Applied Soft Computing 143 (2023) 110415

Contents lists available at ScienceDirect

Applied Soft Computing

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

Review article

A broad review on class imbalance learning techniques

Salim Rezvani [a,*], Xizhao Wang [b]

[a] Department of Computer Science, Toronto Metropolitan University, Toronto, Canada
[b] Big Data Institute, College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, 518060, China

ARTICLE INFO

Article history:
Received 21 January 2022
Received in revised form 24 February 2023
Accepted 8 May 2023
Available online 26 May 2023

Keywords:
Algorithmic structures techniques
Data pre-processing techniques
Hybrid techniques
Imbalanced learning
Support vector machine

ABSTRACT

The imbalanced learning issue is related to the performance of learning algorithms in the presence of asymmetrical class distribution. Due to the complex characteristics of imbalanced datasets, learning from such data need new algorithms and understandings to convert efficient large amounts of initial data into suitable datasets. Although several review papers can be found about imbalanced classification problems, none of them contributed an in-depth review of SVM for imbalanced classification problems. To fill this gap, we present an exhaustive review of existing methods to deal with issues linked with class imbalance learning. The majority of the existing survey addresses only classification tasks. We also describe methods to deal with similar problems in regression tasks. A new taxonomy for class imbalanced learning techniques is proposed and classified into three parts: (1) Data pre-processing, (2) Algorithmic structures, and (3) Hybrid techniques. The advantages and disadvantages of each type of imbalanced learning technique are discussed. Moreover, we explain the main difficulties in distributions of imbalanced datasets and discuss the main approaches that have been proposed to tackle these issues. Finally, to stimulate the next research in this area, we emphasize the main opportunities and challenges, which can be useful in research directions for learning algorithms from imbalanced data.

© 2023 Elsevier B.V. All rights reserved.

Contents

* Corresponding author.
E-mail address: salim.rezvani@torontomu.ca (S. Rezvani).

https://doi.org/10.1016/j.asoc.2023.110415
1568-4946/© 2023 Elsevier B.V. All rights reserved.

1. Introduction

In recent years, the classification of imbalanced datasets is one of the main problems for machine learning techniques. An imbalanced dataset means the number of the majority class is much more than the minority class. Fig. 1 is two dimensions example with points representing examples and different colors of the points representing the class and coordinates are features. As you can see, the number of minority samples (blue class) are much less than the majority samples (red class). With the ongoing extension of data availability in complex, large-scale, and networked systems, such as security, finance, and the internet, it is essential to promote the basic understanding of knowledge in this area.

Even though existing techniques demonstrated great success in many real-world applications, the issue of learning in imbalanced data is justly new. Therefore, this problem needs more attention from researchers.

Several survey papers related to the imbalanced learning area have been published during the past few years. In [1], the authors provided a review of the SVM weakness to handle the imbalanced datasets and described why the traditional approaches of under-sampling are not the best choice for that. Another survey about class imbalance learning methods for SVM was collected in [2]. The purpose of that survey is to review different data pre-processing and algorithmic methods to improve the performance of SVM in learning from imbalanced datasets. The critical issue in the knowledge discovery and data engineering domain is imbalanced learning [3,4]. Their concentration was to present a challenging review of the nature of the issue, existing evaluation metrics, and state-of-the-art technologies that utilized to assess learning performance under the imbalanced learning scenario. In fact, they used some significant assessments as a perfect reference for present and next knowledge discovery.

In [5], the author proposed a taxonomy for ensemble-based techniques to deal with class imbalanced learning. Each method is classified based on the specific ensemble methodology. Moreover, they developed a perfect experimental comparison by the attention of the main published techniques in the ensemble area.

A review paper of predictive modeling for the imbalanced datasets provided in [6,7], which evaluates existing methods to deal with basic applications of predictive analysis. They provided a categorization for the available techniques and classified them into four parts: data pre-processing techniques, special-purpose learning techniques, prediction post-processing techniques, and hybrid techniques. Instead of talking about available techniques, [8] proposed some open problems and challenges that are required to attend for the future study of imbalanced learning. This survey covered some techniques from imbalanced datasets, like clustering, classification, regression, and big data analytics in social media and computer vision.

Some different challenges and examples to deal with the imbalanced datasets are considered in [9]. Also, they discussed the benefits and limitations of each technique and shown some examples of imbalanced datasets. Recently, [10–12] conducted a review of re-sampling techniques and evaluated those techniques for five imbalanced datasets.

Multi-label classification is one of the main challenges of imbalanced data sets that can be considered with three perspectives: imbalance within labels, among labels, and label-sets. In [13], the authors provided a review of some techniques for handling imbalance problems in multi-label data.

However, an in-depth investigation of available tasks about imbalanced datasets will help us to promote our knowledge of machine learning [13–28]. Therefore, we prepared Table 1, which is about the last recently published imbalanced survey papers with the description of the title, journal name, and the subject of contexts.

The idea of prior data information and local neighborhood information proposed in [30] inspires the K-nearest neighbor-based weighted SVM to solve the problem of imbalanced datasets. Local neighborhood information is incorporated via the weight matrix in the objective function. In the proposed technique, they used weight vectors in the corresponding constraints of the objective functions to exploit the interclass information. Also, over-sampling and under-sampling approaches are followed to balance the data in class imbalance problems. One Class SVM is a popular technique for unsupervised anomaly detection. In [31], the authors check out several rule extraction techniques over OneClass SVM models, while presenting alternative designs for some of those algorithms. The first goal was analyzing the quality of the rules extracted. The authors evaluated the suggested technique over different datasets, both from public sources and from Telefónica's, using communications. The results show that clustering-based techniques (K-Means clustering) yield results that are similar to each other. To track employee computers, SmartRadar software was developed to use behavior and detect anomalous behavior [32]. They defined anomalous behavior as computer-based activities or processes carried out during work time which are not related to the tasks for which the employee is responsible. Clicking, mouse wheel scrolling, copying and other similar actions by the user are processed, a summary of the data is developed, and a multi-dimensional dataset is created. The authors used support vector machines to detect anomalous behavior. They have shown the proposed software to detect anomalous computer use behavior by employees with high accuracy. The authors in [33] display an anomaly detection algorithm to establish control functions and to more efficiently analyze test data of a hybrid control unit (HCU). The anomaly detection algorithm automatically recognizes anomalies of control functions from the test data. A data-driven approach using a one-class support vector machine (SVM) applies to make it easy to apply various control functions. The established anomaly detection algorithm shows the feasibility and effectiveness of the proposed algorithm in detecting not only prior-known anomalies but also prior-unknown anomalies.

Fig. 2 describes an estimation of published papers on the imbalanced learning topic from 2009 to 2021 in six journals, which are Information science, Knowledge-based systems, Neural networks, Applied soft computing, Expert systems with application, and Neurocomputing. It is obvious the number of published papers is rising significantly in an imbalanced area. According to the quick development of this area, adaptable evaluation of past and present research in imbalanced learning is necessary for future research.

Many variants of SVM have been offered to deal with imbalanced classification problems. Although several review papers can be found about imbalanced classification problems, none of them contributed an in-depth review of SVM for imbalanced classification problems. To fill this gap, we provide an exhaustive survey for the imbalanced learning techniques. The advantages and disadvantages of each category are highlighted to help scholars in

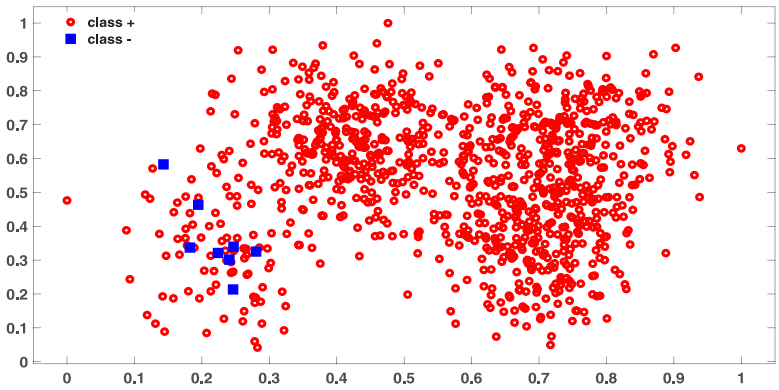

Fig. 1. Imbalanced dataset.

Table 1
Description of the published survey papers on an imbalance area for the last 17 years.

No.	Title	Journal name and year	Description
1	Applying support vector machines to imbalanced datasets [1]	Proceedings of the 15th European Conference, 2004 on Machine Learning	Review the weakness of SVM and why the original under sampling techniques are not the best choice for that problem.
2	Cost-Sensitive Learning vs. Sampling: Which is Best for Handling Unbalanced Classes with Unequal Error Costs? [19]	Proceedings of the International Conference on Data Mining, 2007	Review of cost-sensitive and over-sampling on imbalanced learning.
3	Knowledge discovery from imbalanced and noisy data [15]	Data and Knowledge Engineering, 2009	Investigate on the problem of learning from noisy and imbalanced data.
4	Learning from imbalanced data [4]	IEEE Transactions on Knowledge and Data Engineering, 2009	Review of Sampling, Cost-Sensitive and Kernel-Based techniques for imbalanced datasets.
5	A Hybrid Approach to Learn with Imbalanced Classes using Evolutionary Algorithms [20]	Logic Journal of the IGPL, 2011	Review of created datasets on evolutionary algorithm.
6	An Overview of Classification algorithms for Imbalanced dataset [21]	The International Journal of Emerging Technology and Advanced Engineering,2012	Review of data-level and algorithmic-level approaches for imbalanced datasets.
7	A review on ensembles for the class imbalance problem: bagging boosting, and hybrid-based approaches [5]	IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews, 2012	Review ensemble techniques in imbalanced datasets, like Bagging, Boosting and Hybrid-Based techniques.
8	Imbalanced Learning:Foundations Algorithms, and Applications [29]	IEEE book,2013	Review of imbalanced learning algorithms in SVM, like Random sampling, SMOTE Different Error Costs, and zSVM techniques.
9	A survey of multiple classifier systems as hybrid systems [18]	Information Fusion, 2014	Review on multiple classifier system of Hybrid Intelligent Systems.
10	Imbalance dataset classification and Solutions: A Review [22]	International Journal of Computing and Business Research, 2014	Review of Data, Algorithmic, cost sensitive Feature selection, Ensemble-levels techniques for imbalanced learning.
11	A study on classifying imbalanced datasets [16]	International Conference on Networks Soft Computing, 2014	Review of imbalanced learning and observed that a combination of SMOTE and Bagging with Random Forest has better accuracy for the minority class.
12	A survey of predictive modeling under imbalanced distributions [6]	arXiv:1505.01658v2 [cs.LG], 2015	Review of Data Pre-processing Special-purpose Learning, Prediction Post-processing and Hybrid techniques.
13	Learning from imbalanced data: open challenges and future directions [8]	Progress in Artificial Intelligence, 2016	Review of classification, clustering data streams, big data.
14	Handling imbalanced data: A survey [9]	International Proceedings on Advances in Soft Computing, Intelligent Systems and Applications, 2018	Review of Data-Level, Algorithm-Level Ensemble, and Hybrid Approaches for imbalanced learning.
15	Sampling Approaches for Imbalanced Data Classification Problem in Machine Learning [10]	International Conference on Recent Innovations in Computing, 2019	Review of re-sampling techniques and evaluation of those techniques for five imbalanced datasets.
16	A Review on Solution to Class Imbalance Problem: Undersampling Approaches [14]	2020 International Conference on Computational Performance Evaluation (ComPE)	Insight of class imbalance issue of the undersampling approaches
17	A review of methods for imbalanced multi-label classification [13]	Pattern Recognition 2021	Review of characteristics of imbalanced multi-label datasets, evaluation measures and comparative analysis.

