellipsoid.

A novel approach called evolutionary Mahalanobis distance oversampling (EMDO) is proposed in this paper to overcome the limitations of MDO and AMDO. EMDO utilizes multiple ellipsoids to learn the distribution and orientation of minority class samples in parallel. Gustafson and Kessel proposed a clustering algorithm called the Gustafson–Kessel algorithm (GKA) [44], which is similar to the widely used fuzzy *c*-means [45] clustering approach with Mahalanobis norms. The advantage of the GKA over fuzzy c-means is that it utilizes the Mahalanobis norm instead of the Euclidean norm to learn the underlying sample distribution. However, the GKA assumes a fixed volume before learning the center and orientation of every ellipsoid. The GKA is an effective clustering approach for learning the centers and orientations of data clusters, but it is unsuitable for learning the decision regions of data due to its assumption of a fixed ellipsoid size. The GKA was modified in [46,47] to adaptively learn ellipsoid sizes for pattern recognition problems by using the genetic algorithm with a single objective function. In the proposed EMDO, the GKA is integrated with multi-objective particle swarm optimization (MOPSO) [48,49] to ensure that the centers, orientations, and sizes of multiple ellipsoids, along with the overall misclassification error, are learned in parallel. The misclassification error is defined as the total number of misclassified samples included in a union of multiple ellipsoids. Therefore, EMDO can learn a set of ellipsoids to approximate connected or disconnected complex decision regions with reasonable accuracy. Because multiple ellipsoids are learned in parallel in EMDO, an effective approach is designed to adaptively determine the number of synthetic samples to be generated in every ellipsoid. Similar ideas that design suitable algorithms to search for model parameters for specific applications are shown in [50–52].

The technical novelty and main contribution of this paper are summarized as follows.

1) An effective novel oversampling approach called EMDO is proposed for multi-class imbalanced data problems. Different from the MDO and AMDO approaches that use only one ellipsoid, EMDO learns multiple ellipsoids in parallel to approximate the decision region of the target minority class samples.
2) MOPSO is utilized along with GKA in EMDO to optimize the parameters, including the centers, orientations, and sizes of multiple ellipsoids approximating the target class of decision regions with reasonable accuracy.
3) Synthetic minority samples are generated based on the Mahalanobis distance within every ellipsoid learned by EMDO. A novel adaptive approach is proposed to determine the number of synthetic minority samples to be generated based on the density of minority samples in every ellipsoid.
4) EMDO was evaluated and found to perform better than other widely used oversampling schemes.

The remainder of this paper is organized as follows. Section 2 presents the problem formulation of oversampling for imbalanced data. The GKA is introduced in this section, and it shows that the GKA is suitable to solve the problem formulated herein. Section 3 introduces the proposed multi-objective optimization scheme designed in the EMDO, which uses MOPSO. Section 4 details the method for calculating the number of ellipsoids required to approximate the decision regions of every class. Section 5 describes performance evaluation of EMDO against other widely used oversampling schemes. Finally, Section 6 concludes the study.

2. Problem Statement and GKA

Given a data set $\boldsymbol{S} = \left\{ (\boldsymbol{x_i}, y_i) \,|\boldsymbol{x_i} \in R^d,\ y_i \in \{1 \ldots p\}, i = 1 \ldots N \right\}$, every ith sample $\boldsymbol{x_i} \in \boldsymbol{S}$ belongs to some class y_i among p classes. Let $\boldsymbol{S_j} \subset \boldsymbol{S}$ be the set containing the samples belonging to class j, $j = 1 \ldots p$. Denote $N_j \equiv \left|\boldsymbol{S_j}\right|$ as the number of samples in $\boldsymbol{S_j}$, where $N_{\min} = \min\limits_{j=1\ldots p}(N_j)$, $N_{\max} = \max\limits_{j=1\ldots p}(N_j)$. The data set $\boldsymbol{S}$ is imbalanced if $(N_j/N_{\max})$ is less than a preset imbalance ratio, IR. The value of IR is determined based on the size of $\boldsymbol{S}$ and on the characteristics of the classification problem. Typically, IR ≥ 1.5. $\boldsymbol{S_j}$ is called a minority set if $(\left|\boldsymbol{S_j}\right|/N_{\max}) < IR$, $j = 1 \ldots p$. An oversampling technique is applied in this study to overcome the skewed distribution of samples in $\boldsymbol{S}$. If $\boldsymbol{S_j}$ is a minority set, the synthetic samples belonging to the same jth class are generated in $\boldsymbol{S_j}$ to form an enlarged set $\widetilde{\boldsymbol{S}}_j$ such that $\left|\widetilde{\boldsymbol{S}}_j\right| = N_{\max}/IR$. Denote $\widetilde{N}_j$ as the total number of extra synthetic samples generated to balance the minority set $\boldsymbol{S_j}$,

$$\widetilde{N}_j = (N_{\max}/IR - \left|\boldsymbol{S_j}\right|). \tag{1}$$

Note that there can be more than one minority set in a multi-class problem. An oversampling technique is proposed herein to improve classification accuracy on an imbalanced data set. To generate an adequate number of synthetic samples and place them in the minority sets, the distribution of decision regions of every minority set in the d-dimensional feature space must be located. Multiple ellipsoids are utilized in this study to approximate the decision regions of minority sets. EMDO is proposed to learn these ellipsoids and generate synthetic samples in these ellipsoids for oversampling.

Assume that ellipsoids approximate the decision region of the jth class samples. Denote the center of every nth ellipsoid as $\boldsymbol{v_n^j} \in R^d$. The distance between every kth sample $\boldsymbol{x_k}$ and the ellipsoid center $\boldsymbol{v_n^j}$ is defined in the Mahalanobis form as follows:

$$\lambda_{nk}^j = \left((\boldsymbol{x_k} - \boldsymbol{v_n^j})^T \boldsymbol{M_n^j} (\boldsymbol{x_k} - \boldsymbol{v_n^j})\right)^{1/2}, \tag{2}$$

where $\boldsymbol{M_n^j} \in R^{d \times d}$ is a norm-inducing matrix. The ellipsoid Φ_n^j is defined as follows by using the Mahalanobis distance defined in (2):

$$\boldsymbol{\Phi_n^j}(\boldsymbol{x_k}) = (\boldsymbol{x_k} - \boldsymbol{v_n^j})^T \boldsymbol{M_n^j} (\boldsymbol{x_k} - \boldsymbol{v_n^j}) = 1. \tag{3}$$

The sample $\boldsymbol{x_k}$ is inside or on the ellipsoid if $\boldsymbol{\Phi_n^j}(\boldsymbol{x_k}) \leq 1$, but it is outside the ellipsoid if $\boldsymbol{\Phi_n^j}(\boldsymbol{x_k}) > 1$. Let the decision region of the jth class samples in the feature space be denoted as $\Re^j$; $\Re^j$ is approximated by the union of α^j ellipsoids, that is,

$$\Re^j \cong \bigcup_{n=1\ldots\alpha^j} \boldsymbol{\Phi_n^j}. \tag{4}$$

The GKA is used for learning the α^j ellipsoids in parallel, given that the size of each ellipsoid is assigned. Denote the size of the ellipsoid Φ_n^j in (3) as ξ_n^j, $n = 1 \ldots \alpha^j$. The determinant of the norm-inducing matrix $\boldsymbol{M_n^j}$ is inversely proportional to ξ_n^j. Therefore,

$$\det(\boldsymbol{M_n^j}) = 1/\xi_n^j,\ n = 1 \ldots \alpha^j. \tag{5}$$

The GKA learns the norm-inducing matrices $\boldsymbol{M_n^j}$ and the ellipsoid centers $\boldsymbol{v_n^j}$ through iteratively calculating an auxiliary fuzzy partition matrix $\boldsymbol{U^j} \in R^{\alpha^j \times N^j}$ by using all N^j samples belonging to the jth class. The element $\mu_{nk}^j \in \boldsymbol{U^j}$ represents the membership value of the kth sample $\boldsymbol{x_k}$ associated with the nth ellipsoid Φ_n^j. The membership values sum to 1 for every $\boldsymbol{x_k}$, that is,

$$\sum_{n=1}^{\alpha^j} \mu_{nk}^j = 1, k = 1 \ldots N^j. \tag{6}$$

The GKA is a fast iterative learning algorithm that efficiently updates the membership values in the fuzzy partition matrix $\boldsymbol{U^j}$ while learning both the norm-inducing matrix $\boldsymbol{M_n^j}$ and the center $\boldsymbol{v_n^j}$ of every nth ellipsoid. Note that the GKA learns all α^j ellipsoids in parallel. Denote the matrices containing the ellipsoid centers and the norm-inducing matrices as $\boldsymbol{V^j}$ and $\boldsymbol{M^j}$, respectively, that is, $\boldsymbol{V^j} = [\boldsymbol{v_1^j}, \boldsymbol{v_2^j} \ldots, \boldsymbol{v_{\alpha^j}^j}]$ and $\boldsymbol{M^j} = [\boldsymbol{M_1^j}, \boldsymbol{M_2^j}, \ldots, \boldsymbol{M_{\alpha^j}^j}]$. All elements in the triple $\left(\boldsymbol{U^j}, \boldsymbol{V^j}, \boldsymbol{M^j}\right)$ are learned by iteratively minimizing the distance in (2) weighted with the membership values in $\boldsymbol{U^j}$ subject to the constraints in (5) and (6). Let ω_n^j, $n = 1 \ldots \alpha^j$ and ϖ_k^j, $k = 1 \ldots N^j$ be the Lagrange multipliers of the constraints in (5) and (6), respectively. The triple $(\boldsymbol{U^j}, \boldsymbol{V^j}, \boldsymbol{M^j})$ is iteratively learned as follows:

$$(\boldsymbol{U^j}, \boldsymbol{V^j}, \boldsymbol{M^j}) = \text{argmin}(\sum_{n=1}^{\alpha^j}\sum_{k=1}^{N^j} \left(\mu_{nk}^j\right)^b \left(\lambda_{nk}^j\right)^2 + \sum_{n=1}^{\alpha^j} \omega_n^j(\det(\boldsymbol{M_n^j}) - 1/\xi_n^j) + \sum_{k=1}^{N^j} \varpi_k^j(\sum_{n=1}^{\alpha^j} (\mu_{nk}^j - 1)), \tag{7}$$

where b is an adjusted weighting index. The optimization described in (7) is realized by differentiating (7) with respect to μ_{nk}^j, $\boldsymbol{v_n^j}$, ω_n^j, and ϖ_k^j,and by equating the result to 0.