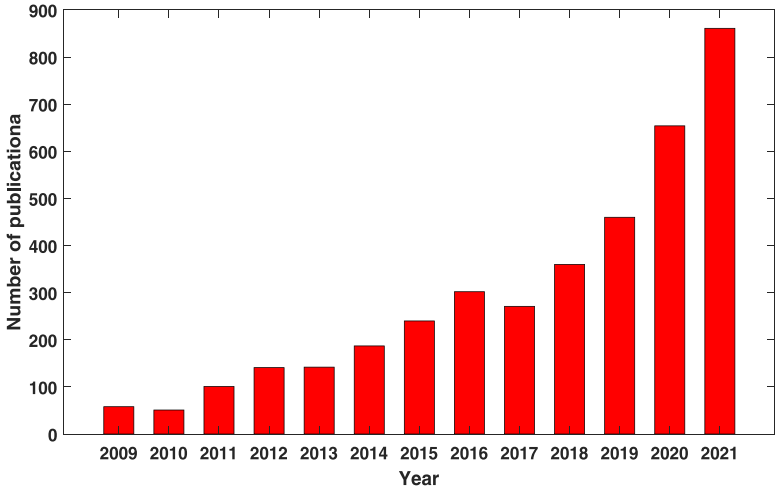

Fig. 2. Published papers on imbalanced domains.

future research. Moreover, we highlight the main difficulties and opportunities for the learning of imbalanced datasets. The major contributions of this paper are as follows:

(1) Most of the existing survey addresses only classification tasks. We also describe methods to deal with similar problems in regression tasks.

(2) We proposed a new taxonomy for class imbalance learning techniques.

(3) We discussed the advantages and disadvantages of each type of imbalanced learning technique.

(4) We emphasized the major opportunities and challenges in the imbalance area, which can be useful in research directions for learning algorithms.

The remaining of this paper is organized as follows: Section 2 presents performance evaluation in an imbalanced area. Section 4 considers multiple techniques for imbalanced class learning and categorized those techniques into three parts. Experimental outcomes are considered in Section 5. Some software and open source application for imbalanced classification are discussed in Section 6. We discuss and mention open challenges for future research on the imbalanced areas in Section 7. Conclusions are presented in Section 8.

2. Performance evaluation in imbalanced areas

As we know for class imbalanced datasets, predictive accuracy is a poor performance measure. For example, consider a dataset with two classes and 5000 records, with the 99% ratio of the majority class to the 1% minority class. If none of the minority samples is rightly classified, the predictive accuracy can be computed as $(4950 + 0)/(4950 + 0 + 50 + 0)$, which is 99 percent predictive accuracy. The main task in imbalanced datasets is to detect the minority sample. Here, we got 99% accuracy without considering the minority samples (i.e., zero percent for minority samples). Therefore, predictive accuracy cannot be applied to measure the performance of imbalanced datasets.

Accuracy and its complement error rate mostly used as indicators for estimating the performance of learning systems in classification issues. For two-class problems, accuracy can be considered as follows (Fig. 3): where TP = number of true positives, FP = number of false positives, FN = number of false negatives, TN = number of true negatives.

$$Accuracy = \frac{(TP + TN)}{(TP + FN + TN + FP)} \tag{1}$$

An alternatives measure for assessing the performance of an imbalanced dataset is the F-measure. It is the mean of the precision and recall score [34,35]. Therefore, the F-measure is calculated from the following equation:

$$F\text{-}measure = 2 * \frac{(Precision * Recall)}{(Precision + Recall)} \tag{2}$$

Positive predictive rate or precision is the number of correct positive predictions over adding the amount of correct positive predictions and incorrect positive predictions [36]. It can be computed by the following equation:

$$Precision = \frac{(TP)}{(TP + FP)} \tag{3}$$

Sensitivity or recall is the number of true positives over adding the number of true positives and the amount of false negatives [37]:

$$Recall = \frac{(TP)}{(TP + FN)} \tag{4}$$

Another performance measure is G-mean. It is the square root of the product of the specificity and sensitivity [38,39]. Specificity is the number of true negatives divided by the sum of the number of true negatives and the number of false positives. It can be displayed by the following equation:

$$G\text{-}Mean = \sqrt{sensitivity * specificity} \tag{5}$$

The receiver operating characteristics (ROC) curve (Fig. 4) and the corresponding area under the ROC curve (AUC) [40] are two general measures for imbalanced areas. In [41], authors offered ROC and AUC as substitutes for accuracy. The ROC curve helps the scholars for visualization of the relative trade-off between TP_{rate} and FP_{rate}.

The ROC curves not only present a single-value performance score but also allows the comparison of the most suitable model on average.

$$AUC = \frac{1 + TP_{rate} - FP_{rate}}{2} = \frac{TP_{rate} + TN_{rate}}{2} \tag{6}$$

3. Support vector machine(SVM)

This survey is based on the support vector machine and its application. I presented a detailed review of existing methods to deal with issues linked with class imbalance learning. So in this section, I am going to introduce SVM.

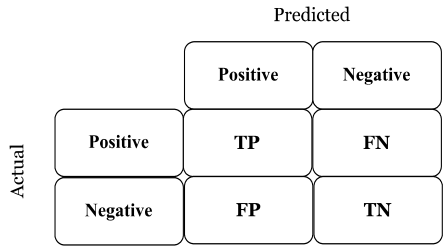

Fig. 3. Confusion matrix for two-class performance evaluation.

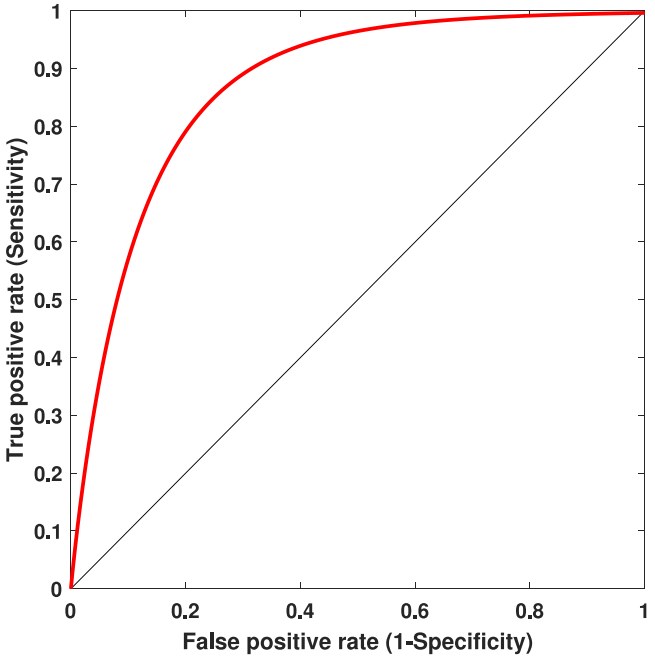

Fig. 4. ROC curve.

Support vector machines (SVMs) [42] have been proposed in the framework of the structural risk minimization principle. Consider the problem of binary classification problem with the training set $T = \{(x_1, y_1), \ldots, (x_l, y_l)\}$, where $x_i \in R^n$ is the input labeled by $y_i \in \{-1, +1\}$ for $i = 1, 2, \ldots, l$. The linear SVM classifier search for an optimal separating hyperplane

$$w^T x + b = 0, \tag{7}$$

where $b \in R$ is the bias term and $w \in R^n$ is the normal vector to the hyper-planes. The hyperplane described by (7) is between the bounding hyper-planes given by

$$w^T x + b = -1, \quad and \quad w^T x + b = 1. \tag{8}$$

The margin of separation between the two disjoint hyper-planes is given by $\frac{2}{\|w\|_2}$. The standard SVM is the maximum margin classifier, which is collected by maximizing this margin and gets the following problem:

$$\min_{w,b,\xi} \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{l}\xi_i,$$

$$s.t. \quad -y_i(w^T x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \ i = 1, 2, \ldots, l. \tag{9}$$

where $\geq_i \in R$ is the soft margin error of the *ith* training point, $C > 0$ is the regularization parameter that balances the importance between the maximization of the margin and the minimization of the training error. The Wolfe dual of (9) is:

$$\max_{\alpha} \sum_{j=1}^{l}\alpha_j - \frac{1}{2}\sum_{i=1}^{l}\alpha_j\sum_{j=1}^{l}y_i y_j(x_i.x_j)\alpha_i\alpha_j,$$

$$s.t. \quad \sum_{i=1}^{l}\alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C, \ i = 1, 2, \ldots, l. \tag{10}$$

where $\alpha \in R^l$ are Lagrangian multipliers. The optimal separating hyperplane (7) can be obtained from the solution α of (10) by

$$w = \sum_{i=1}^{l}\alpha_i y_i x_i, \qquad b = \frac{1}{N_{SV}}\sum_{j=1}^{N_{SV}}(y_j - \sum_{i=1}^{l}\alpha_i y_i(x_i.x_j)), \tag{11}$$

where N_{SV} displays the number of support vectors such that $0 < \alpha_i < C$. We classify a new data point as $+1$ (*respectively*, -1) according to whether the decision function, $Class\ i = sgn(w^T x + b)$, yields 1 (*respectively*, 0). Algorithm 1 shows the pseudo code of the support vector machine.

Algorithm 1 Training SVM

Require: **x** and **y** loaded with training labeled data, $\alpha \Leftarrow 0$ or $\alpha \Leftarrow 0$ partially trained SVM
1: $C \Leftarrow$ some value (for example $[-10, 10]$)
2: **repeat**
3: **for all** $\{x_i, y_i\}$, $\{x_j, y_j\}$ **do**
4: Optimize α_i and α_j
5: **end for**
6: **until** no change in α or other resource constraint criteria met
Ensure: Retain only the support vectors ($\alpha_i > 0$)

4. Modeling strategies for handling imbalanced domains

We will face significant challenges during building predictive models in imbalanced domains. Several strategies have been developed in the classification tasks to address this problem. We classified existing approaches in the imbalanced area into the following three main categories:

(1) Data Pre-processing,
(2) Algorithmic structures,
(3) Hybrid techniques.

Those three types of strategies will be reviewed in Sections 4.1–4.3, including solutions for both classification and regression tasks. Fig. 5 represents the different existing approaches of each category. Also, we mentioned the disadvantages and advantages of different imbalanced learning techniques in Table 2.

4.1. Data pre-processing

The strategy of data pre-processing techniques is to pre-process the given imbalanced dataset and modify the data distribution to make use of the standard algorithms. We pre-process the datasets to modify the distribution, instead of directly using learning techniques to the presented datasets. These techniques have the following benefits:

(1) This strategy can be utilized for any existing learning methods,

(2) Selected models are biased to the targets of the researcher because the data distribution was already modified to fix these targets, and it is expected the models are most explainable in the circumstances of these targets.

These techniques have the following drawbacks:

Table 2
Advantages and disadvantages of imbalance learning methods.

Methods	Advantages	Disadvantages	References
Data Pre-processing	(1) This strategy can utilize for any existing learning methods, (2) Selected models are biased to the targets of the researcher and expected that the models are explainable, (3) Reduces the risk of overfitting which is introduced when replicas of the examples are inserted in the training set, (4) Improves the ability of generalization which was compromised by the over-sampling methods.	(1) It is tough to connect the data distribution with the target loss function, (2) It is tough to define which are the relevant observations and the "normal" samples, (3) It is tough to produce new synthetic samples, (4) It is tough to determine the value of the target variable in the synthetic samples.	[6–86]
Algorithmic structures	(1) The aims of the user are included straight into the models, (2) The models used in this technique are much more understandable for users.	(1) We do not know the chosen algorithm for modification can optimize our aim or should build a new one, (2) An often unavailable cost/cost–benefit matrix, (3) We need strong knowledge to select an appropriate algorithm for good modification.	[87–119]
Hybrid techniques	(1) Hybrid can solve advanced and complex problems, (2) Hybrid can allow classifiers has better performance than other classification techniques and instance selection if set up an appropriate assessment metric.	(1) Perfectly balanced data may not be optimal, (2) It is difficult to determine the appropriate number of over/under-sample to utilize in this system, (3) It is difficult to find an effective estimation approach for parameters.	[120–152]

1- It is tough to connect the data distribution with the target loss function because mapping the data distribution into the proper new one is not easy,

2- The initial algorithm is based on the information presented by the user concerning which class value is usually known as the minority or positive class. So, It is tough to define which are the relevant observations and the "normal" samples,

3- It is tough to produce new synthetic samples,

4- For the generation of new synthetic samples, the target value of each synthetic sample was calculated as a weighted average of the target variable values of the two samples. So, it is tough to determine the value of the target variable in the synthetic samples.

4.1.1. Data pre-processing for classification

(i) Random Sampling: The available re-sampling techniques are designed based on different strategies, such as random over-sampling, random under-sampling, clustering algorithms, data cleaning approaches, and evolutionary algorithms. Under-sampling and over-sampling [6,43–47] are two main re-sampling techniques that can be used for data distribution. These techniques can be decreased and increased the influence of one class in the dataset to track-back the optimal misclassification costs, respectively. By eliminating samples from the dataset, it prevents the increase of the imbalance ratio. On the other hand, it has a bad influence on our performance because it can raise the likelihood of over-fitting (particularly for high over-sampling scores).

Random over-sampling and under-sampling are two straightforward techniques for dealing with the issue of class imbalance learning [43–47]. The additional computational cost of using over-sampling is unjustified as the performance obtained by under-sampling [48–53]. Under-sampling and ensemble learning combined as bagging [54–56] and balanced random forest (BRF) [57], which modify the distribution of datasets.

Although randomly selecting examples is an excellent strategy, over-sampling and under-sampling techniques can also be carried out as powerful techniques. For example, under-sampling can be performed as an employee to distance evaluations [58]. These techniques desire to have positive class over-generalized based on the selection of distant examples. The main weakness of these techniques is poor time-consuming when working with large datasets.

To make a balance between classification accuracy and distribution for the under-sampling imbalanced dataset, we can use Evolutionary Algorithms (EAs). The purpose of this technique is to offer evolutionary prototype selection algorithms that are dealing with the issue of the imbalanced dataset, via applying a new fitness function [59,60].

(ii) Synthetic Sampling with Data Generation: Another significant approach as a pre-processing technique for handling imbalance issues is a repetition of the decision region in a classification decision for the minority class. Synthesizing new data has multiple famous benefits [48,61], like:

(1) Decreases the risk of over-fitting, which happens when replication of the samples is placed in the training set,

(2) Increases the capability of generalization, which is damaged by the over-sampling techniques.

The synthesizing new data techniques can be divided into two groups: 1- uses an interpolation of existing examples, 2- present perturbations. A well-known technique that used interpolation is the Synthetic Minority Over-sampling Technique-SMOTE [48]. This method presents a different experiment on some datasets with different imbalance ratios and different numbers of data in the training set. This technique shows a combination of over-sampling the minority class and under-sampling the majority class. Because of that reason, the proposed method conducts better than Ripper's loss ratio and Naive Bayes. However, this technique is still learning with different cost functions from the information presented in the dataset.

Class-imbalanced crash prediction based on real-time traffic and weather data was proposed in [62], which tries to develop real-time crash prediction models that might be employed within

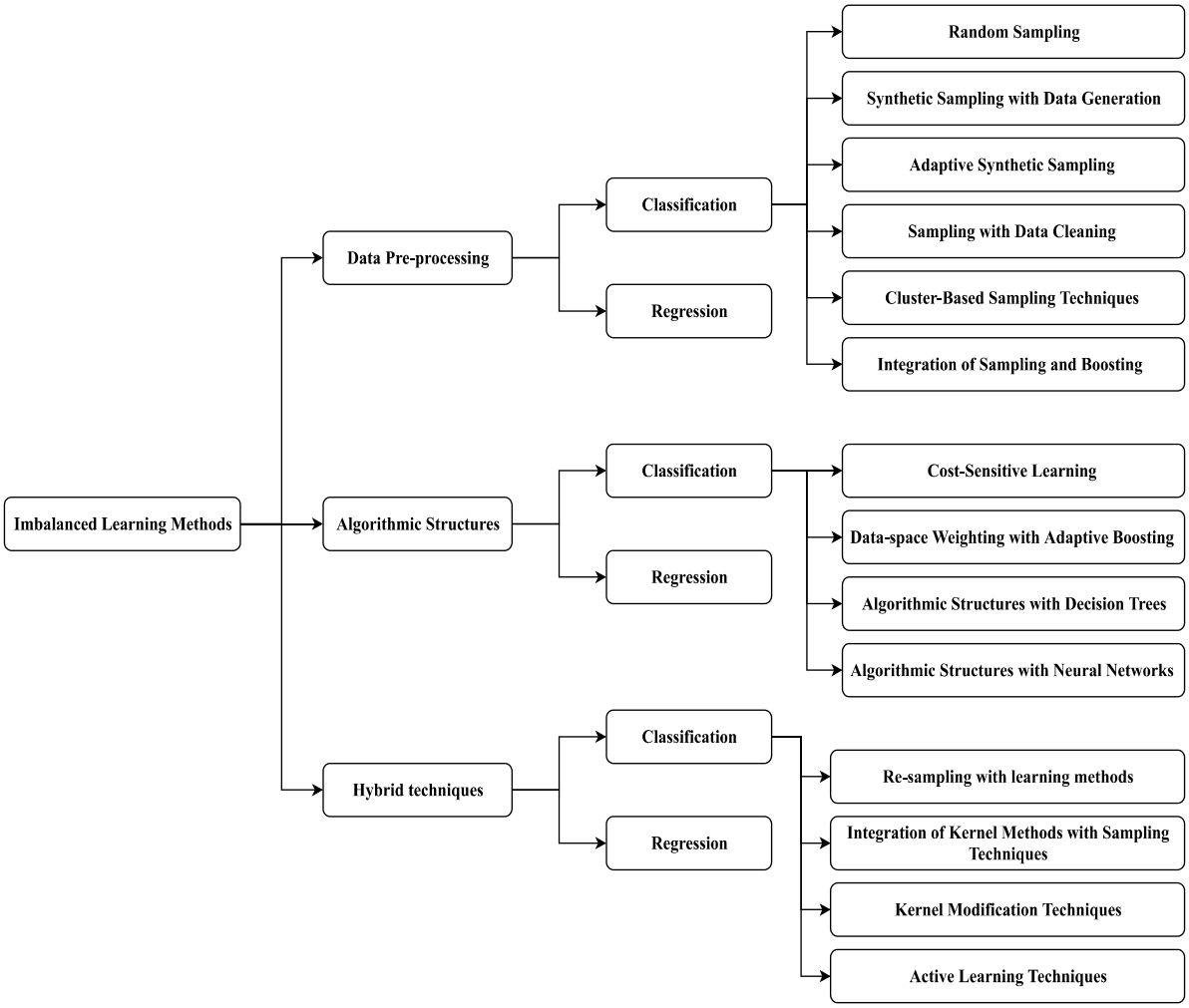

Fig. 5. Main modeling strategies and the main references for imbalanced domains.

traffic management systems. In this technique, the authors have developed two model predictions based on the popular machine learning techniques, Support Vector Machine (SVM) and deep neural network Multilayer Perceptron (MLP). Moreover, since crash events typically happen in rare instances tending to be under-represented in the dataset, an imbalance-aware approach to overcome the issue was adopted using SMOTE. The outcomes show that MLP displayed the best-performing prediction results, in which MLP recall values were above 94%. Over the last decade, SMOTE-based techniques have been used and extended to overcome the issue of imbalanced datasets. In [63], the authors introduce alterations to a priori based techniques Safe Level Oversampling Using Propensity Scores (OUPS). This method result in an improvement in sensitivity measures over competing approaches using the SMOTE-based method such as the Local neighborhood extension to SMOTE (LN-SMOTE), Borderline-SMOTE and Safe-Level-SMOTE. This technique resulted in the highest average sensitivity and G-mean measures overall and did well with SVM-based learners. Using multi techniques emphasized the strength of the Borderline methods when using feed-forward neural networks and probabilistic learners on average.

A case study for patients with contingent primary cancers has been proposed in [64], which is used an Artificial neural network and a Bayesian network. Moreover, a synthetic minority oversampling technique was utilized to handle the problem of imbalanced datasets. COVID-19 is a popular issue these days, and it is generating a huge number of infections in people. Deep learning-based image diagnostic technology can effectively improve the weaknesses of the current main detection technique. In [65], the model merged with Generative Adversarial Network (WGAN), and the Deep Neural Network classifier is utilized to effectively solve the issue of multi-classification of COVID-19 images with small samples, to achieve the goal of effectively distinguishing COVID-19 patients. This paper used SMOTE to generate the simulated sample and trained the classification model, and then utilized it in the original sample test model. A new method named Multinomial Mixture Modeling with Median Absolute Deviation and Random Forest Algorithm (MMM-RF) is suggested for the classification of network attacks [66]. The study investigates the use of SMOTE associated with random under-sampling in controlling imbalance in the dataset.

Three disadvantages in the SMOTE algorithm are discovered, which is the appearance of some different kinds of these methods [3,34,67–73].

(1) Using the post or pre-processing for after or before the application of SMOTE,

(2) Just in certain chosen areas of input space, we can use of SMOTE,

(3) We do not have enough information about the modification of the SMOTE algorithm.

(iii) Adaptive Synthetic Sampling: The concern of the binary classification problem for an imbalanced dataset is the appropriation of a new sample for one of two classes (positive or negative), in which the number of the first class is much more than in the second class. In [74], the author suggested an impressive

classification by allocating more weights for the minority class. This technique is for generating noisy replicates for the minority class while keeping the majority class without any change. The average of various model estimates is taken to create several variants of noise-added training sets from a given dataset [75]. This method prepares an impressive and model-free regularization for classification techniques.

Another type of SMOTE only creates synthetic samples in particular areas, which is helpful for learning algorithms. But, the meaning of "good region" is still unclear. Many strategies have been introduced to solve this problem. Some of those methods concentrate the synthesizing attempt on the boundaries between classes. Others effort to decide which are harder to learn samples and focus on boundaries. Borderline-SMOTE [34], MWMOTE [69], ADASYN [3], modified synthetic minority over-sampling technique (MSMOTE) [76], FSMOTE [77] are some instances of those strategies. Based on another kind of SMOTE, some modifications are applied to generate synthetic instances. For example, the synthetic instances can be produced near or far away from a sample regarding some measure, like LN-SMOTE [68], Safe-Level-SMOTE [70], Safe Level Graph [78], and DBSMOTE [79].

(iv) Sampling with Data Cleaning Techniques: Under-sampling methods also can be achieved by data cleaning techniques. The main aim of these techniques is to recognize possible noisy examples or overlapping regions and then eliminate those samples [80,81]. These techniques worthlessly reduce the size of the dataset by removing most samples on the incorrect side boundary. If the modified system is processed by the condensed nearest neighbor (CNN) or another technique, the outcome is alleviating highly significant as compared to available on the original set. Those significant outcomes happened because the modified dataset is cleaner than the original one. The concept of CNN [82] either used to conduct under-sampling [83]. The CNN correctly classifies the rest of the left points in the sample set when applied as a saved reference set for the NN role. Tomek links and CNN techniques were mixed in the minimum consistent subset (i.e., a consistent subset with a minimum number of elements) with a strategy named One-Sided-Selection (OSS) [83]. This technique is the opposite of the proposed method in [81].

The initial SMOTE techniques are using the SMOTE algorithm, and then they utilize a post-processing strategy for the modification of datasets. For instance, SMOTE+ENN [81], SMOTE+Tomek [81], SMOTE+FRST [71] or SMOTE+RSB [84]. A new development of SMOTE over-sampling techniques is the fuzzy rough imbalanced prototype selection (FRIPS)-SMOTE and the first time introduced by [72]. In this technique, first, the noise level of each sample is calculated by applying a measure based on fuzzy rough theory. Then, all samples deleted that have a noise level more formidable than a defined threshold. In this step, recognizing the noise level is managed by a wrapper system. Finally, after the aforementioned pre-processing strategy, SMOTE is employed and called FRIPS-SMOTE.