The parameters are obtained as follows:

$$\mu_{nk}^j = \frac{1}{\sum_{i=1}^{\alpha^j} \left(\lambda_{nk}^j / \lambda_{ik}^j\right)^{2/(b-1)}},\ n = 1 \ldots \alpha^j,\ k = 1 \ldots N^j; \tag{8}$$

$$\boldsymbol{v_n^j} = \frac{\sum_{k=1}^{N^j} \left(\mu_{nk}^j\right)^b \boldsymbol{x_k}}{\sum_{k=1}^{N^j} \left(\mu_{nk}^j\right)^b},\ n = 1 \ldots \alpha^j; \tag{9}$$

$$\boldsymbol{F_n^j} = \frac{\sum_{k=1}^{N^j} \left(\mu_{nk}^j\right)^b (\boldsymbol{x_k} - \boldsymbol{v_n^j})(\boldsymbol{x_k} - \boldsymbol{v_n^j})^T}{\sum_{k=1}^{N^j} \left(\mu_{nk}^j\right)^b},\ n = 1 \ldots \alpha^j; \tag{10}$$

$$\text{and} \boldsymbol{M_n^j} = \left(\zeta_n^j \det(\boldsymbol{M_n^j})\right)^{1/d} \left(\boldsymbol{M_n^j}\right)^{-1}\ n = 1 \ldots \alpha^j. \tag{11}$$

The iteration in GKA is stopped when no significant improvement is made in the fuzzy partition matrix $\boldsymbol{U^j}$. Let $\left(\boldsymbol{U^j}\right)^{(m)}$ be the fuzzy partition matrix learned in the mth iteration, the norm of the difference between $\left(\boldsymbol{U^j}\right)^{(m)}$ and $\left(\boldsymbol{U^j}\right)^{(m+1)}$ can be defined as

$$\delta^j = \left\| \left(\boldsymbol{U^j}\right)^{(m+1)} - \left(\boldsymbol{U^j}\right)^{(m)} \right\| \equiv \max_{n,k} \left| \left(\mu_{nk}^j\right)^{(m+1)} - \left(\mu_{nk}^j\right)^{(m)} \right|. \tag{12}$$

The GKA iteratively learns $\boldsymbol{U^j}$, $\boldsymbol{V^j}$, and $\boldsymbol{M^j}$ until $\delta^j < \varepsilon^j$, where ε^j is a small constant. The flowchart of the GKA is illustrated in Figure 1.

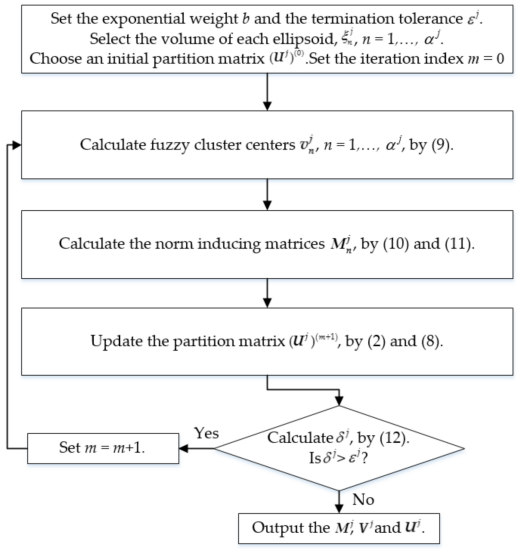

Figure 1. Flow chart of GKA.

3. Multi-Objective Optimization in EMDO

As depicted in Figure 1 and described in Section 2, the GKA optimizes the centers and norm-inducing matrices of multiple ellipsoids in parallel with a preset size of every ellipsoid. If the ellipsoid size is set inappropriately, the ellipsoids learned by the GKA cannot accurately include all minority class samples. According to (4), each jth-class decision region $\Re^j$ is approximated by the union of α^j ellipsoids Φ_n^j of size ξ_n^j, $n = 1 \ldots \alpha^j$. Consider the sets $\boldsymbol{\Phi^j} = \left\{\Phi_1^j, \Phi_2^j, \ldots, \Phi_{\alpha^j}^j\right\}$ and $\boldsymbol{\Xi^j} = \left\{\xi_1^j, \xi_2^j, \ldots, \xi_{\alpha^j}^j\right\}$. The distance between the kth sample $\boldsymbol{x_k}$ and $\Re^j$, denoted $L(\boldsymbol{x_k}, \Re^j)$, can be defined as the minimum distance between $\boldsymbol{x_k}$ and the center of each ellipsoid, as follows:

$$L(\boldsymbol{x_k}, \Re^j) \cong L(x_k, \boldsymbol{\Phi^j}) = \min_{n=1\ldots\alpha^j} \lambda_{nk}^j, \tag{13}$$

where λ_{nk}^j is defined in (2). The sample $\boldsymbol{x_k}$ is included in $\Re^j$ if $L(x_k, \Re^j) \leq 1$ and the corresponding class $y_k = j$. Denote a binary function $H(\cdot)$ as follows:

$$H(o) = \begin{cases} 1, \text{ if the logic statement } o \text{ is true;} \\ 0, \text{ if the logic statement } o \text{ is false.} \end{cases} \tag{14}$$

The total number of jth-class samples included in the set $\boldsymbol{\Phi^j}$ can be calculated as

$$O_{included}^j(\boldsymbol{\Phi^j}) = \sum_{k=1}^{N} H(L(\boldsymbol{x_k}, \boldsymbol{\Phi^j}) \leq 1 \text{ and } y_k = j). \tag{15}$$

It is possible that several samples that do not belong to the jth class are included in the set of ellipsoids $\boldsymbol{\Phi^j}$. The total number of samples not belonging to the jth class but included in $\boldsymbol{\Phi^j}$ can be calculated as

$$O_{included}^{\setminus j}(\boldsymbol{\Phi^j}) = \sum_{k=1}^{N} H(L(\boldsymbol{x_k}, \boldsymbol{\Phi^j}) \leq 1 \text{ and } y_k \neq j), \tag{16}$$

where $N_{included}^{\setminus j}(\cdot) \leq (N - N^j)$.

Referring to (5), the total size of the ellipsoids contained in $\mathbf{\Phi}^j$ can be calculated as

$$\mathbb{F}_1(\Xi^j) = \sum_{n=1}^{\alpha^j} \xi_n^j. \tag{17}$$

The proposed EMDO not only minimizes the total ellipsoid sizes but also aims to simultaneously maximize the number of jth class samples included in the set of ellipsoids $\mathbf{\Phi}^j$ and minimize the number of samples included in $\mathbf{\Phi}^j$ but not belonging to jth class. The misclassification error can be defined as the summation of the number of jth class samples not included in $\mathbf{\Phi}^j$, calculated as $(N^j - O^j_{included}(\mathbf{\Phi}^j|_{\Xi^j})$, and the number of samples that are included in $\mathbf{\Phi}^j$ but that do not belong to the jth class is calculated as $O^{\setminus j}_{included}(\mathbf{\Phi}^j|_{\Xi^j})$. Therefore, the misclassification error can be defined as

$$\mathbb{F}_2(\mathbf{\Phi}^j|_{\Xi^j}) = N^j - O^j_{included}(\mathbf{\Phi}^j|_{\Xi^j}) + O^{\setminus j}_{included}(\mathbf{\Phi}^j|_{\Xi^j}). \tag{18}$$

The misclassification error can be utilized as an objective function to optimize the ellipsoid sizes. The set of ellipsoid sizes $\Xi^j = [\xi_1^j, \xi_2^j, \ldots, \xi_{\alpha^j}^j]$ can be optimized using a multi-objective optimization scheme that minimizes both (17) and (18).

MOPSO is utilized to perform this multi-objective optimization by searching for the best set of ellipsoid sizes Ξ^j by minimizing the objective functions $\mathbb{F}_1(\cdot)$ and $\mathbb{F}_2(\cdot)$. Assume that G particles are utilized in the MOPSO. Denote $\Xi^j(k,g)$ as the gth particle in the kth iteration and $\overline{\Xi}^j(g)$ as the non-dominated solution subject to the following multi-objective optimization. Note that the multi-objective optimization searches for the non-dominated solution of every particle.

$$\overline{\Xi}^j(g) = \underset{\Xi^j(k,g),\ \forall \boldsymbol{x}^k \in S,\ g=1\ldots G}{Arg\min} (\mathbb{F}_1(\Xi^j(k,g)), \mathbb{F}_2(\mathbf{\Phi}^j|_{\Xi^j(k,g)})). \tag{19}$$

$\Xi^j(k,g)$ is defined to be a non-dominated solution, as given by (19), if it is not dominated by any other particle, that is, if both $\mathbb{F}_1(\Xi^j(k,g)) \leq \mathbb{F}_1(\Xi^j(k,i))$ and $\mathbb{F}_2(\mathbf{\Phi}^j|_{\Xi^j(k,g)}) \leq \mathbb{F}_2(\mathbf{\Phi}^j|_{\Xi^j(k,i)})$, i=1 … G, $i \neq g$. Let

$$a_i(\Xi^j(k,g)) = H(\mathbb{F}_1(\Xi^j(k,g)) > \mathbb{F}_1(\Xi^j(k,i)) \text{ or } \mathbb{F}_2(\mathbf{\Phi}^j|_{\Xi^j(k,g)}) > \mathbb{F}_2(\mathbf{\Phi}^j|_{\Xi^j(k,i)})) \tag{20}$$

$$A(\Xi^j(k,g)) = \sum_{i=1\ldots G, i\neq g} a_i(\Xi^j(k,g)). \tag{21}$$

The non-dominated solution obtained using the gth particle, denoted as $\overline{\Xi}^j(g)$, is updated as $\Xi^j(k,g)$ if $A(\Xi^j(k,g))$=0. However, $\overline{\Xi}^j(g)$ remains unchanged if $A(\Xi^j(k,g))$>0.

$$\overline{\Xi}^j(g) = \begin{cases} \Xi^j(k,g), \text{ if } A(\Xi^j(k,g)) = 0; \\ \overline{\Xi}^j(g), \text{ otherwise.} \end{cases} \tag{22}$$

Only if $\Xi^j(k,g)$ is the non-dominated solution of (19) will it be included in the repository S^j_{rep}, which is the set of all of non-dominated solutions of (19). The best solution achieved by the gth particle, denoted $\Xi^j_{p\_best}(g)$, is updated using this non-dominated solution generated by the gth particle, that is, $\Xi^j_{p\_best}(g) = \overline{\Xi}^j(g)$. With reference to (22), if $\overline{\Xi}^j(g)$ is not generated in the current kth iteration, $\Xi^j_{p\_best}(g)$ remains unchanged.