These techniques consisted of random over/under-sampling, SmoteSVM, and SCMs [48,85–89]. In [43], the author proposed a technique that is effective for over-sampling in SVMs. In this technique, first of all, the separating hyperplane located by training SVM on the primary imbalanced datasets are utilized to choose the most informative instances for a given classification issue, which are the instances lying around the class boundary area. Then, just these chosen samples are balanced by over-sampling instead of picking the complete dataset. This technique decreases the SVM training time effectively as compared to the classification outcomes in the original over-sampling method.

(v) Cluster-Based Sampling Techniques: In [90], the authors investigated the variations of performance in one-class and binary classifiers as a level of increasing imbalance ratio, and therefore, they considered uncertainty in the second class. They did experiment on different datasets (artificial and UCI), and efficiency observed the one-class and binary classifiers as the size of the second class progressively decreased. The outcomes indicated that whereas one-class classifiers were almost stable, and the performance of binary classifiers decreased. In [91,92], two techniques under two different situations are discussed. The aim was to test and prove the impact of class imbalance learning in clustering. In [93], the authors suggested cluster-based over-sampling (CBO) as an over-sampling method. The CBO technique can deal with between-class within-class imbalance simultaneously. Also, it used random over-sampling to over-sample the sub-clusters, although the issue of over-fitting maybe happens. Many other ideas on clustering methods exist in [94–96].

In [97], the authors presented support cluster machines (SCMs) to consider as another re-sampling technique for SVM. This technique separated the negative samples into disjoint clusters by utilizing the kernel-k-means clustering technique. Then, it trains the primal SVM algorithm to utilize positive samples and the delegates of the negative clusters, i.e., the data samples delegating the cluster centers. Therefore, a shrinking method is employed to delete the instances which probably are not support vectors. This approach of shrinking and clustering is conducted iterative many times till convergence.

These techniques also applied as an explanation for training SVM in imbalanced areas [98–101]. In fact, in these techniques, the majority samples are divided into several sub-samples such that every one of these sub-samples has the same number of samples in minority samples. This may be prepared via random sampling, bootstrapping, and clustering methods. Then, a couple of SVM classifiers are used, which each method trained by different negative sub-samples and similar positive samples. In the end, the decisions made base on a method like majority voting, which is a classifier ensemble.

(vi) Integration of Sampling and Boosting: SMOTE method has been used with a couple of different classifiers and also unified with boosting [67] and bagging [54]. This technique produces synthetic examples in positive samples. Moreover, the negative samples are ignoring because of over-generalization [68,94,95]. This technique may be especially problematic when the minority class samples are very scattered in highly skewed class distributions. Therefore, it is better to use the class mixture.

Random over-sampling examples (ROSE) is another framework for handling imbalanced classification problems according to a smoothed bootstrap re-sampling method that is introduced by [61]. The main benefit of this technique is the idea of blending the generation of synthetic data with ensemble learning. They claimed that ROSE is stringently associated with bootstrap techniques, which is supporting its general extension regarding bagging learners. Therefore, this strategy leads to improve performance of classification. Furthermore, smoothed boosting may be readily done in conjunction with ROSE by corresponding the weights update strategies to the probability of data generation.

Moreover, particular boosting approaches have been applied in class imbalance learning via ensemble configuration like Adacost [102], RareBoost [103], and SMOTEBoost [67]. Also, those techniques can be used with SVM.

In this part, we want to investigate some pre-processing techniques on SVM, kNN, and Random Forest to find out which one is better to use. We conducted experiments on 4 imbalanced datasets with an imbalance ratio (IR) from 8.10 to 41.40 and these binary datasets are downloaded from KEEL imbalanced datasets [104]. We describe all datasets in Table 3, and 5-fold cross-validation is utilized for all datasets. These outcomes with respect to the AUC rate are listed in Tables 5, 6, and 7.

For the experiment, we used original data, Over-sampling, Under-sampling, SMOTE, and Both Over-sampling & Under-

Table 3
Details of the imbalanced datasets.

No	dataset	Positive	Negative	Instance	Dimension	Im. Ratio
1	Yeast 3	163	1,321	1,484	8	8.10
2	Yeast 4	51	1,433	1,484	8	28.10
3	Yeast 5	44	1,440	1,484	8	32.73
4	Yeast 6	35	1,449	1,484	8	41.40

Table 4
Parameters of the preprocessing techniques.

No	dataset	Over-sampling	Under-sampling	SMOTE	Both Over and Under-sampling
1	Yeast 3	N = 326	N = 2642	perc.over = 100, perc.under = 200	N = 1000, p = 0.5
2	Yeast 4	N = 102	N = 2866	perc.over = 100, perc.under = 200	N = 1000, p = 0.5
3	Yeast 5	N = 88	N = 2880	perc.over = 100, perc.under = 200	N = 1000, p = 0.5
4	Yeast 6	N = 70	N = 2898	perc.over = 100, perc.under = 200	N = 1000, p = 0.5

Table 5
Classification outcomes (AUC rate) for SVM algorithm.

No	dataset	Original	Over-sampling	Under-sampling	SMOTE	Both Over and Under-sampling
1	Yeast 3	86.42	92.72	91.53	90.56	91.48
2	Yeast 4	53.80	91.93	84.68	69.09	91.51
3	Yeast 5	74.24	93.19	97.08	95.45	93.16
4	Yeast 6	65.47	92.79	87.60	82.97	92.65

Table 6
Classification outcomes (AUC rate) for kNN algorithm.

No	dataset	Original	Over-sampling	Under-sampling	SMOTE	Both Over and Under-sampling
1	Yeast 3	84.58	96.92	93.44	91.26	94.38
2	Yeast 4	68.68	93.04	89.17	87.52	94.79
3	Yeast 5	84.40	98.78	97.12	95.97	98.12
4	Yeast 6	78.26	95.37	90.00	88.12	95.93

sampling techniques. If we use both under-sampling and over-sampling on the imbalanced data, the minority class is over-sampled with replacement and the majority class is under-sampled without replacement. All the datasets used the parameters shown in Table 4. N points out the number of samples in the resulting balanced set. For example, there were originally 35 positive samples in the yeast6 dataset. N is equal to 70, which results from a balanced dataset after under-sampling. Therefore, we have 35 positive and 35 negative samples. The parameter p is the probability of the newly generated sample of a positive class. We set $prec.over = 100$ to double the number of positive samples, and set $perc.under = 200$ to keep half of what it created as negative samples [105].

Based on Table 5, we conducted four different pre-processing techniques of SVM on four imbalanced datasets. One can see, Over-sampling technique gets the best results for Yeast3, Yeast4, and Yeast6 datasets. While for the Yeast5 dataset, Under-sampling technique obtains the best outcomes.

Based on Table 6, we conducted four different pre-processing techniques of kNN on four imbalanced datasets. Obviously, Over-sampling technique gets the best result for Yeast3 and Yeast5 datasets. While for the Yeast4 and Yeast6 datasets, Both Over and Under-
sampling technique obtains the best outcomes.

We conducted four different pre-processing techniques of Random Forest on four imbalanced datasets. The outcomes are shown in Table 7. We can see, Over-sampling technique gets the best result for Yeast3, Yeast5, and Yeast6 datasets. While for the Yeast4 dataset, Both Over and Under-sampling technique obtains the best outcomes.

Figs. 6–9 compare pre-processing techniques on four datasets that are conducted on different classification methods. For the Yeast3 dataset, we can conclude that the Random forest gets almost the best results. But the kNN method obtains almost the best outcomes for the Yeast4 Yeast5, and Yeast6 datasets.

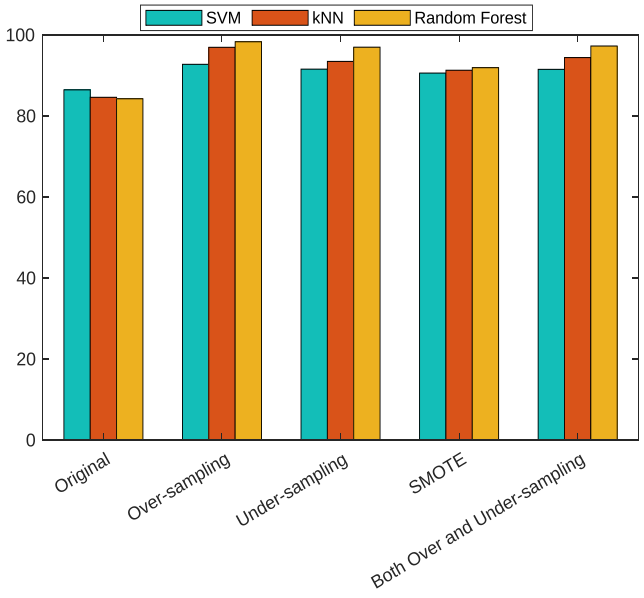

Fig. 6. AUC rate between different techniques for Yeast3 dataset.

4.1.2. Data pre-processing for regression

There are many methods for handling classification problems [106]. But for regression issues, only a few methods for creating new synthetic data were introduced.

For regression tasks, [107] modifies the distribution of the dataset to overcome the problem of imbalance between the rare target cases and the most frequent ones. They used the Smote algorithm for a modification, which allows them to apply it to the regression tasks. In spite of the prospective randomly selecting

Table 7
Classification outcomes for (AUC rate) Random Forest algorithm.

No	dataset	Original	Over-sampling	Under-sampling	SMOTE	Both Over and Under-sampling
1	Yeast 3	84.22	98.29	96.95	91.89	97.24
2	Yeast 4	56.77	91.93	91.99	82.57	93.81
3	Yeast 5	77.99	98.68	96.91	94.34	98.26
4	Yeast 6	71.26	95.47	92.76	87.36	95.09

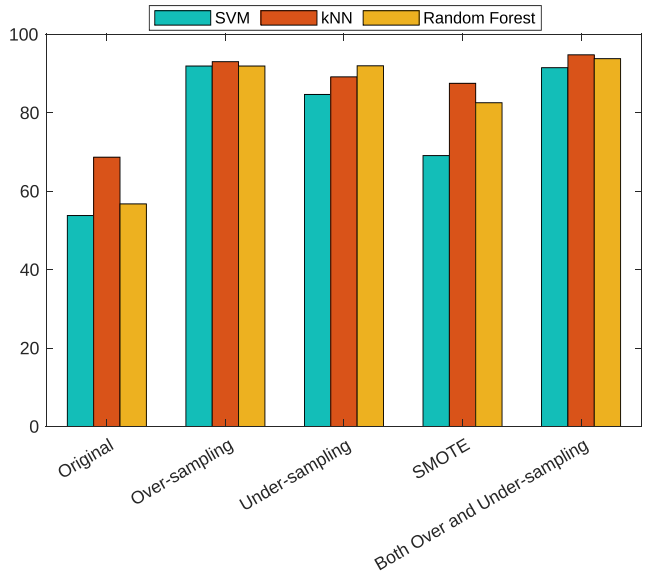

Fig. 7. AUC rate between different techniques for Yeast4 dataset.

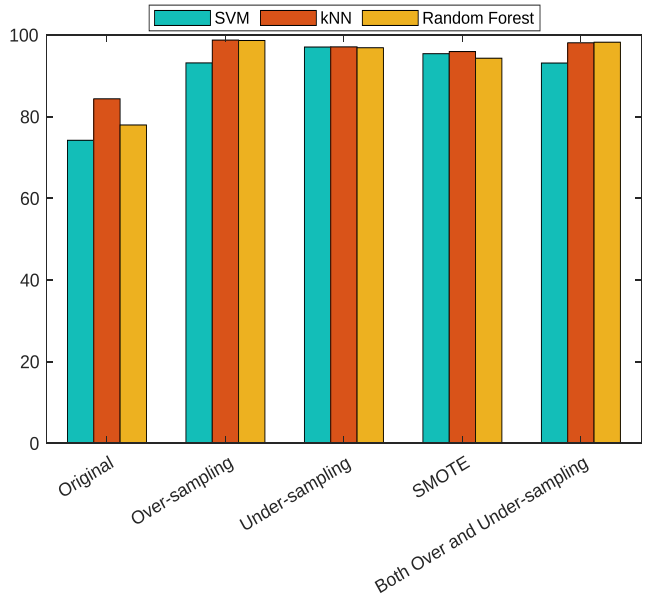

Fig. 8. AUC rate between different techniques for Yeast5 dataset.

samples, over and under-sampling approaches can be handled by another knowledgeable technique.

A well-known modification of the SMOTE algorithm is proposed in [47], which allows us to use these regression tasks. Three main questions of the smote algorithm have existed, which is required to handle in order to accommodate it for regression tasks:

(1) How can we explain which observations are appropriate and regular cases?

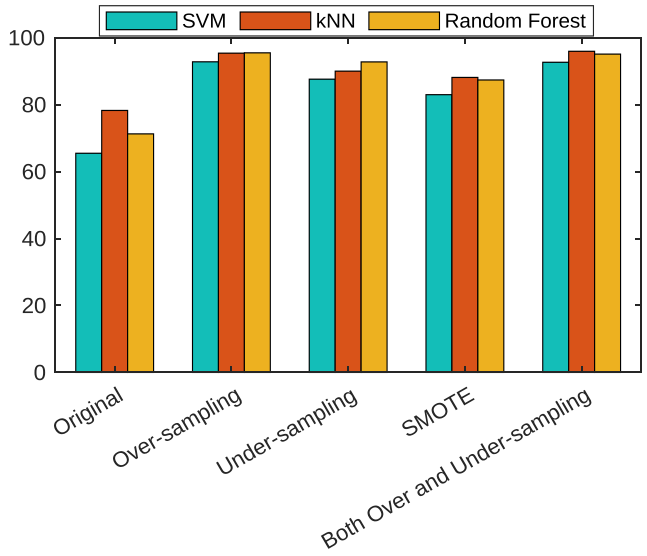

Fig. 9. AUC rate between different techniques for Yeast6 dataset.

(2) How can we create new synthetic examples like under-sampling or over-sampling?

(3) How can we choose the objective variable value of those new synthetic samples?

As an answer to the first question, the original algorithm is deployed on the information presented by the scholar concerning which class value (positive or negative class) is the target class. For creating new samples in the second question, a similar approach as the main algorithm is employed. However, some small modifications were introduced for being capable of dealing with both nominal and numeric attributes. In the end, the main problem with the third problem is to choose the target variable value of the created samples [47].

In this part, we want to investigate Logistic Regression technique on four datasets. 5-fold cross-validation is utilized for all datasets. These outcomes with respect to the AUC rate are listed in Table 8 and Fig. 10. One can see, Both Over and Under-sampling strategy gets the best results for Yeast3 and Yeast5. On the other hand, Over-sampling obtains the best outcomes for Yeast4 and Yeast6.

4.2. Algorithmic structures

Algorithmic structure approaches create new algorithms or modify existing algorithms to handle imbalanced problems. These methods have the following advantages:

(1) The aims of the user are included straight into the models,

(2) The models used in this technique are much more understandable for users.

These techniques have the following drawbacks:

(1) An often unavailable cost/cost–benefit matrix,

(2) We need strong knowledge to select an appropriate algorithm for the modification in the data distribution,

(3) It is difficult using existing methods with a different learning system, which contrasts with pre-processing approaches.

Table 8
AUC rate for Logistic Regression.

No	dataset	Original	Over-sampling	Under-sampling	SMOTE	Both Over and Under-sampling
1	Yeast 3	82.86	91.12	89.74	89.35	91.13
2	Yeast 4	56.82	85.23	80.98	81.44	84.27
3	Yeast 5	73.7	97.57	95.00	92.99	97.77
4	Yeast 6	66.94	89.50	87.88	87.10	88.75

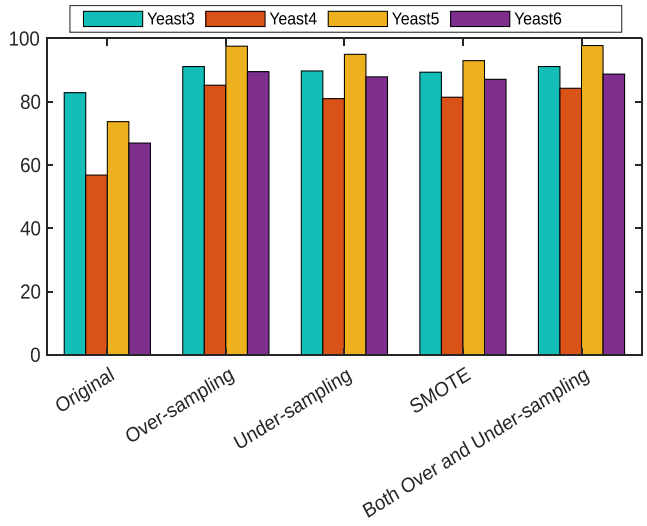

Fig. 10. AUC rate for Logistic Regression between different datasets.

4.2.1. Algorithmic structures for classification

(i) Cost-Sensitive Learning: The approach of weighting samples is a solution for implementing cost-sensitive learning. Misclassification costs are used in a dataset to choose better training distribution. In several cases, the cost of not identifying some samples in the minority class are high. Therefore, for the classifier in this case that does not get misclassification costs, the performance is not well. In exceptional cases, due to many samples associating with the majority class, disregarding costs generate a useless model (although misclassifications cost in the minority class is high). Some techniques for converting classifiers and classification models into cost-sensitive algorithms are considered in [108]. The recommended transformation is according to the cost weighting of the training samples, which can be earned by accurate sub-sampling or by feeding the weights to the classification algorithm. They proposed a cost function according to the ensemble aggregation and cost-proportionate rejection sampling, which obtains perfect predictive accuracy on two publicly available datasets, while intensely decreasing the estimation needed by other techniques.

Two fundamental techniques exist for dealing with those problems: (1) Transparent Box: provide the costs of the training sample as instance weights to the classifier algorithm, while unable to be used to arbitrary classifier learners. (2) Black Box: re-sample based on those same weights that outcomes are in extreme over-fitting if we are using re-sampling with substitution. To defeat those disadvantages, scholars have offered a technique named cost-proportionate rejection sampling that allows every sample in the input data with probability equivalent to its related weight.

Random forests (RF) algorithms have also been considered to tackling with imbalanced datasets with cost-sensitive transformation. Algorithm 2 illustrates the algorithm of Random forest. In [57,109], the authors employed random forest to suggest two ways to tackle with imbalanced dataset classification issue. One solution is cost-sensitive learning, and the other one is the under-sampling method. In the weighted random forest, we set more weights on the minority samples, therefore penalizing much highly on misclassifying the minority samples.

Algorithm 2 Random forest classifier

1. Select randomly M features from the feature set.
2. For each x in M
   i) calculate the Information Gain

$$Gain(t,x) = E(t) - E(t,x)$$

$$E(t) = \sum_{i=1}^{c} -P_i \log_2 P_i$$

$$E(t,x) = \sum_{c \in X} -P(c)E(c)$$

   where $E(t)$ is the entropy of the two classes, $E(t,x)$ is the entropy of feature x.
   ii) select the node d which has the highest information gain
   iii) split the node into sub-nodes
   iv) repeat step i), ii), and iii) to construct the tree until reach minimum number of samples required to split
3. Repeat steps 1 and 2 for N times to build forest of N trees

There exist another strategy that is also changing the preference metrics of the algorithms while not trust indirectly on the explanation of a cost/cost–benefit matrix. For example, when the training samples of the target class are sufficiently further by the other training samples, the class-boundary determined by SVMs can be seriously skewed within the target class [110]. To deal with the class imbalance issue in the presence of noise and outliers, fuzzy support vector machines for class imbalance learning (FSVM-CIL) introduced by [111]. A potential support vector machine (P-SVM) is a new technique that introduces a novel optimization approach different from normal SVM [112]. By the way, this technique has limitations in tackling with high imbalanced ratio and large-scale datasets.

The K-NN algorithm was also prepared to deal with the imbalance issue. Weighted distance hired for the classification tasks, which is proposed in [113]. This method lonely generates a remarkable improvement for all datasets in the performance measures. Therefore, the weighted distance illustrates itself as an excellent tool to transform the classification method for getting into account the class imbalance, even though other weighting circumstances necessity yet investigated. Although with acceptable outcomes, a common issue to all those make smaller methods is that they do not let to control the number of samples which eliminated.

Some classifiers are called soft classifiers since they present a score function for indicating the degree to which an example is a member of a class. The score function can be applied as a threshold to produce other classifiers. This strategy with different thresholds can be employed relating to the classes [114]. The actions of shifting the decision threshold are using a sampling strategy and set the cost matrix to generate classifiers with the same efficiency [115].

Utilizing different error costs for the negative (C^-) and positive (C^+) samples are suggested by [116]. The author proposed changing the Lagrangian function to the following equation,

$$L_p = \frac{\|w\|^2}{2} + C^+ \sum_{\{i|y_i=+1\}}^{n_+} \xi_i + C^- \sum_{\{j|y_j=-1\}}^{n_-} \xi_j - \sum_{i=1}^{n} \alpha_i [y_i(w.x_i + b) - 1 + \xi_i] - \sum_{i=1}^{n} r_i \xi_i \quad (12)$$

With the following conditions:

(1) If $y_i = +1$, then $0 \le \alpha_i \le C^+$,

(2) If $y_i = -1$, then $0 \le \alpha_i \le C^-$.

Moreover, we note that $\xi_i > 0$ only when $\alpha_i = C$ [117]. Thus, non-zero error costs on negative support vectors (resp. positive support vectors) will have a smaller α_i (resp. larger α_i). It is efficient that the boundary is pushed further in the direction of negative samples. To utilize the SMOTE method of over-sampling the minority samples, the distribution of positive samples was denser. Finally, the strategy of the SDC technique is including three steps:

(1) Due to the loss of information, the under-sampling of the majority samples does not happen,

(2) SDC utilized various error costs for different classes to push the boundary far from the positive samples,

(3) To build the boundary, SMOTE utilized to make positive samples more densely distributed.

In [118], the authors proposed the bilateral-weighted fuzzy support vector machine (B-FSVM) algorithm to decrease the issue of class imbalance. This technique just regarded all the datasets are balanced, and it usually generates a biased to the majority samples when using in the imbalanced dataset. In this case, the minority class has a low performance.

In B-FSVM, every sample allocates a negative and positive class membership function. Suppose in sample x_i, negative and positive membership amounts are defined as s_i^- and s_i^+. These membership functions are denoted by:

$$s_i^+ = m_i^+ r^+, \; s_i^- = m_i^- r^- \quad (13)$$

In Eq. (8), r^- and r^+ are decreased the impact of class imbalance, so that $r^+ \ge r^-$. Hence, each instance can get a negative class membership function in $[0, r]$ and positive membership function in $[0, 1]$, where $r^- = r$, $r^+ = 1$ and $r < 1$ is the class ratio. The most important key is how to generate the membership function. Therefore, the decision function can define as:

$$f(x) = sgn(w.\Phi(x) + b) = sgn(\sum_{i=1}^{n} (\alpha_i - \beta_i) K(x, x_i) + b). \quad (14)$$

Another modification technique for handling class imbalanced learning for SVM algorithms is zSVM [119]. In this technique, first, the SVM algorithm is improved by utilizing the primary imbalanced training dataset. Next, the decision boundary of the outcome model is changed to eliminate its bias to the majority samples. General SVM decision function can be changed as follows:

$$f(x) = sign(\sum_{i=1}^{l} \alpha_i y_i K(x, x_i) + b) = sign(\sum_{i=1}^{l_1} \alpha_i^+ y_i K(x, x_i) + \sum_{j=1}^{l_2} \alpha_j^- y_j K(x, x_j) + b) \quad (15)$$

where α_j^- (resp. α_i^+) are the coefficients of the negative support vectors (resp. positive support vectors), l_1 and l_2 demonstrate the number of positive and negative training samples. zSVM technique increases the value of α_i^+ in the positive support vectors by multiplying them with a special positive value z. Therefore, the changed SVM decision function demonstrated as follows:

$$f(x) = sign(z * \sum_{i=1}^{l_1} \alpha_i^+ y_i K(x, x_i) + \sum_{j=1}^{l_2} \alpha_i^- y_j K(x, x_j) + b) \quad (16)$$

The weights of the positive support vector in the decision function are increased by modification of α_i^+, and then it would reduce its bias to the majority negative samples. The value of z admits the best classification outcomes for the training dataset, which chosen as the optimal value [119].