It is possible that several original non-dominated solutions stored in S^j_{rep} become dominated after a new non-dominated solution is included in S^j_{rep}. A process for filtering out non-dominated solutions, similar to the one proposed in (20) and (21), is executed for

all of the non-dominated solutions in $\boldsymbol{S}_{\boldsymbol{rep}}^{\boldsymbol{j}}$. Assume a total of m_{rep} non-dominated solutions are in the repository, including the newly generated one. Each of the m_{rep} non-dominated solutions is compared with all of the other solutions to evaluate whether they are being dominated. Denote $\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}$ as the mth non-dominated solution. Let

$$a_i(\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}) = H(\mathbb{F}_1(\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}) > \mathbb{F}_1(\overline{\boldsymbol{\Xi}}_{\boldsymbol{i}}^{\boldsymbol{j}}) \text{ or } \mathbb{F}_2(\boldsymbol{\Phi}^{\boldsymbol{j}}|_{\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}}) > \mathbb{F}_2(\boldsymbol{\Phi}^{\boldsymbol{j}}|_{\overline{\boldsymbol{\Xi}}_{\boldsymbol{i}}^{\boldsymbol{j}}})) \tag{23}$$

$$A(\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}) = \sum_{i=1\ldots m_{rep}, i\neq m} a_i(\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}). \tag{24}$$

$\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}$ is no longer a non-dominated solution and is excluded from $\boldsymbol{S}_{\boldsymbol{rep}}^{\boldsymbol{j}}$ if $A(\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}) > 0$.

An adaptive grid algorithm [53] is applied to $\boldsymbol{S}_{\boldsymbol{rep}}^{\boldsymbol{j}}$ after the filtering process is completed, as given in (23) and (24), to place all the non-dominated solutions into several grids. The global best particle $\boldsymbol{\Xi}_{\boldsymbol{g\_best}}^{\boldsymbol{j}}$ is randomly selected from among the grids in $\boldsymbol{S}_{\boldsymbol{rep}}^{\boldsymbol{j}}$ by using the roulette wheel selection scheme. After both $\boldsymbol{\Xi}_{\boldsymbol{p\_best}}^{\boldsymbol{j}}(g)$ and $\boldsymbol{\Xi}_{\boldsymbol{g\_best}}^{\boldsymbol{j}}$ are determined, each particle is updated as follows:

$$\boldsymbol{\tau}(k,g) = \boldsymbol{\tau}(k-1,g) + c_1\gamma_1(\boldsymbol{\Xi}_{\boldsymbol{p\_best}}^{\boldsymbol{j}}(g) - \boldsymbol{\Xi}^{\boldsymbol{j}}(k,g)) + c_2\gamma_2(\boldsymbol{\Xi}_{\boldsymbol{g\_best}}^{\boldsymbol{j}} - \boldsymbol{\Xi}^{\boldsymbol{j}}(k,g)), \tag{25}$$

$$\boldsymbol{\Xi}^{\boldsymbol{j}}(k,g) = \boldsymbol{\Xi}^{\boldsymbol{j}}(k-1,g) + \boldsymbol{\tau}(k,g),\ g = 1\ldots G, \tag{26}$$

where c_1 and c_2 are preset constants and $\gamma_1,\ \gamma_2 \in [0,1]$ are randomly generated real numbers. If no new non-dominated solution can be successfully included in $\boldsymbol{S}_{\boldsymbol{rep}}^{\boldsymbol{j}}$ after the filtering process for certain preset K_{thr} iterations, MOPSO is saturated, and the iterative learning of particles given by (19)–(26) is stopped.

The optimal solution of the multi-objective optimization problem in (19) is searched from $\boldsymbol{S}_{\boldsymbol{rep}}^{\boldsymbol{j}}$ after the iterative learning process is stopped. Let $\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}} = \left\{\overline{\xi}_{m1}^{j}, \overline{\xi}_{m2}^{j}, \ldots, \overline{\xi}_{m\alpha^j}^{j}\right\}$ be the mth non-dominated solution in $\boldsymbol{S}_{\boldsymbol{rep}}^{\boldsymbol{j}}$. The average density of the ellipsoids associated with $\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}$ is defined as

$$d(\boldsymbol{\Phi}^{\boldsymbol{j}}|_{\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}}) = \frac{O_{included}^{j}(\boldsymbol{\Phi}^{\boldsymbol{j}}|_{\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}})}{\sum_{n=1}^{\alpha^j} \overline{\xi}_{mn}^{j}}. \tag{27}$$

The optimal solution $(\boldsymbol{\Xi}^{\boldsymbol{j}})^*$ can be selected based on the average density of the non-dominated solutions because the ellipsoids associated with the optimal solution tend to have a small size but a large number of samples. However, the average density in (27) cannot be directly utilized as the optimization index for selecting the optimal solution. It is modified as the ratio of the total number of jth class samples included in the set $\boldsymbol{\Phi}^{\boldsymbol{j}}$ multiplied with the ratio of the total number of samples not belonging to the jth class but included in $\boldsymbol{\Phi}^{\boldsymbol{j}}$. Let the evaluation index for the mth non-dominated solution $\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}$ be $\sigma^j(\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}})$, which is defined based on (27) as follows:

$$\sigma^j(\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}) = d(\boldsymbol{\Phi}^{\boldsymbol{j}}|_{\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}}) \times \frac{O_{included}^{j}(\boldsymbol{\Phi}^{\boldsymbol{j}}|_{\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}})}{N^j} \times \frac{(N - N^j - O_{included}^{\setminus j}(\boldsymbol{\Phi}^{\boldsymbol{j}}|_{\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}}))}{N - N^j}, \tag{28}$$

where $N^{\setminus j}(\boldsymbol{\Phi}^{\boldsymbol{j}}|_{\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}})$ denotes the number of samples not belonging to the jth class but included in $\boldsymbol{\Phi}^{\boldsymbol{j}}$. The optimal solution $(\boldsymbol{\Xi}^{\boldsymbol{j}})^*$ can be defined as the set of ellipsoid sizes that maximize the index $\sigma^j(\cdot)$, that is,

$$(\boldsymbol{\Xi}^{\boldsymbol{j}})^* = \underset{\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}},\ m=1\ldots m_{rep}}{Arg\max}\ \sigma^j(\overline{\boldsymbol{\Xi}}_{\boldsymbol{m}}^{\boldsymbol{j}}). \tag{29}$$

After the optimal ellipsoid sizes are determined by MOPSO according to (19) and (29), the GKA is applied based on the optimal ellipsoid sizes $(\Xi^j)^*$ to calculate the other optimal ellipsoid parameters such as the norm-inducing matrix $(\boldsymbol{M}^j)^*$ and the ellipsoid centers $(\boldsymbol{V}^j)^*$. Note that the orientations of all the ellipsoids approximating the jth class decision are determined by the norm-inducing matrix $(\boldsymbol{M}^j)^*$. The proposed MOPSO integrated with the GKA is illustrated in Figure 2.

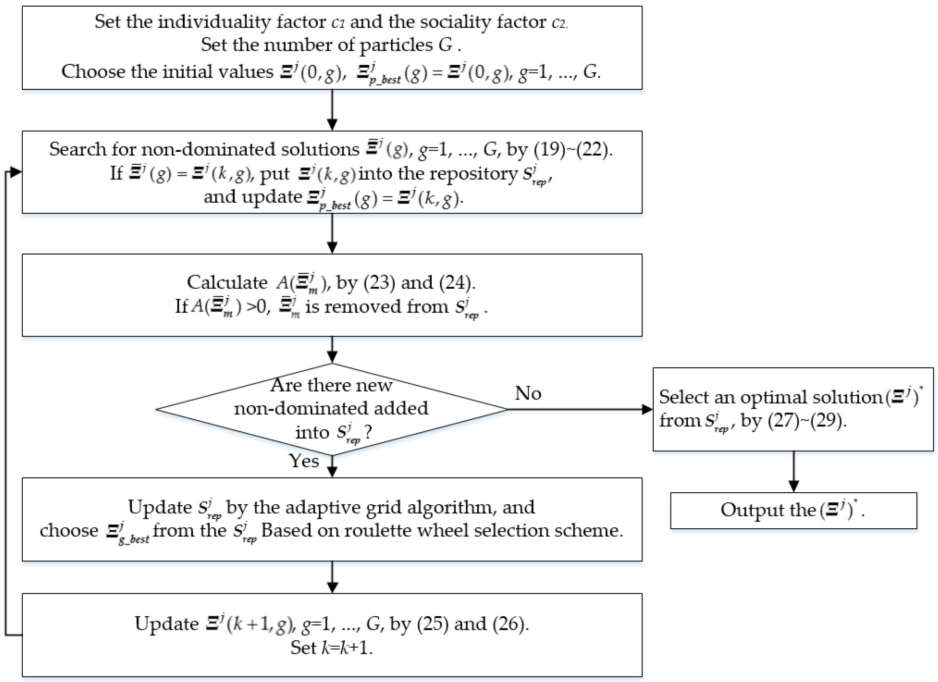

Figure 2. Flow chart of MOPSO.

4. Determining Number of Ellipsoids

The ellipsoid parameters, such as size, centers, and orientation, are optimized using MOPSO integrated with the GKA, as described in Sections 2 and 3. These ellipsoid parameters are calculated under the condition that the total number of ellipsoids α^j used to approximate the jth class decision region is assigned in advance $j = 1 \ldots p$. If α^j is too small, the jth class samples might be included in an insufficient number of ellipsoids, resulting in several ellipsoids having large sizes. It is possible that certain samples that do not belong to the jth class are included in these large ellipsoids. Moreover, samples other than those belonging to the jth class might be included because an insufficient number of ellipsoids is assigned to model the jth class samples. A binary-class dataset with 1000 samples in each class is illustrated in Figure 3. The samples in either class 1 or class 2 are randomly generated within the range [0, 3] on X and Y axis, respectively. The problem caused by a small α^j is illustrated in Figure 3a. Conversely, the jth class samples might be included in too many ellipsoids if α^j is too large. This results in a scenario where several ellipsoids overlap with each another, resulting in the ellipsoids being learned inefficiently, as illustrated in Figure 3b. If a suitable α^j is assigned, as illustrated in Figure 3c, the learning result leads to a set of ellipsoids with of the appropriate size, center, and orientation.

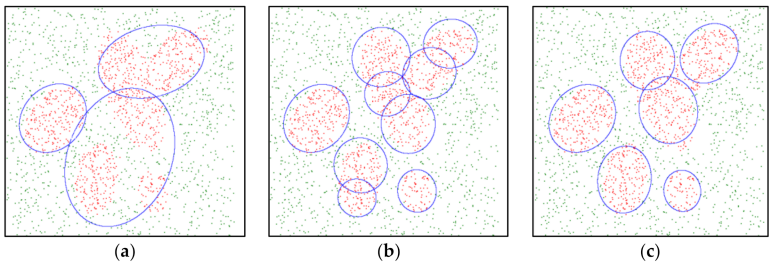

Figure 3. Learning results with different number of ellipsoids α^j. (**a**) $\alpha^j = 3$; (**b**) $\alpha^j = 9$; (**c**) $\alpha^j = 6$.

To determine the suitable number of ellipsoids, the ellipsoid parameters are optimized using MOPSO integrated with the GKA by setting α^j from 1 to an appropriate number q. Denote $(\Xi^j)^*\big|_{\alpha^j=i}$ as the optimal ellipsoid sizes calculated using MOPSO, according to (29), with α^j set to i ellipsoids, $i = 1 \ldots q$. The index $\sigma^j((\Xi^j)^*\big|_{\alpha^j=i})$ in (28) is utilized to evaluate the effectiveness and efficiency for different numbers of ellipsoids α^j. The suitable number of ellipsoids α^j can be determined at the value corresponding to the corner of the curve $\sigma^j((\Xi^j)^*\big|_{\alpha^j=i})$ with respect to α^j. Figure 4 shows a typical curve $\sigma^j((\Xi^j)^*\big|_{\alpha^j=i})$ with respect to α^j. According to this figure, $\alpha^j = 6$ is a suitable choice because the curve corner appears at $\alpha^j = 6$.

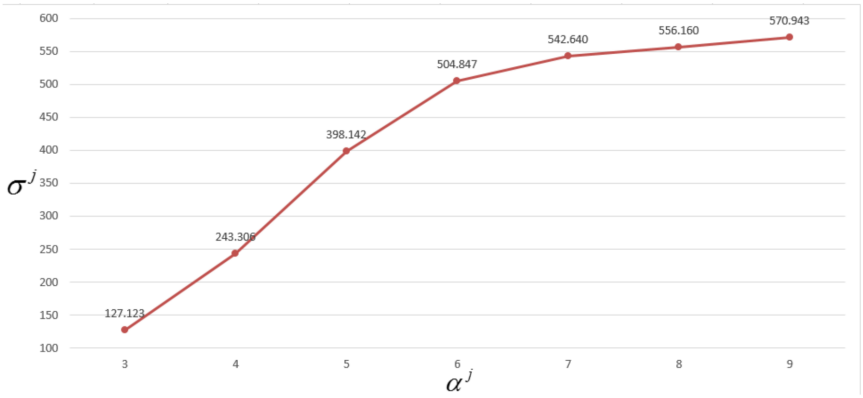

Figure 4. The typical curve of $\sigma^j((\Xi^j)^*\big|_{\alpha^j=i})$ vs. α^j.

5. Generating Synthetic Samples

Any set of jth class samples with $(|S_j|/N_{\max}) < IR$, $j = 1 \ldots p$, is considered a minority set. With reference to (1), $\widetilde{N}_j$ synthetic samples are to be generated and added into the minority set. Recall that the minority set of the jth class samples is approximated by α^j ellipsoids. The $\widetilde{N}_j$ synthetic samples must be proportionally added into each of the α^j ellipsoids based on the density d_n^j of every nth ellipsoid, $n = 1 \ldots \alpha^j$. The number of jth class samples in the nth ellipsoid Φ_n^j is defined in a manner similar to (15) as

$$O_{included}^{j}(\Phi_n^j) = \sum_{k=1}^{N} H(\lambda_{nk}^j \leq 1 \text{ and } y_k = j) \tag{30}$$

The number of samples included in the nth ellipsoid but not belonging to the jth class is

$$O_{included}^{\setminus j}(\Phi_n^j) = \sum_{k=1}^{N} H(\lambda_{nk}^j \leq 1 \text{ and } y_k \neq j). \tag{31}$$

The density of ellipsoid Φ_n^j is defined as

$$d(\Phi_n^j) = \frac{O_{included}^j(\Phi_n^j)}{\xi_n^j}. \tag{32}$$

The weight of the ellipsoid Φ_n^j for sharing the generated synthetic samples is defined as the reciprocal of the density $d(\Phi_n^j)$ modified by the ratio of $O_{included}^j(\Phi_n^j)$ to the total number samples in Φ_n^j:

$$\beta_n^j = \frac{1}{d(\Phi_n^j)} \times \frac{O_{included}^j(\Phi_n^j)}{O_{included}^j(\Phi_n^j) + O_{included}^{\setminus j}(\Phi_n^j)}. \tag{33}$$

Denote the number of synthetic samples added to Φ_n^j as $\widetilde{N}_n^j$, which is determined based on the weight β_n^j given in (33)

$$\widetilde{N}_n^j = \widetilde{N}^j \times \frac{\beta_n^j}{\sum_{i=1}^{\alpha^j} \beta_i^j},\ n = 1 \ldots \alpha^j. \tag{34}$$

The scheme for generating synthetic samples for every ellipsoid Φ_n^j is designed to resolve the oversampling problem for the following two scenarios:

(A) $O_{included}^j(\Phi_n^j)/(O_{included}^j(\Phi_n^j) + O_{included}^{\setminus j}(\Phi_n^j)) \geq 0.9$

In this case, more than 90% of the samples in Φ_n^j belong to the minority class. The samples other than those belonging to the *j*th class can be considered noise. The generated synthetic *j*th class samples do not affect the classification accuracy if they are randomly included in the ellipsoid Φ_n^j. According to (2), $\Delta_n^j = (\lambda_{nk}^j)^2 = (\boldsymbol{x_k} - \boldsymbol{v_n^j})^T \boldsymbol{M_n^j} (\boldsymbol{x_k} - \boldsymbol{v_n^j})$. Denote $\boldsymbol{z_{ni}^j}$ as the *i*th eigenvector of $\boldsymbol{M_n^j}$ corresponding to the *i*th eigenvalue ψ_{ni}^j, $i = 1 \ldots d$. Let $\boldsymbol{Z_n^j} = [\boldsymbol{z_{n1}^j}, \ldots, \boldsymbol{z_{\mathrm{nd}}^j}]$, $\boldsymbol{\Psi_n^j} = diag(\psi_{n1}^j, \ldots, \psi_{nd}^j)$. Because $\boldsymbol{M_n^j}$ is a positive-definite matrix,

$$\Delta_n^j = (\lambda_{nk}^j)^2 = (\boldsymbol{x_k} - \boldsymbol{v_n^j})^T \boldsymbol{M_n^j} (\boldsymbol{x_k} - \boldsymbol{v_n^j}),\ n = 1 \ldots \alpha^j. \tag{35}$$

Let $p_{ni}^j = (\boldsymbol{z_{ni}^j})^T (\boldsymbol{x_k} - \boldsymbol{v_n^j})$, $i = 1 \ldots d$; then, (35) can be rewritten as

$$\Delta_n^j = \sum_{i=1}^{d} (p_{ni}^j)^2 \psi_{ni}^j,\ n = 1 \ldots \alpha^j. \tag{36}$$

Note that the sample $\boldsymbol{x_k}$ is considered to be included in the ellipsoid if $\Delta_n^j \leq 1$. The ellipsoid has the center $\boldsymbol{v_n^j}$, and all the eigenvectors $\boldsymbol{z_{ni}^j}$, $i = 1 \ldots d$, are orthogonal to one another. According to (36), every *i*th orthogonal eigenvector intersects the ellipsoid's boundary sphere, where $\Delta_n^j = 1$ at $1/\sqrt{\psi_{ni}^j}$ and $-1/\sqrt{\psi_{ni}^j}$.