Two re-balancing techniques for training SVM with a high imbalance ratio are collected in [120,121]. The first technique trained the SVM algorithm just with the minority samples and the other one, i.e., DEC (different error costs) technique, has been developed to define misclassification cost $C^- = 0$ (resp. $C^+ = 1/N^+$) for the majority (resp. minority) samples, which N^+ is the total number of minority samples. Based on experiments for high imbalance ratio real-world and synthetic datasets, these techniques performed more efficiently as compared to main data re-balancing techniques.

There are many techniques for making strategy cost-sensitive in a post hoc manner. These kinds of approaches are mostly investigated for classification tasks and for making it cost-sensitive, and the goal is just changing the model predictions [122,123]. It means these approaches can be used to imbalanced dataset distributions.

(ii) Data-space Weighting with Adaptive Boosting: In machine learning, boosting is a famous method according to the idea of making high prediction accuracy by mixing several almost weak and inaccurate rules. AdaBoost is an ensemble technique according to the principle of producing several predictors and weighted voting between weak classifiers [124]. This technique gives equal weight to every misclassified sample, in which the error rate is not the same for each class. Usually, the misclassification error of the minority samples is higher than the majority samples. Therefore, the Adaboost algorithm causes a smaller margin and higher bias when faced with the skew distribution. The algorithm of AdaBoost can be found in algorithm 3. Some famous papers for imbalanced distribution concepts are RareBoost [103], AdaC1, AdaC2 and AdaC3 [125], and BABoost [126]. All of those techniques changed the AdaBoost by presenting costs in the applied weight updating formula. Besides, changing the update rule for those techniques are different.

(iii) Algorithmic Structures with Decision Trees: The impact of the incorporation of costs on decision trees under an imbalanced area was introduced in [115]. Class confidence proportion decision tree (CCPDT) is a robust technique that sensitiveness to class distribution, which produces statistically remarkable regulations [127]. The authors start to declaring information gain (IG) to perform robust decision trees with the metric used in C4.5, in terms of confidence of a rule. This permit instantly demonstrates why Information Gain (like confidence) appears in rules, which are biased regarding the majority class. To defeat this bias, a novel measure that represents the basis of CCPDT is established and named class confidence proportion (CCP). A new and impressive top-down and bottom-up method is advanced to produce statistically significant regulations, which utilizes Fisher's exact test to prune branches of the tree that is not statistically significant. Another decision tree technique is Hellinger distance decision trees (HDDT), which combines the Hellinger distance as the tree splitting measure [128]. Properties of the HDDT in the data mining field is a skew insensitive decision tree splitting metric. Plus, a novel decision tree method that utilizes Hellinger distance as the splitting criterion is offered by [129]. The algorithm of HDDT can be found in algorithm 4.

Algorithm 3 AdaBoost algorithm

1. Input: Training set $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$, where $x_i \in X$, $y_i \in Y = \{-1, +1\}$,
Base-learner algorithm, Number of iterations M.
2. Initialization: Weight the training samples $w_i^1 = \frac{1}{n}$, $i = 1, 2, ..., n$.
3. Iteration: For $m = 1, 2, ..., M$
(i) Use the Base-learner algorithm to fit a classifier $f^m(x)$ to the training data using weights w_i^m.
(ii) Calculate the training error $err^{(m)}$ of the classifier f^m:

$$err^{(m)} = \frac{\sum_1^n w_i^m.(f^m(x_i) \neq y_i)}{\sum_1^n w_i^m}$$

(iii) Calculate the weight α^m for the classifier f^m:

$$\alpha^m = \log \frac{1 - err^{(m)}}{err^{(m)}}$$

(iv) Update the weight of the training samples:

$$w_i^{m+1} = w_i^m.exp(\alpha^m.(f^m(x_i) \neq y_i)) \quad i = 1, 2, ..., n$$

4. Output: The final ensemble model:

$$F(x) = sign(\sum_{m=1}^{M} \alpha^m f^m(x))$$

Algorithm 4 Hellinger distance decision trees

Require: : Training set T, Cut-off size C, Tree node n
1: **if** $| T | < C$ **then**
2: **return**
3: **end if**
4: $n \leftarrow argmax_f$Calc_Binary_Hellinger(T,f)[128]
5: **for** each value **v** of b **do**
6: create n', a child of n
7: $HDDT(T_{xb=v}, C, n')$
8: **end for**

(iv) Algorithmic Structures with Neural Networks: Based on neural networks, the possibility of performing different classifiers into cost-sensitive has been regarded by [130–132]. In [133], the authors proposed a cost-sensitive multilayer perceptron (CSMLP) technique for asymmetrical learning of MLPs by a back-propagation weight update rule. An impressive wrapper approach introduced in [134], i.e., intrinsic structure parameters and misclassification costs, which is a combination of the evaluation measure straight into the objective function of cost-sensitive neural network by concurrently optimizing the excellent pair of feature subset to enhance the performance of classification. The algorithm of PSO can be expressed as in algorithm 5. The empirical study for imbalance issues in the area of the RBF neural network trained with the back-propagation algorithm is considered in [131]. Moreover, a cost function in the training process to recompense imbalance class and one strategy to decrease the influence of the cost function in the data probability distribution is suggested in [131]. Ensembles learning is also regarded in the cost-sensitive framework to deal with the imbalance problems. Many ensemble techniques were perfectly adapted to contain costs over the learning process.

Algorithm 5 PSO used to optimize SVM parameters

begin
$t \rightarrow 0$ //iteration number
Initialize $Z(t)$ //$Z(t)$:swarm for iteration t
Evaluate $f(Z(t))$
while (not termination condition) **do**
begin
$t \rightarrow t + 1$
Update velocity $v(t)$ and position of each particle $z(t)$
if $v(t) > v_{max}$, $v(t) = v_{max}$ **end**
if $v(t) < -v_{max}$, $v(t) = -v_{max}$ **end**
Evaluate $f(Z(t))$
Update $\widetilde{z}$ if the new position is better than previous $\widetilde{z}$
Update z if the new position is better than previous z
end
end

4.2.2. Algorithmic structures for regression

Some tasks have referred to the issue of imbalanced area for regression tasks by modifying the dividing criteria of regression trees [135,136].

The utility-based approach is utilized in regression issues to handle the problem of imbalance area. A regression rules ensemble system that includes a measure based on utility as a preference criterion in the generation of approaches proposed as utility-based Rules (ubaRules) [137]. This method aims to admit accurate and understandable predictions by using the evaluation measures for target applications.

Presenting costs at a post-processing level in regression has been introduced by [138,139]. In this area, the problem is still under-investigated with few restricted strategies. Same as classification, still no improvement was constructed for evaluating these strategies in imbalanced areas. However, an efficient method proposed that named reframing [140,141]. The aim of this method was to displays that the cost-sensitive applications in regression could effectively conduct by probabilistic reframing and using enriched regression techniques in the system of a two-parameter normal conditional distribution. To perform this aim, the author's equivalence enrichment techniques to other methods for conditional density estimation concerning estimation quality and efficiency.

4.3. Hybrid methods

A hybrid technique for tackling class imbalance is consists of a couple of methods, which can be used for dealing with imbalance issues or applied several approaches for a particular part of the general solution. These techniques have a burden on the confidence that the variations in the methods rightly complement each other. Hybrid techniques mix strategies of various types and attempt to get the advantage of their best characteristics. The main disadvantages of these approaches are:

(1) Perfectly balanced data may not be optimal,

(2) It is difficult to determine the appropriate number of over/under-sample to utilize in this system.

4.3.1. Hybrid methods for classification

(i) Re-sampling with learning methods: The ensemble of classifiers in machine learning is well-known to improve the accuracy of individual classifiers by mixing some of them. But

none of those learning methods cannot lonely fix the class imbalance issue. To deal with that issue, we need to design ensemble learning algorithms [5].

The initial hybrid techniques were proposed by [44,142]. The motivation for those papers is to describe that which classifier is mixed. Also, in which way we can mix them. The mixture-of-experts method assumes that mixing several re-sampling techniques with the mixing of the expert framework is an effective presentation in the tuning problem [44,142]. A new hybrid technique is a combination of boosting and data sampling, which is named RUSBoost [143]. This technique tackles two overall kinds of re-sampling techniques: (1) Over-sample the minority samples to achieve a size almost similar to the majority samples, (2) Under-sample the majority instance to achieve a size almost similar to the minority samples. They proposed a technique for mixing several classifiers that is under and over-sample the dataset at various rates in a combination of a professional's framework. The authors claimed, it was unknown that under-sampling was more efficient than over-sampling and which rate of under-sampling or over-sampling should be studied.

An ensemble learning technique on multi-class imbalance sample sets is proposed in [144]. This technique produces one-versus-others classifiers that can learn over multi-class samples under the skewed normal distribution of the training samples. Another ensemble learning technique is ensemble knowledge for imbalance sample sets (eKISS), which mixes the rules of the basic classifiers to produce new classifiers for the last decision-making. Recently, more complicated techniques used in [145] as a dynamic classifier ensemble technique for imbalanced data (DCEID). This technique combines ensemble with cost-sensitive learning and introduces a dynamic classifier ensemble method for imbalanced datasets.

In [146], the authors proposed other views corresponding to re-sampling and the mixture of various learners. The main problem in the imbalance domain is a bad performance in the minority class. Therefore, first, the authors present a well-organized investigation of the different methods that have decided to manage this issue. Next, it performs an experimental comparison of these methods with an introduced blend of expert agents. It confirms that this strategy can be a more efficient explication to that issue. Finally, they proposed an agent-based knowledge discovery (ABKD) technique. This method utilizes three classifiers, i.e., C4.5, 5NN, and Naive Bayes, on some datasets to increases the reliability and speed of learning systems. In [147], the new strategy considered using C4.5, naive Bayesian (NB), and back-propagation (BP) for processing the same partitioned numerical data. So, one hybrid meta-learning method and three classification algorithms are justified for the new technique. By applying the strategy of the procedure of tested data partitions, and employing a simple cost model to study the classifiers, the most desirable mixture of classifiers can choose for an organization.

One-class classification (OCC) [148] attempts to recognize objects of a particular class between all objects, even though there exist variants of one-class classifiers where counterexamples are employed to more correction the classification boundary. This technique is much more complicated than traditional classification issues, which attempt to distinguish between two or more classes with the training set consisting of all samples. The major problem is: how to set the threshold (maximum of reconstruction error on the training set)? If the training data polluted by noise, then considering a negative sample as a part of the positive class could be a result. So, this threshold requires to be tight. With all these possible disadvantages, [120] proposed recognition-based learning algorithms to present a suitable prediction performance in most scopes. Scholars solved those problems with using of an auto-encoder (or auto-associator) [149,150] and one-class SVMs [151–154].

(ii) Integration of Kernel Methods with Sampling Techniques: In [101], the authors used SVM with soft margins as a basic algorithm to fix the skewed vector space issues with a boosting algorithm. Results can be damaged if misclassification errors are considered only one class. This problem happened by the reality that, classes with a smaller instance in the training set have an inferior prior probability and an inferior error cost. Particularly in imbalanced areas, most normal algorithms will desire to learn how to predict the majority samples.

Another technique that mixed pre-processing methods with boosting and bagging were proposed by [142], and at the same time, it uses an ensemble of ensembles. This system ignored most of the majority samples, which is the main disadvantage of the proposed technique. Two algorithms are employed to defeat this difficulty. The first algorithm is easy-ensemble samples to find some subsets from the majority instance and mixes the outputs of those learners. The other is balance-cascade [155] that train the learners consecutively in every step. In the balance-cascade method, the majority class samples that are correctly classified by the currently trained learners are eliminated (from more evidence). In both algorithms (easy-ensemble and balance-cascade), it is better to use the majority samples because multiple subsets include more information than a single one.