The synthetic samples $\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{\boldsymbol{j}}$ are randomly generated in ellipsoid Φ_n^j. The generated synthetic sample $\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{\boldsymbol{j}}$ is expressed as a linear combination of eigenvectors because all eigenvectors $\boldsymbol{z_{ni}^j}$ are orthogonal axes of the ellipsoid, that is,

$$\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{\boldsymbol{j}} = \sum_{i=1}^{d} \boldsymbol{z_{ni}^j} b_{ni}, \tag{37}$$

where b_{ni} is the projection of the vector $(\hat{\boldsymbol{x}}_k - \boldsymbol{v}_n^j)$ onto the eigenvector $\boldsymbol{z}_{ni}^j$. To ensure random generation of the synthetic samples inside Φ_n^j, b_{ni} is set to be a random number within the range.

$$-1/\sqrt{\psi_{ni}^j} \leq b_{ni}^j \leq 1/\sqrt{\psi_{ni}^j} \tag{38}$$

because each of the eigenvectors intersects the boundary sphere at $1/\sqrt{\psi_{ni}^j}$ and $-1/\sqrt{\psi_{ni}^j}$. However, the approach to randomly generate synthetic samples, as expressed in in (37) and (38), does not guarantee that the generated synthetic sample $\hat{\boldsymbol{x}}_k^j$ is always in the ellipsoid Φ_n^j. For every randomly generated $\hat{\boldsymbol{x}}_k^j$, calculate the Mahalanobis distance according to (2) as

$$\hat{\lambda}_{nk}^j = \left((\hat{\boldsymbol{x}}_k^j - \boldsymbol{v}_n^j)^T \boldsymbol{M}_n^j (\hat{\boldsymbol{x}}_k^j - \boldsymbol{v}_n^j)\right)^{1/2}. \tag{39}$$

The generated $\hat{\boldsymbol{x}}_k^j$ is in Φ_n^j if $\hat{\lambda}_{nk}^j \leq 1$. No additional processing is required if $\hat{\boldsymbol{x}}_k^j$ is in Φ_n^j. The generated $\hat{\boldsymbol{x}}_k^j$ is outside Φ_n^j if $\hat{\lambda}_{nk}^j > 1$. Further processing is required if $\hat{\boldsymbol{x}}_k^j$ is outside Φ_n^j. The multiplication of $\hat{\boldsymbol{x}}_k^j$ with a random number κ, where $0 < \kappa \leq (1/\hat{\lambda}_{nk}^j)$, leads to the multiplicative product $\kappa\hat{\boldsymbol{x}}_k^j$ in Φ_n^j. Denote the finally determined synthetic samples as $\widetilde{\boldsymbol{x}}_k^j$

$$\widetilde{\boldsymbol{x}}_k^j = \begin{cases} \hat{\boldsymbol{x}}_k^j, \text{ if } \hat{\lambda}_{nk}^j \leq 1; \\ \kappa_1 \hat{\boldsymbol{x}}_k^j, \text{ if } \hat{\lambda}_{nk}^j > 1; \end{cases}, \tag{40}$$

where κ_1 is $\boldsymbol{a}$ random number and $\kappa_1 \in (0, 1/\hat{\lambda}_{nk}^j]$.(B) $O_{included}^j(\Phi_n^j)/(O_{included}^j(\Phi_n^j) + O_{included}^{\setminus j}(\Phi_n^j)) < 0.9$

In this case, more samples not belonging to the jth class are in Φ_n^j. The random placement of synthetic samples in Φ_n^j, as in the previous case, cannot effectively improve the classification accuracy. Borderline SMOTE [30] is modified to generate synthetic samples in this case. The samples located at the borderline between the clusters belonging and not belonging to the jth class must be first identified using Borderline SMOTE. For every jth class sample $\boldsymbol{x}_k^j \in \Phi_n^j$, define the set $\boldsymbol{S}_k^j$ containing all m-nearest neighbors. The m-nearest neighbors of $\boldsymbol{x}_k^j$ are defined as the samples with the m-shortest Mahalanobis distances from $\boldsymbol{x}_k^j$. Note that the Mahalanobis distance is calculated using the same norm-inducing matrix $\boldsymbol{M}_n^j$ as in the case of Φ_n^j, that is,

$$\widetilde{\lambda}_{ki}^j = \left((\boldsymbol{x}_k^j - \boldsymbol{x}_i)^T \boldsymbol{M}_n^j (\boldsymbol{x}_k^j - \boldsymbol{x}_i)\right)^{1/2}, \ \forall \boldsymbol{x}_i \in \Phi_n^j, \text{ but } \boldsymbol{x}_i \neq \boldsymbol{x}_k^j. \tag{41}$$

The sample with all the m-nearest neighbors belonging to jth class is the sample not on the borderline. Conversely, the borderline sample contains at least one sample among the m-nearest neighbors not belonging to the jth class. Therefore, a sample is a borderline sample if at least one sample in the set of m-nearest neighbors $\boldsymbol{S}_k^j$ does not belong to the jth class for every $\boldsymbol{x}_k^j \in \Phi_n^j$.

After the borderline samples are identified, the synthetic samples are generated through random interpolation between the borderline sample $\boldsymbol{x}_k^j$ and any other $\boldsymbol{x}_l^j \in \boldsymbol{S}_k^j$; that is, the synthetic sample is generated as follows:

$$\widetilde{\boldsymbol{x}}_k^j = \boldsymbol{x}_k^j + \kappa_2(\boldsymbol{x}_k^j - \boldsymbol{x}_l^j), \ \forall \boldsymbol{x}_l^j \in \boldsymbol{S}_k^j, \tag{42}$$

where κ_2 is a random number and $\kappa_2 \in [0, 1]$.

6. Simulation

The proposed EMDO was evaluated against other multi-class imbalanced data learning algorithms on different numerical data sets. The classifier C4.5 is usually utilized

as the classifier to verify the oversampling results for various oversampling approaches. For instance, the oversampling approaches in [30,32,39], and [40–43] all used C4.5 as the classifier to verify the proposed oversampling schemes. This is mainly due to the fact that the classification results with C4.5 do not change as long as the parameter setting and datasets are fixed. No randomness exists in the classification results with C4.5 for the same parameter setting and dataset. Note that the number of ellipsoids utilized for the minority class is determined using the scheme proposed in Section 4 and is listed in the rightmost column of Table 1 and Table 6. The five nearest neighbors are considered for synthetic sample generation in case (B) of Section 5.

Table 1. Characteristics of the data sets for simulations in Example 1.

Data Set	Size	Attributes	Classes	Class Distribution	IR_{max}	No. of Ellipsoids
Balance	625	4	3	288/49/288	5.88	NA/4/NA
Hayes-Roth	132	4	3	51/51/30	1.7	NA/NA/3
New-Thyroid	215	5	3	150/35/30	5	NA/3/3
Page-Blocks	5472	10	5	4913/329/28/87/115	175.46	NA/9/4/6/9
Dermatology	358	34	6	111/60/71/48/48/20	5.55	NA/4/4/3/3/3
Breast-Tissue	106	9	6	21/15/18/16/14/22	1.57	NA/NA/NA/ NA/4/NA
User-Knowledge-Modelling (UKM)	403	5	5	50/102/129/122	2.58	4/NA/NA/NA
Vertebral-Column	310	6	3	60/150/100	2.5	6/NA/NA
Ecoli	327	7	5	143/77/52/35/20	7.15	NA/6/4/3/3

NA: not available.

The simulations in this study were conducted using five-fold cross-validation with 10 independent runs. Every data set was tested using different oversampling schemes and compared with the proposed EMDO. The minority class with the minimum size was selected to validate the effectiveness and efficiency of the proposed EMDO.

Several evaluation metrics were designed to evaluate the effectiveness and efficiency of the proposed EMDO. The classification accuracy for the jth class is defined as follows:

$$P_j = \frac{TP_j}{TP_j + FP_j}, \tag{43}$$

where TP_j is the number of true-positive classified samples, that is, the samples that are correctly classified as belonging to the jth class. FP_j is the number of false-positive classified samples, that is, the samples that are incorrectly classified as belonging to the jth class. The metric P_{avg} is defined as the average classification accuracy over all p classes, that is,

$$P_{avg} = \frac{1}{p}\sum_{j=1}^{p} P_j. \tag{44}$$

The metric $P_{\min}$ refers to the classification accuracy defined in (43) for the minority class with the minimum size. To measure the capability of EMDO to separate any pair of classes, the area under curve (AUC) [54,55] is widely used in [56,57]. Denote $A_{m,n}$ as the AUC between class m and class n. The metric AUCm is defined as follows for measuring the capability of EMDO to separate the smallest minority class with the minimum size from the other classes.

$$\text{AUCm} = \frac{1}{p-1}\sum_{n \neq n'} \frac{A_{n,n'} + A_{n',n}}{2} \tag{45}$$

where n' denotes the minority class with the minimum size. In addition to AUCm, the average of AUC over all pairs of classes for a multi-class problem, denoted as MAUC, is defined as

$$\text{MAUC} = \frac{2}{p(p-1)} \sum_{m<n} \frac{A_{m,n} + A_{n,m}}{2}. \tag{46}$$

In order to evaluate the imbalance condition of every data set, the maximum imbalance ratio IRmax is defined as follows:

$$IR_{\max} = \frac{N_{\max}}{N_{\min}} \tag{47}$$

where $N_{\min} = \min_{j=1\ldots p}(N_j)$, $N_{\max} = \max_{j=1\ldots p}(N_j)$.

Example 1:

The data sets used in the simulation are the same as those used in [43] for comparing the performance of the EMDO against AMDO and other learning algorithms. The data sets used in [43] were mainly from data repositories such as the ones Knowledge Extraction based on Evolutionary Learning (KEEL) [58] and UCI (University of California, Irvine) Machine Learning Repository [59]. Table 1 describes these data sets. The performance comparison based on different indices are made in Tables 2–5. Within Tables 2–5, the algorithms such as SSMOTE refers to Static-SMOTE [60], GCS refers to RESCALE [61], ABNC refers to AdaBoost.NC [62], and OSMOTE refers to OVOSMOTE [63]. The MDO in [42], MDO+ and AMDO in [43] are also compared in Tables 2–5 with the proposed EMDO. The Baseline algorithm is the classifier C4.5 without any oversampling technique.

Table 2. Comparison of $P_{\min}$ (%) for every over sampling scheme on different data sets.

Data Set	Baseline	SSMOTE	GCS	ABNC	OSMOTE	MDO	MDO+	AMDO	EMDO
Balance	$0.00_{0.00}$	$8.44_{9.18}$	$6.44_{9.83}$	$2.22_{4.97}$	$12.44_{13.41}$	2.004.47	$0.00_{0.00}$	$10.22_{0.50}$	$\mathbf{20.41_{10.95}}$
Hayes-Roth	$\mathbf{100.00_{0.00}}$	$\mathbf{100.00_{0.00}}$	$\mathbf{100.00_{0.00}}$	$\mathbf{100.00_{0.00}}$	$\mathbf{100.00_{0.00}}$	$\mathbf{100.00_{0.00}}$	$\mathbf{100.00_{0.00}}$	$\mathbf{100.00_{0.00}}$	$\mathbf{100.00_{0.00}}$
New-Thyroid	$83.33_{11.79}$	$90.00_{14.91}$	$93.33_{9.13}$	$93.33_{9.13}$	$90.00_{9.13}$	$83.33_{16.67}$	$96.67_{7.45}$	$\mathbf{100.00_{0.00}}$	$\mathbf{100.00_{0.00}}$
Page-Blocks	$82.67_{11.88}$	$78.67_{6.91}$	$93.33_{14.91}$	$72.67_{24.99}$	$93.33_{9.13}$	$75.33_{16.26}$	$93.33_{9.13}$	$\mathbf{96.67_{7.45}}$	$\mathbf{96.67_{5.33}}$
Dermatology	$95.00_{11.18}$	$95.00_{11.18}$	$90.00_{13.69}$	$\mathbf{100.00_{0.00}}$	$90.00_{13.69}$	$95.00_{11.18}$	$95.00_{11.18}$	$\mathbf{100.00_{0.00}}$	$\mathbf{100.00_{0.00}}$
Breast-Tissue	$60.00_{27.89}$	$40.00_{36.51}$	$46.67_{29.81}$	$60.00_{27.89}$	$53.33_{29.81}$	$60.00_{27.89}$	$53.33_{29.81}$	$60.00_{27.89}$	$\mathbf{73.36_{13.32}}$
UKM	$88.00_{13.04}$	$92.00_{13.04}$	$90.00_{10.00}$	$94.00_{8.94}$	$88.00_{16.43}$	$86.00_{13.42}$	$86.00_{13.42}$	$94.00_{8.94}$	$\mathbf{96.00_{5.76}}$
Vertebral-Column	$65.00_{16.03}$	$65.00_{19.00}$	$60.00_{19.90}$	$61.67_{18.26}$	$66.67_{5.89}$	$68.33_{22.36}$	$66.67_{28.87}$	$86.67_{9.50}$	$\mathbf{90.67_{6.02}}$
Ecoli	$65.00_{37.91}$	$70.00_{27.39}$	$55.00_{32.60}$	$70.00_{27.39}$	$55.00_{32.60}$	$75.00_{30.62}$	$90.00_{13.69}$	$\mathbf{90.00_{13.69}}$	$\mathbf{90.00_{12.40}}$
Average	71.00	71.01	70.53	72.65	72.09	71.67	75.67	81.95	**85.23**
Rank Avg.	6.33	5.89	6.39	5.11	5.78	5.89	5.28	2.56	**1.78**

The best result is in bold face.

Table 3. Comparison of P_{avg} (%) for every oversampling scheme on different data sets.

Data Set	Baseline	SSMOTE	GCS	ABNC	OSMOTE	MDO	MDO+	AMDO	EMDO
Balance	$56.38_{1.75}$	$58.38_{2.94}$	$56.65_{4.03}$	$62.33_{3.29}$	$57.86_{2.78}$	$57.28_{2.92}$	$55.45_{1.66}$	$60.37_{2.06}$	$\mathbf{64.38_{3.59}}$
Hayes-Roth	$84.91_{6.71}$	$84.67_{5.06}$	$85.33_{5.58}$	$83.52_{7.08}$	$84.97_{6.74}$	$84.91_{6.71}$	$84.97_{6.74}$	$84.97_{6.74}$	$\mathbf{85.61_{2.39}}$
New-Thyroid	$88.86_{6.02}$	$91.08_{3.02}$	$93.14_{4.55}$	$93.59_{1.45}$	$92.54_{6.61}$	$89.81_{6.84}$	$94.98_{3.59}$	$96.54_{2.99}$	$\mathbf{96.60_{2.05}}$
Page-Blocks	$84.30_{2.14}$	$84.57_{1.88}$	$88.77_{4.49}$	$79.70_{5.57}$	$89.61_{2.98}$	$81.24_{1.79}$	$86.13_{2.48}$	$88.77_{1.92}$	$\mathbf{90.15_{1.87}}$
Dermatology	$95.67_{2.05}$	$95.73_{1.83}$	$93.50_{2.71}$	$97.10_{0.75}$	$95.31_{2.45}$	$95.67_{2.05}$	$96.06_{1.62}$	$96.88_{0.26}$	$\mathbf{97.13_{0.20}}$
Breast-Tissue	$63.22_{3.74}$	$60.89_{3.94}$	$68.78_{5.77}$	$66.00_{4.88}$	$65.83_{7.05}$	$63.22_{3.74}$	$66.56_{2.17}$	$63.22_{3.74}$	$\mathbf{70.60_{6.28}}$
UKM	$92.18_{2.02}$	$92.57_{4.85}$	$91.03_{2.00}$	$94.49_{2.45}$	$91.78_{2.32}$	$91.45_{2.48}$	$91.92_{2.50}$	$94.23_{2.14}$	$\mathbf{95.24_{1.37}}$
Vertebral-Column	$76.44_{2.65}$	$77.22_{4.66}$	$75.67_{5.86}$	$76.67_{3.12}$	$77.00_{4.71}$	$78.22_{5.55}$	$76.56_{7.41}$	$81.89_{2.38}$	$\mathbf{85.77_{1.53}}$
Ecoli	$74.64_{7.88}$	$72.81_{12.01}$	$72.73_{11.35}$	$76.23_{7.14}$	$73.12_{8.72}$	$77.24_{7.03}$	$82.30_{5.21}$	$82.44_{5.08}$	$\mathbf{85.31_{1.62}}$
Average	79.61	79.77	80.62	81.07	80.89	79.89	81.66	83.26	**85.65**
Rank Avg.	7.00	6.11	6.17	4.78	5.44	6.33	4.89	3.28	**1.00**

The best result is in bold face.

Table 4. Comparison of AUCm (%) for every oversampling scheme on different data sets.

Data Set	Base	SSMOTE	GCS	ABNC	OSMOTE	MDO	MDO+	AMDO	EMDO
Balance	$56.95_{1.91}$	$58.12_{3.27}$	$57.10_{3.43}$	$60.52_{2.89}$	$58.58_{3.18}$	$57.40_{2.78}$	$56.60_{0.88}$	$60.61_{1.24}$	$\mathbf{65.61_{4.30}}$
Hayes-Roth	$94.34_{2.52}$	$94.25_{1.90}$	$94.50_{2.09}$	$93.82_{2.65}$	$94.36_{2.53}$	$94.34_{2.52}$	$94.36_{2.53}$	$94.36_{2.53}$	$\mathbf{94.67_{2.81}}$
New-Thyroid	$91.40_{5.33}$	$93.90_{4.45}$	$95.43_{3.76}$	$95.76_{2.37}$	$94.04_{5.11}$	$91.85_{6.74}$	$97.04_{2.94}$	$98.37_{1.41}$	$\mathbf{98.55_{1.23}}$
Page-Blocks	$90.75_{3.57}$	$89.84_{2.16}$	$94.81_{4.66}$	$86.82_{7.87}$	$94.96_{3.16}$	$87.90_{4.66}$	$93.97_{2.91}$	$95.51_{2.07}$	$\mathbf{96.07_{2.24}}$
Dermatology	$97.32_{3.46}$	$97.39_{3.27}$	$95.55_{4.17}$	$99.05_{0.28}$	$96.09_{4.09}$	$97.32_{3.46}$	$97.48_{3.23}$	$98.98_{0.26}$	$\mathbf{99.06_{0.22}}$
Breast-Tissue	$76.80_{7.31}$	$72.18_{10.10}$	$75.80_{9.99}$	$78.30_{8.92}$	$76.92_{9.75}$	$76.80_{7.31}$	$77.30_{8.09}$	$76.80_{7.31}$	$\mathbf{82.37_{3.77}}$
UKM	$94.14_{3.65}$	$94.94_{5.27}$	$94.19_{3.04}$	$96.41_{2.61}$	$94.01_{4.64}$	93.333.88	$93.55_{3.92}$	$96.45_{3.02}$	$\mathbf{97.26_{2.49}}$
Vertebral-Column	$79.25_{3.17}$	$79.96_{5.56}$	$78.38_{6.36}$	$79.42_{5.04}$	$80.04_{2.48}$	$81.13_{6.76}$	$79.54_{8.76}$	$86.04_{1.72}$	$\mathbf{89.07_{2.00}}$
Ecoli	$83.07_{11.54}$	$83.43_{10.88}$	$79.78_{11.44}$	$84.78_{9.21}$	$79.74_{11.00}$	$86.38_{9.56}$	$91.97_{4.61}$	$91.59_{4.19}$	$\mathbf{92.91_{2.92}}$
Average	84.89	84.89	85.06	86.10	85.42	85.16	86.87	88.75	**90.62**
Rank Avg.	7.00	6.22	6.33	4.89	5.44	6.33	4.89	2.89	**1.00**

The best result is in bold face.

Table 5. Comparison of MAUC (%) for every over sampling scheme on different data sets.

Data Set	Baseline	SSMOTE	GCS	ABNC	OSMOTE	MDO	MDO+	AMDO	EMDO
Balance	$67.29_{1.31}$	$68.79_{2.20}$	$67.49_{3.02}$	$71.75_{2.47}$	$68.39_{2.09}$	$67.96_{2.19}$	$66.59_{1.25}$	$70.27_{1.54}$	$\mathbf{73.13_{2.77}}$
Hayes-Roth	$88.68_{5.03}$	$88.50_{3.79}$	$89.00_{4.18}$	$87.64_{5.31}$	$88.73_{5.06}$	$88.68_{5.03}$	$88.73_{5.06}$	$88.73_{5.06}$	$\mathbf{89.15_{4.86}}$
New-Thyroid	$91.64_{4.52}$	$93.31_{2.27}$	$94.86_{3.41}$	$95.19_{1.09}$	$94.40_{4.96}$	$92.36_{5.13}$	$96.24_{2.69}$	$97.40_{2.24}$	$\mathbf{97.76_{2.14}}$
Page-Blocks	$90.19_{1.34}$	$90.35_{1.18}$	$92.98_{2.80}$	$87.31_{3.48}$	$93.51_{1.86}$	$88.27_{1.12}$	$91.33_{1.55}$	$92.98_{1.20}$	$\mathbf{93.85_{1.17}}$
Dermatology	$97.40_{1.23}$	$97.44_{1.10}$	$96.10_{1.62}$	$\mathbf{98.26_{0.45}}$	$97.19_{1.47}$	$97.40_{1.23}$	$97.63_{0.97}$	$98.13_{0.16}$	$\mathbf{98.26_{0.16}}$
Breast-Tissue	$77.93_{2.24}$	$76.53_{2.36}$	$81.27_{3.46}$	$79.60_{2.93}$	$79.50_{4.23}$	$77.93_{2.24}$	$79.93_{1.30}$	$77.93_{2.24}$	$\mathbf{84.07_{5.16}}$
UKM	$94.79_{1.34}$	$95.05_{3.23}$	$94.02_{1.33}$	$96.32_{1.64}$	$94.52_{1.54}$	$94.30_{1.65}$	$94.61_{1.67}$	$96.15_{1.42}$	$\mathbf{96.91_{1.05}}$
Vertebral-Column	$82.33_{1.99}$	$82.92_{3.50}$	$81.75_{4.39}$	$82.50_{2.34}$	$82.75_{3.53}$	$83.67_{4.16}$	$82.42_{5.55}$	$86.42_{1.78}$	$\mathbf{89.05_{1.33}}$
Ecoli	$84.15_{4.93}$	$83.01_{7.50}$	$82.96_{7.10}$	$85.15_{4.46}$	$83.20_{5.45}$	$85.77_{4.39}$	$88.93_{3.26}$	$89.03_{3.18}$	$\mathbf{90.82_{1.01}}$
Average	86.04	86.21	86.71	87.08	86.91	86.26	87.38	88.56	**90.33**
Rank Avg.	7.00	6.11	6.17	4.72	5.44	6.33	4.89	3.28	**1.06**

The best result is in bold face.