A remarkable technique to build the classifier from the imbalanced area is introduced by [116,156], which mixes SMOTE and Biased-SVM. For these techniques, it admits a good sense of growing the sensitivity of a classifier to the minority samples by utilizing the SMOTE technique in support vectors. There are two different over-sampling approaches to tackle with the support vectors that over-samples with its neighbors from the k-nearest neighbors. Two different strategies for over-sampling algorithms are employed, which use not only support vectors but also utilizes whole minority samples.

(iii) Kernel Modification Techniques: A kernel-boundary-alignment algorithm technique is offered in [157], which estimates the training imbalance data as prior information to boost SVMs to increase prediction accuracy. The authors have shown a simple example that SVM may suffer from a high occurrence of false negatives rate when the training samples of the target class are remarkably added numbers by the training samples of the non-target class.

(iv) Active Learning Techniques: Researchers attempt to present a generic model for the evolution of genetic programming (GP) classifiers on imbalanced datasets via a mixed approach of the cost function [158] and class imbalanced stochastic sampling. But, genetic algorithms (GA) applied over-sampling to increase the ratio of the positive sample and then use clustering on over-sampled training datasets as a data reduction technique (deleting the waste or noisy instances) for both classes [159,160]. One of the easiest and reliable ensemble algorithms is enhancing ensembles for highly imbalanced datasets by evolutionary under-sampling (EUSBoost) [161], which mixes random under-sampling with the Boosting algorithm.

However, some other methods have been developed that mix some of the earlier techniques [162–165]. For example, [166] mixed SMOTE and complementary neural network (CMTNN) to deal with the issue of classification of imbalanced datasets. The mixture of approaches also can be used for ensembles [100,167,168]. An Ensemble of Under-Sampled SVMs (EUS-SVMs) introduced in [99], which constructs several training sets by sampling instances from the majority class and mixing them with the minority class samples. Every training set is employed to train the SVM classifier. By taken the ensemble method, EUS-SVMs may not be just making up for the sampling relationship of the under-sampling technique but also performed a sensible time complexity associated with the over-sampling technique. EnSVM [100] is

another same method that offered a technique, which utilizes an ensemble of SVM classifiers. This method unified a re-balancing technique that blends over and under-sampling.

In [5], the authors proposed the ensemble of classifiers to raise the performance of classifiers in the imbalance domain by mixing some approaches. Several methods proposed for handling imbalance learning according to active learning [169–171]. These techniques utilize an effective active learning strategy to choose the advisory instances from the training set in the beginning steps of the learning process.

In [171], the authors proposed a new method for class imbalanced learning issues, which is named virtual instances re-sampling technique using active learning (VIRTUAL). This technique mixes over-sampling and active learning to design an adaptive method of minority samples. Different from conventional re-sampling techniques that need pre-processing of the data, Virtual creates synthetic samples for the minority class through the training process. Hence, it eliminates the requirement for an additional pre-processing step. In [172], the authors developed the classification performance of the support vector machine and provide an approach according to active learning SMOTE to classify the imbalanced datasets. Some attempts also have been constructed for integrating active learning with other classifiers. An active learning technique for the imbalanced dataset utilized a stochastic sensitivity measure (ST-SM) of radial basis function neural network (RBFNN) [173].

4.3.2. Hybrid methods for regression

Usually, we use a single-layer logistic regression as a mixture technique. So, we can efficiently utilize this idea to fine-tune the outcome sets for archiving the best outcomes in our problem [174]. The main benefit of this method is using several techniques. Therefore, it can present a useful element of all the possible techniques. However, scalability can be problematic, but accuracy in the new model presents huge progress.

5. Experimental outcomes

In this section, we perform the empirical comparison of the algorithms that we have analyzed. The goal is to investigate which one of the approaches can better manage imbalanced datasets with different imbalance ratios to show which one is the most robust method. Also, we want to investigate the improvement of each dataset based on different techniques.

We conducted experiments on 10 selective algorithms and 32 imbalanced datasets with an imbalance ratio (IR) from 1 to 129.44 and these binary datasets are downloaded from KEEL imbalanced datasets [104] and UCI machine learning repository [175]. The details of the imbalanced datasets are provided in Table 9 and 5-fold cross-validation is utilized for all datasets. The bootstrap method [176] with 95% confidence interval is applied to quantify the outcomes statistically. All the experiments are implemented in a MATLAB 2018a environment on a PC with an Intel(R) Core i5 processor (3.30 GHz) and 12 GB RAM. The area under the receiver operating characteristic curve (AUC) [177] is utilized to validate the classification performance of the algorithms on imbalanced datasets. A larger AUC illustrates better performance.

5.1. AUC (%) with standard deviation (CD) of the compared algorithms on imbalanced datasets

The AUC (%) with standard deviation (CD) is utilized to evaluate all algorithms. We compared the best IFTSVM-ID [178] with the best F-ART-IFTSVM [7], DLSSVM-CIL, 1-NN, AdaBoost, EasyEnsemble, SVM-RUS, SVM-OSS, SVM-SMOTE, and EFSVM. We summarize the results of the experimental analysis for each dataset in Table 10. One can see that the highest results on each dataset are highlighted in bold.

From these outcomes, for 23 datasets, IFTSVM-ID technique are almost better than those given by other imbalanced learning techniques. For six datasets, the best F-ART-IFTSVM is better than our method. For the Vehicle 1 dataset, the DLSSVM-CIL method archived better results compared to the other algorithms. Also, for the Vowel dataset, the performance of 1-NN is better than all methods.

The methods with the most robust behavior are IFTSVM-ID, F.ART-IFTSVM, and DLSSVM-CIL. Among them, in terms of average AUC, IFTSVM-ID stands out obtaining better results. The reason is that the IFTSVM technique combines the idea of an intuitionistic fuzzy number with a twin support vector machine. This method not only reduces the influence of noises but also distinguishes the noises from the support vectors. Further, this modification can minimize a newly formulated structural risk and improve performance. The main weakness of this technique is its complexity, which is undeniable.

On the other hand, AdaBoost, 1-NN, and SVM-OSS do not perform very well. I must point out that, although the AdaBoost algorithm can directly process imbalanced data, the algorithm concentrates more on the misclassified samples than samples of the minority class. In addition, it may generate many redundant or useless weak classifiers, increasing the processing overhead and leading to performance reduction. The kNN technique is sensitive to the majority samples and thus performs poorly for imbalanced datasets. So, the performance of k-NN classifiers will be significantly impacted by this problem.

5.2. Statistical test outcomes of the best WIFTSVM-CIL and other imbalanced learning techniques

In the imbalanced domain, the Friedman test [179] is usually used to evaluate the performance of the suggested technique and other available class imbalance learning techniques. First of all in the Friedman test, we rated the algorithms on every dataset separately, i.e., the first rank is for an algorithm with higher performance, the next highest is ranked 2, etc. Then, we considered the number of imbalanced datasets (resp. algorithms) as n (resp. k). Also, the rank of the jth algorithm on the ith dataset is r_i^j. The average rank of algorithms $R_j = \frac{1}{n}\sum_{i=1}^{n} r_i^j$ is evaluated by this test. The ranks are equal under the null hypothesis, it means all the algorithms are equivalent. The Friedman test

$$\chi_F^2 = \frac{12n}{k(k+1)}\left[\sum_{j=1}^{k} R_j^2 - \frac{k(k+1)^2}{4}\right], \tag{17}$$

is established on a χ_F^2 distribution with degrees of freedom, i.e., $(k-1)$ and $(k-1)(n-1)$ when k and n are adequately large. The Friedman χ_F^2 generated an advisable statistic [157]

$$F_F = \frac{(n-1)\chi_F^2}{n(k-1)-\chi_F^2}. \tag{18}$$

To evaluate the outcomes of the 10 class imbalance learning techniques, I identify the rank of every algorithm for every dataset individually (see Table 11).

Then, the Friedman test is computed from Table 11 under the null hypothesis when $n = 32$ and $k = 10$:

$$\chi_F^2 = \frac{12 \times 32}{10 \times (10+1)}\left[(5.34^2 + 5.64^2 + 7.30^2 + 6.31^2 + 6.09^2 + 8.92^2 + 8.42^2 + 3.53^2 + 2.16^2 + 1.28^2) - \frac{10 \times (10+1)^2}{4}\right]$$

$$= \frac{384}{110}\left[359.7471 - \frac{1210}{4}\right] = 199.79,$$

Table 9
Details of the imbalanced datasets.

No	dataset	Positive	Negative	Instance	Dimension	Im. Ratio
1	Wisconsin	239	444	683	9	1.86
2	Pima	268	500	768	8	1.87
3	Yeast 1	429	1,055	1,484	8	2.46
4	Vehicle 2	218	628	864	18	2.88
5	Vehicle 1	217	629	864	18	2.52
6	Segment	329	1979	2,308	18	6.02
7	Yeast 3	163	1,321	1,484	8	8.10
8	Page blocks	560	4,913	5,473	10	164
9	Yeast 2–vs–4	463	51	514	8	9.08
10	Ecoli 0–2–3–4–vs–5	20	182	202	7	9.10
11	Yeast 0–3–5–9–vs–7–8	50	456	506	8	9.12
12	Yeast 0–2–5–6–vs–3–7–8–9	99	905	1,004	8	9.14
13	Ecoli 0–4–6–vs–5	20	183	203	7	9.15
14	Ecoli 0–1–vs–2–3–5	24	220	244	7	9.17
15	Yeast 0–5–6–7–9–vs–4	51	477	528	8	9.35
16	Vowel	90	898	988	10	9.98
17	Ecoli 0–6–7–vs–5	20	200	220	6	10
18	Led7digit 0–2–4–5–6–7–8–9–vs–1	37	406	443	7	10.97
19	Ecoli 0–1–vs–5	20	220	240	6	11
20	Ecoli 0–1–4–7–vs–5–6	25	307	332	6	12.28
21	Ecoli 0–1–4–6–vs–5	20	260	280	6	13
22	Glass 4	201	13	214	9	15.47
23	Ecoli 4	20	316	336	7	15.80
24	Yeast 1–4–5–8–Vs–7	30	663	693	7	22.10
25	Glass 5	9	205	214	9	22.78
26	Yeast 2–Vs–8	20	462	482	8	23.10
27	Yeast 4	51	1,433	1,484	8	28.10
28	Yeast 1–2–8–9–Vs–7	30	917	947	8	30.57
29	Yeast 5	44	1,440	1,484	7	32.73
30	Ecoli 0–1–3–7–vs–2–6	7	274	281	7	39.14
31	Yeast 6	35	1,449	1,484	8	41.40
32	Abalone 19	32	4,142	4,174	7	129.44

Table 10
AUC (%) with standard deviation (CD) and ranks of the compared algorithms on imbalanced datasets.