To compare the performance of EMDO with those of the other schemes, the rank average of every scheme was calculated. All oversampling schemes were tested on each of the data sets listed in Table 1. The ranking of algorithm performance was based on each of the metrics. For instance, the algorithm with the best performance is ranked first, the algorithm with the second-to-best performance is ranked second, etc. The average rank of every algorithm is then calculated. The schemes with the same metric values share ranks. For instance, if two schemes are ranked second because they have the same metric values, the two schemes share the second and third ranks. These two schemes are thus ranked 2.5. The means and standard deviations of $P_{\min}$ and P_{avg} are listed in Tables 2 and 3, respectively, for every oversampling scheme, including the proposed EMDO, applied to different data sets. According to Tables 2 and 3, EMDO outperforms all of the other schemes on every data set. EMDO has the lowest average rank. The results shown in both Tables 2 and 3 imply that the oversampling performed using EMDO significantly improves the classification accuracy for the smallest minority class. Moreover, the synthetic samples generated for the minority class samples improve the overall average classification accuracy.

For all schemes, the mean and standard deviation are listed in Table 4 and the AUCm defined in (45) and MAUCm defined in (46) are also compared in Tables 4 and 5, respectively. As indicated in Table 4, EMDO outperform the other schemes in separating the smallest minority class from the other classes for every listed data set. Moreover, according to Table 5, the capability of EMDO to separate all pairs of classes in the multi-class problem is superior to that of the other schemes.

Example 2:

The performance of EMDO is evaluated on the sensory data in this example. Two data sets, Statlog (Shuttle) from UCI and Mafalda [64] from Github are utilized in this example. The data set Statlog (Shuttle) is the set of the recorded sensory data from NASA's space

shuttle while the data set Mafalda is the set of the recorded sensory data from different brands of cars. The characteristics of these two data sets is shown in Table 6. Table 6 shows that both data sets are extremely imbalanced because the maximum imbalance ratios IR_{max} are as high as 5684.7 and 5.94, respectively. Four indices P_{min}, P_{avg}, AUCm, and MAUC are calculated and compared in Table 7 for both data sets with classifier C4.5. The classification results are greatly improved with oversampling scheme EMDO compared with the results without EMDO. EMDO helps improve classification results for both highly imbalanced data sets according to the four evaluation indices listed in Table 7.

Table 6. Characteristics of the data sets for simulations in Example 2.

Data Set	Size	Attributes	Classes	Class Distribution	IR_{max}	No. of Ellipsoids
Statlog (Shuttle)	58000	9	7	45586/50/171/8903/3267/10/13	5684.67	NA/3/4/5/5/3/1
Mafalda	23762	14	3	17757/2990/3015	5.94	NA/11/11

NA: not available.

Table 7. Comparison of the performance with and without EMDO for imbalanced sensory data.

Data Set		P_{min}	P_{avg}	AUCm	MAUC
Statlog (Shuttle)	w/ EMDO	$89.33_{13.73}$	$96.60_{1.95}$	$99.66_{0.47}$	$99.32_{0.94}$
	w/o EMDO	$60_{48.99}$	$93.21_{7.42}$	$77.65_{6.17}$	$92.80_{1.66}$
Mafalda	w/ EMDO	$57.38_{18.33}$	$72.34_{9.33}$	$76.39_{9.31}$	$78.19_{8.22}$
	w/o EMDO	$33.17_{12.13}$	$60.77_{10.53}$	$65.25_{7.88}$	$67.80_{6.99}$

7. Conclusions

EMDO was demonstrated to outperform competing oversampling approaches in simulations. EMDO performed well because it approximates the decision region of the target minority class with reasonable accuracy by using a set of ellipsoids. In problems involving multi-class imbalanced data, EMDO performs exceptionally well if the decision region of the minority class is separated in the feature space. EMDO can learn the sizes, centers, and orientations of the ellipsoids that approximate the minority class decision region by using the underlying distribution of minority class samples. IoT is a key emerging technology, and imbalanced data will become an increasingly common problem as the number of IoT sensors increases. The proposed EMDO is suitable for solving such multi-class imbalanced data classification problems. One of the future works related to this study involves applying EMDO to address the problem of imbalanced data encountered in real-world IoT sensing data. Although EMDO is a data-level learning approach, it can easily be integrated with other cost-sensitive methods to increase the effectiveness and efficiency of learning. Further studies on variants of integration can be another direction for future research.

Author Contributions: Conceptualization, L.Y.; methodology, L.Y.; software, T.-B.L.; validation, T.-B.L.; writing—original draft preparation, T.-B.L.; writing—review and editing, L.Y.; supervision, L.Y.; project administration, L.Y.; funding acquisition, L.Y.; Both authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Ministry of Science and Technology, Taiwan, grant number MOST 110-2221-E-027-054-MY3.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, Y.; Zhong, X.; Ma, Z.; Liu, H. The outlier and integrity detection of rail profile based on profile registration. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 1074–1085. [CrossRef]

2. Kang, S.; Sristi, S.; Karachiwala, J.; Hu, Y.-C. Detection of anomaly in train speed for intelligent railway systems. In Proceedings of the 2018 International Conference on Control, Automation and Diagnosis (ICCAD), Marrakech, Morocco, 19–21 March 2018; pp. 1–6.
3. Wang, H. Unsupervised anomaly detection in railway catenary condition monitor4ing using auto-encoders. In Proceedings of the IECON 2020 the 46th Annual Conference of the IEEE Industrial Electronics Society, Singapore, 18–21 October 2020; pp. 2636–2641.
4. Qian, G.; Lu, S.; Pan, D.; Tang, H.; Liu, Y.; Wang, Q. Edge computing: A promising framework for real-time fault diagnosis and dynamic control of rotating machines using multi-sensor data. *IEEE Sens. J.* **2019**, *19*, 4211–4220. [CrossRef]
5. Maruthi, G.S.; Hegde, V. Application of MEMS accelerometer for detection and diagnosis of multiple faults in roller element bearings of three phase induction motor. *IEEE Sens. J.* **2016**, *16*, 145–152. [CrossRef]
6. Tong, Z.Y.; Dong, Z.Y.; Li, M. A new entropy bi-cepstrum based method for DC motor brush abnormality recognition. *IEEE Sens. J.* **2017**, *17*, 745–754. [CrossRef]
7. Kim, E.; Cho, S.; Lee, B.; Cho, M. Fault detection and diagnosis using self-attentive convolutional neural networks for variable-length sensor data in semiconductor manufacturing. *IEEE Trans. Semicond. Manuf.* **2019**, *32*, 302–309. [CrossRef]
8. Azamfar, M.; Li, X.; Lee, J. Deep learning-based domain adaptation method for fault diagnosis in semiconductor manufacturing. *IEEE Trans. Semicond. Manuf.* **2020**, *33*, 445–453. [CrossRef]
9. Ghosh, A.; Qin, S.; Lee, J.; Wang, G. FBMTP: An automated fault and behavioral anomaly detection and isolation tool for PLC-controlled manufacturing systems. *IEEE Trans. Syst. Man Cyber. Syst.* **2017**, *47*, 3397–3417. [CrossRef]
10. Quang, X.; Huo, H.; Xia, L.; Shan, F.; Liu, J.; Mo, Z.; Yan, F.; Ding, Z.; Yang, Q.; Song, B.; et al. Dual-sampling attention network for diagnosis of COVID-19 from community acquired pneumonia. *IEEE Trans. Med. Imaging* **2020**, *39*, 2595–2605.
11. Liu, N.; Li, E.; Qi, M.; Xu, L.; Gao, B. A novel ensemble learning paradigm for medical diagnosis with imbalanced data. *IEEE Access* **2020**, *8*, 171263–171280. [CrossRef]
12. Huda, S.; Yearwood, J.; Jelinek, H.F.; Hassan, M.M.; Fortino, G.; Buckland, M. A hybrid feature selection with ensemble classification for imbalanced healthcare data: A case study for brain tumor diagnosis. *IEEE Access* **2016**, *4*, 9145–9154. [CrossRef]
13. He, H.; Garcia, E.A. Learning from imbalanced data. *IEEE Trans. Knowl. Data Eng.* **2009**, *21*, 1263–1284. [CrossRef]
14. Batista, G.E.; Prati, R.C.; Monard, M.C. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explor. Newsl.* **2004**, *6*, 20–29. [CrossRef]
15. Krawczyk, B. Learning from imbalanced data: Open challenges and future directions. *Prog. Artif. Intell.* **2016**, *5*, 221–232. [CrossRef]
16. Guo, H.-X.; Li, Y.-J.; Shang, J.; Gu, M.-Y.; Huang, Y.-Y. Learning from class-imbalanced data: Review of methods and applications. *Expert Syst. Appl.* **2017**, *73*, 220–239.
17. Wu, G.; Chang, E.Y. KBA: Kernel boundary alignment considering imbalanced data classification. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 786–795. [CrossRef]
18. Ohsaki, M.; Wang, P.; Matsuda, K.; Katagiri, S.; Watanabe, H.; Ralescu, A. Confusion-matrix-based kernel logistic regression for imbalanced data classification. *IEEE Trans. Knowl. Data Eng.* **2017**, *29*, 1806–1819. [CrossRef]
19. Manevitz, L.M.; Yousef, M. One-class SVMs for document classification. *J. Mach. Learn. Res.* **2002**, *2*, 139–154.
20. Raskutti, B.; Kowalczyk, A. Extreme rebalancing for SVMs:a case study. *ACM SIGKDD Explor. Newsl.* **2004**, *6*, 60–69. [CrossRef]
21. Khan, S.H.; Hayat, M.; Bennamoun, M.; Sohel, F.A.; Togneri, R. Cost-sensitive learning of deep feature representations form imbalanced data. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 3573–3587. [PubMed]
22. Huang, C.; Li, Y.; Loy, C.C.; Tang, X. Learning deep representation for imbalanced classification. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 5375–5384.
23. Ng, W.W.Y.; Hu, J.; Yeung, D.S.; Yin, S.; Roli, F. Diversified sensitivity-based undersampling for imbalance classification problems. *IEEE Trans. Cybern.* **2015**, *45*, 2402–2412. [CrossRef] [PubMed]
24. Tang, Y.; Zhang, Y.Q.; Chawla, N.V.; Krasser, S. SVMs modeling for highly imbalanced classification. *IEEE Trans. Syst. Man Cybern. Part B* **2009**, *39*, 281–288. [CrossRef] [PubMed]
25. Liu, X.Y.; Wu, J.; Zhou, Z.H. Exploratory undersampling for class-imbalance learning. *IEEE Trans. Syst. Man Cybern. Part B* **2009**, *39*, 539–550.
26. Kang, Q.; Shi, L.; Zhou, M.; Wang, X.; Wu, Q.; Wei, Z. A distance-based weighted undersampling scheme for support vector machines and its application to imbalanced classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 4152–4165. [CrossRef]
27. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *26*, 321–357. [CrossRef]
28. He, H.; Bai, Y.; Garcia, E.A.; Li, S. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In Proceedings of the 2008 IEEE International Joint Conference on Neural Networks, Hong Kong, China, 1–8 June 2008; pp. 1322–1328.
29. Chawla, N.V.; Lazarevic, A.; Hall, L.O.; Bowyer, K.W. SMOTEBoost: Improving prediction of the minority class in boosting. In Proceedings of the Knowledge Discovery in Databases: PKDD (Lecture Notes in Computer Science), Cavtat-Dubrovnik, Croatia, 22–26 September 2003; Springer: Berlin/Heidelberg, Germany, 2003; pp. 107–119.
30. Han, H.; Wang, W.-Y.; Mao, B.-H. Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In Proceedings of the International Conference on Intelligent Computing, Hefei, China, 23–26 August 2005; pp. 878–887.

31. Xie, Z.; Jiang, L.; Ye, T.; Li, X. A synthetic minority oversampling method based on local densities in low-dimensional space for imbalanced learning. International Conference on Database Systems for Advanced Applications, Taipei, Taiwan, 20–23 April 2015; pp. 3–18.
32. Das, B.; Krishnan, N.C.; Cook, D.J. RACOG and wRACOG: Two probabilistic oversampling techniques. *IEEE Trans. Knowl. Data Eng.* **2015**, *27*, 222–234. [CrossRef]
33. Pérez-Ortiz, M.; Gutiérrez, P.A.; Hervás-Martínez, C.; Yao, X. Graph-based approaches for over-sampling in the context of ordinal regression. *IEEE Trans. Knowl. Data Eng.* **2015**, *27*, 1233–1245. [CrossRef]
34. Schapire, R.E. The boosting approach to machine learning: An overview. In *Nonlinear Estimation Classification*; Springer: New York, NY, USA, 2003; pp. 149–171.
35. Breiman, L. Bagging predictors. *Mach. Learn.* **1996**, *24*, 123–140. [CrossRef]
36. Polikar, R. Ensemble learning. In *Ensemble Machine Learning*; Springer: Berlin, Germany, 2012; pp. 1–34.
37. Moniz, N.; Ribeiro, R.P.; Cerqueira, V.; Chawla, N. SMOTEBoost for regression: Improving the prediction of extreme values. In Proceedings of the 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA), Turin, Italy, 1–3 October 2018; pp. 150–159.
38. Freund, Y.; Schapire, R.E. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [CrossRef]
39. Guo, H.; Viktor, H.L. Learning form imbalanced data sets with boosting and data generation: The Databoost-IM approach. *ACM SIGKDD Explor. Newsl.* **2004**, *6*, 30–39. [CrossRef]
40. Khoshgoftaar, T.M.; Hulse, J.V.; Napolitano, A. Comparing boosting and bagging techniques with noisy and imbalanced data. *IEEE Trans. Syst. Man Cybern. Part A* **2011**, *41*, 552–568. [CrossRef]
41. Galar, M.; Fernandez, A.; Barrenechea, E.; Bustince, H.; Herrera, F. A review on ensembles for the class imbalanced problem: Bagging, boosting, and hybrid-based approaches. *IEEE Trans. Syst. Man Cybern. Part C* **2012**, *42*, 473–484. [CrossRef]
42. Abdi, L.; Hashemi, S. To combat multi-class imbalanced problems by means of over-sampling techniques. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 238–251. [CrossRef]
43. Yang, X.; Kuang, Q.; Zhang, W.; Zhang, G. AMDO: An over-sampling technique for multi-class imbalanced problems. *IEEE Trans. Knowl. Data Eng.* **2018**, *30*, 1672–1685. [CrossRef]
44. Gustafson, D.E.; Kessel, W.C. Fuzzy clustering with a fuzzy covariance matrix. In Proceedings of the 1978 IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes, San Diego, CA, USA, 10–12 January 1979; pp. 761–766.
45. Bezdek, J. *Pattern Recognition with Fuzzy Objective Function*; Plenum Press: New York, NY, USA, 1981.
46. Yao, L.; Weng, K.-S. Imputation of incomplete data using adaptive ellipsoids with liner regression. *J. Intell. Fuzzy Syst.* **2015**, *29*, 253–265. [CrossRef]
47. Yao, L.; Weng, K.-S.; Wu, M.S. Evolutionary learning of classifiers for disc discrimination. *IEEE/ASME Trans. Mechatron.* **2015**, *20*, 3194–3203. [CrossRef]
48. Reyes-Sierra, M.; Coello, C.A.C. Multi-objective particle swarm optimizers: A survey of the state-of-the art. *Int. J. Comput. Intell. Res.* **2006**, *2*, 287–308.
49. Hu, W.; Yen, G.G. Adaptive multi-objective particle swarm optimization based on parallel cell coordinate system. *IEEE Trans. Evol. Comput.* **2015**, *19*, 1–18.
50. Chen, V.C.P.; Ruppert, D.; Shoemaker, C.A. Applying experimental design and regression splines to high-dimensional continuous-state stochastic dynamic programming. *Oper. Res.* **1999**, *47*, 38–53. [CrossRef]
51. Liu, F.; Ju, X.; Wang, N.; Wang, L.; Lee, W.-J. Wind farm macro-siting optimization with insightful bi-criteria identification and relocation mechanism in genetic algorithm. *Energy Convers. Manag.* **2020**, *217*, 112964. [CrossRef]
52. Ahmed, W.; Hanif, A.; Kallu, K.D.; Kouzani, A.Z.; Ali, M.U.; Zafar, A. Photovoltaic panels classification using isolated and transfer learned deep neural models using infrared thermographic images. *Sensors* **2021**, *21*, 5668. [CrossRef]
53. Knowles, J.D.; Corne, D.W. Approximating the Nondominated Front Using the Pareto Archived Evolution Strateg. *Evol. Comput.* **2000**, *8*, 149–172. [CrossRef] [PubMed]
54. Bradley, A.P. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recog.* **1997**, *30*, 1145–1159. [CrossRef]
55. Tang, K.; Wang, R.; Chen, T. Towards maximizing the area under the ROC curve for multi-class classification problems. In Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 7–11 August 2011; pp. 483–488.
56. Ferri, C.; Hernandez-Navarro, J.; Modroiu, R. An experimental comparison of performance measures for classification. *Pattern Recognit. Lett.* **2009**, *30*, 27–38. [CrossRef]
57. Loyola-Gonzalez, O.; Martınez-Trinidad, J.F.; Carrasco-Ochoa, J.A.; Garcıa-Borroto, M. Study of the impact of resampling methods for contrast pattern based classifiers in imbalanced databases. *Neurocomputing* **2016**, *175*, 935–947. [CrossRef]
58. Alcalá, J.; Fernández, A.; Luengo, J.; Derrac, J.; García, S.; Sánchez, L.; Herrera, F. KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *J. Mult.-Valued Logic Soft. Comput.* **2010**, *17*, 255–287.
59. Frank, A.; Asuncion, A. UCI machine learning repository. 2010. Available online: http://archive.ics.uci.edu/ml (accessed on 5 March 2020).

60. Fernández-Navarro, F.; Hervás-Martínez, C.; Gutiérrez, P.A. A dynamic over-sampling procedure based on sensitivity for multi-class problems. *Pattern Recog.* **2011**, *44*, 1821–1833. [CrossRef]
61. Zhou, Z.-H.; Liu, X.-Y. On multi-class cost-sensitive learning. *Comput. Intell.* **2010**, *26*, 232–257. [CrossRef]
62. Wang, S.; Yao, X. Multi-class imbalance problems: Analysis and potential solutions. *IEEE Trans. Syst. Man Cybern. B* **2012**, *42*, 1119–1130. [CrossRef] [PubMed]
63. Fernández, A.; López, V.; Galar, M.; del Jesus, M.J.; Herrera, F. Analysing the classification of imbalacned data-sets with multiple classes: Binarization techniques and ad-hoc approaches. *Knowl.-Based Syst.* **2013**, *42*, 97–110. [CrossRef]
64. Malfada. 13 December 2017. Available online: https://github.com/sisinflab-swot/mafalda (accessed on 3 September 2021).