Dataset	EFSVM	SVM-SMOTE	SVM-OSS	SVM-RUS	EasyEnsemble	AdaBoost	1-NN	DLSSVM-CIL	B-F.ART-IFTSVM	B-IFTSVM-ID
Wisconsin	96.84 ± 0.24	96.64 ± 0.86	96.61 ± 0.58	96.35 ± 1.32	96.58 ± 0.81	94.43 ± 2.01	94.40 ± 1.12	97.18 ± 1.62	98.48 ± 0.26	**98.57 ± 0.12**
Pima	71.14 ± 2.82	72.31 ± 3.49	70.53 ± 3.98	72.46 ± 1.02	75.22 ± 1.68	72.09 ± 2.53	65.47 ± 2.19	76.56 ± 2.11	83.77 ± 2.35	**84.36 ± 1.99**
Yeast 1	70.32 ± 3.95	70.18 ± 2.26	68.55 ± 2.03	69.68 ± 3.06	73.46 ± 3.74	67.39 ± 3.23	64.42 ± 2.25	77.96 ± 3.45	78.10 ± 4.06	**79.08 ± 1.42**
Vehicle 2	74.29 ± 4.84	72.40 ± 3.88	67.31 ± 4.90	68.17 ± 4.96	97.64 ± 1.21	96.83 ± 0.85	91.49 ± 2.14	98.10 ± 2.19	98.08 ± 0.35	**98.39 ± 0.40**
Vehicle 1	67.83 ± 3.86	65.35 ± 4.10	68.59 ± 4.22	68.72 ± 3.54	79.45 ± 3.59	67.79 ± 3.58	59.81 ± 1.78	**84.74 ± 4.01**	83.64 ± 1.85	83.87 ± 0.99
Segment	83.61 ± 2.01	85.56 ± 2.36	79.28 ± 3.22	83.68 ± 2.98	99.62 ± 0.29	99.53 ± 0.70	99.01 ± 1.35	98.53 ± 0.98	99.86 ± 0.14	**99.94 ± 0.09**
Yeast 3	91.29 ± 1.88	92.47 ± 1.54	89.65 ± 2.06	90.12 ± 1.94	93.03 ± 3.14	87.48 ± 2.18	80.59 ± 2.22	94.72 ± 1.73	95.70 ± 1.29	**96.08 ± 1.15**
Page blocks	66.14 ± 5.85	65.71 ± 10.82	62.01 ± 4.53	65.64 ± 4.26	96.16 ± 1.01	89.63 ± 0.86	88.49 ± 2.39	95.53 ± 2.57	**96.94 ± 1.44**	96.90 ± 0.99
Yeast 2–vs–4	90.52 ± 3.11	88.98 ± 1.14	89.81 ± 3.46	88.81 ± 3.32	93.15 ± 2.01	84.49 ± 4.01	86.11 ± 5.41	85.71 ± 3.89	93.11 ± 4.44	**94.80 ± 4.34**
Ecoli 0–2–3–4–vs–5	93.38 ± 8.23	91.48 ± 9.14	92.16 ± 9.43	95.68 ± 4.81	91.59 ± 8.64	91.38 ± 10.19	87.35 ± 11.36	99.46 ± 2.11	99.86 ± 0.23	**99.87 ± 0.22**
Yeast 0–3–5–9–vs–7–-8	72.18 ± 6.98	73.71 ± 4.01	66.44 ± 5.49	72.38 ± 3.89	75.64 ± 4.52	66.29 ± 4.15	68.42 ± 4.99	79.04 ± 5.41	**80.60 ± 8.81**	80.27 ± 5.03
Yeast 0–2–5–6–vs–3–7–8–9	81.38 ± 4.22	81.44 ± 4.83	81.23 ± 4.01	79.44 ± 4.39	80.47 ± 4.58	72.40 ± 3.14	77.12 ± 2.76	85.52 ± 4.24	86.57 ± 9.45	**86.75 ± 4.23**
Ecoli 0–4–6–vs–5	92.01 ± 9.89	92.18 ± 9.13	91.42 ± 10.33	90.73 ± 8.01	89.56 ± 5.87	87.68 ± 12.80	87.68 ± 12.57	96.80 ± 8.41	**98.31 ± 2.10**	**98.31 ± 0.84**
Ecoli 0–1–vs–2–3–5	92.66 ± 4.31	89.12 ± 7.98	88.26 ± 10.19	88.34 ± 7.29	87.84 ± 6.72	80.16 ± 10.03	80.48 ± 12.38	96.41 ± 7.42	95.03 ± 6.26	**97.85 ± 1.92**
Yeast 0–5–6–7–9–vs–4	82.49 ± 4.86	81.73 ± 4.42	81.56 ± 5.85	82.27 ± 4.25	80.20 ± 5.24	68.65 ± 5.82	71.34 ± 6.16	78.46 ± 5.54	84.59 ± 7.36	**85.70 ± 5.29**
Vowel	88.74 ± 2.99	90.39 ± 3.45	87.90 ± 5.21	89.36 ± 3.45	96.72 ± 2.54	94.68 ± 4.86	**100.00 ± 0.00**	96.57 ± 3.79	97.89 ± 0.88	98.93 ± 0.12
Ecoli 0–6–7–vs–5	92.48 ± 5.77	90.01 ± 5.55	90.01 ± 6.48	88.68 ± 5.35	86.14 ± 5.68	77.81 ± 4.33	85.04 ± 4.76	93.70 ± 4.68	94.48 ± 7.07	**95.02 ± 4.49**
Led7digit 0–2–4–5–6–7–8–9–vs–1	91.21 ± 5.92	91.46 ± 4.84	88.76 ± 6.46	89.91 ± 4.22	87.98 ± 7.78	87.15 ± 6.49	62.79 ± 7.56	92.62 ± 6.51	91.48 ± 8.07	**94.83 ± 2.98**
Ecoli 0–1–vs–5	94.38 ± 4.83	92.27 ± 6.01	93.14 ± 7.12	92.44 ± 4.37	87.70 ± 10.84	85.12 ± 13.45	88.43 ± 12.11	91.67 ± 5.85	**96.54 ± 3.75**	96.47 ± 2.90
Ecoli 0–1–4–7–vs–5–6	91.84 ± 4.14	91.79 ± 3.76	90.04 ± 5.48	90.11 ± 3.20	90.34 ± 2.28	88.37 ± 9.49	88.43 ± 7.51	92.74 ± 2.46	**95.16 ± 3.48**	94.67 ± 4.60
Ecoli 0–1–4–6–vs–5	91.50 ± 9.50	91.64 ± 9.99	92.68 ± 4.48	90.69 ± 9.83	89.76 ± 6.62	79.01 ± 14.18	87.48 ± 10.33	92.01 ± 7.83	**94.90 ± 5.27**	93.31 ± 7.38
Glass 4	91.34 ± 8.01	91.22 ± 9.84	88.94 ± 11.68	92.18 ± 3.49	86.32 ± 11.48	87.47 ± 10.12	93.89 ± 8.87	94.68 ± 3.92	94.27 ± 4.77	**95.96 ± 2.51**
Ecoli 4	97.05 ± 1.33	94.71 ± 4.89	93.84 ± 6.21	94.64 ± 4.13	90.56 ± 5.03	84.39 ± 13.70	88.78 ± 4.39	97.95 ± 2.68	98.38 ± 0.93	**98.45 ± 0.80**
Yeast 1–4–5–8–Vs–7	70.62 ± 5.84	67.42 ± 6.92	63.38 ± 4.50	65.72 ± 7.14	65.59 ± 5.12	52.13 ± 0.16	57.18 ± 5.75	71.85 ± 6.46	**75.21 ± 10.58**	72.81 ± 7.36
Glass 5	93.38 ± 4.31	92.45 ± 8.78	81.25 ± 11.65	89.35 ± 2.99	87.43 ± 10.19	80.41 ± 18.64	85.68 ± 15.34	94.97 ± 4.47	93.80 ± 5.96	**97.34 ± 2.41**
Yeast 2–Vs–8	82.89 ± 5.16	78.56 ± 10.19	77.24 ± 7.14	79.28 ± 8.09	81.34 ± 9.75	75.44 ± 7.78	75.36 ± 7.60	87.23 ± 5.28	84.77 ± 10.16	**88.97 ± 7.40**
Yeast 4	85.27 ± 2.18	85.12 ± 0.98	82.53 ± 3.22	84.58 ± 2.73	82.61 ± 3.16	59.40 ± 4.88	67.59 ± 6.69	87.08 ± 3.80	88.08 ± 5.53	**88.28 ± 4.68**
Yeast 1–2–8–9–Vs–7	70.14 ± 7.59	73.68 ± 5.15	66.45 ± 6.89	75.71 ± 4.21	71.98 ± 10.88	60.37 ± 7.76	56.49 ± 4.16	76.74 ± 5.18	75.93 ± 1.51	**80.03 ± 9.67**
Yeast 5	96.48 ± 0.98	96.52 ± 0.67	95.76 ± 1.21	96.27 ± 1.32	95.91 ± 1.39	86.42 ± 6.58	84.53 ± 2.98	86.91 ± 1.79	97.34 ± 0.34	**98.78 ± 0.24**
Ecoli 0–1–3–7–vs–2–6	95.67 ± 9.70	89.56 ± 17.41	85.78 ± 15.54	85.40 ± 16.11	77.43 ± 14.60	65.79 ± 17.19	85.59 ± 15.48	98.48 ± 8.78	**100 ± 0.00**	**100 ± 0.00**
Yeast 6	89.26 ± 3.80	91.34 ± 4.82	87.20 ± 5.75	89.48 ± 5.32	85.28 ± 4.71	73.66 ± 10.12	78.84 ± 8.82	92.12 ± 3.19	92.82 ± 6.32	**94.86 ± 2.69**
Abalone 19	57.08 ± 9.18	66.13 ± 7.28	54.38 ± 8.01	67.36 ± 6.15	71.46 ± 12.89	50.51 ± 0.95	52.19 ± 3.15	78.39 ± 4.77	68.23 ± 6.09	**81.47 ± 4.56**
AUC Mean	84.54	84.17	81.65	83.55	85.76	78.57	79.39	89.70	91.00	**92.22**

Table 11
Ranks of the compared algorithms on imbalanced datasets.

Dataset	EFSVM	SVM-SMOTE	SVM-OSS	SVM-RUS	EasyEnsemble	AdaBoost	1-NN	DLSSVM-CIL	B-F.ART-IFTSVM	B-IFTSVM-ID
Wisconsin	4	5	6	8	7	9	10	3	2	1
Pima	8	6	9	5	4	7	10	3	2	1
Yeast 1	5	6	8	7	4	9	10	3	2	1
Vehicle 2	7	8	10	9	4	5	6	2	3	1
Vehicle 1	7	9	6	5	4	8	10	1	3	2
Segment	9	7	10	8	3	4	5	6	2	1
Yeast 3	6	5	8	7	4	9	10	3	2	1
Page blocks	7	8	10	9	3	5	6	4	1	2
Yeast 2–vs–4	4	6	5	7	2	10	8	9	3	1
Ecoli 0–2–3–4–vs–5	5	8	6	4	7	9	10	3	2	1
Yeast 0–3–5–9–vs–7–8	7	5	9	6	4	10	8	3	1	2
Yeast 0–2–5–6–vs–3–7–8–9	5	4	6	8	7	10	9	3	2	1
Ecoli 0–4–6–vs–5	5	4	6	7	8	9.5	9.5	3	1.5	1.5
Ecoli 0–1–vs–2–3–5	4	5	7	6	8	10	9	2	3	1
Yeast 0–5–6–7–9–vs–4	3	5	6	4	7	10	9	8	2	1
Vowel	9	7	10	8	4	6	1	5	3	2
Ecoli 0–6–7–vs–5	4	5.5	5.5	7	8	10	9	3	2	1
Led7digit 0–2–4–5–6–7–8–9–vs–1	5	4	7	6	8	9	10	2	3	1
Ecoli 0–1–vs–5	3	6	4	5	9	10	8	7	1	2
Ecoli 0–1–4–7–vs–5–6	4	5	8	7	6	10	9	3	1	2
Ecoli 0–1–4–6–vs–5	6	5	3	7	8	10	9	4	1	2
Glass 4	6	7	8	5	10	9	4	2	3	1
Ecoli 4	4	5	7	6	8	10	9	3	2	1
Yeast 1–4–5–8–Vs–7	4	5	8	6	7	10	9	3	1	2
Glass 5	4	5	9	6	7	10	8	2	3	1
Yeast 2–Vs–8	4	7	8	6	5	9	10	2	3	1
Yeast 4	4	5	8	6	7	10	9	3	2	1
Yeast 1–2–8–9–Vs–7	7	5	8	4	6	9	10	2	3	1
Yeast 5	4	3	7	5	6	9	10	8	2	1
Ecoli 0–1–3–7–vs–2–6	4	5	6	8	9	10	7	3	1.5	1.5
Yeast 6	6	4	7	5	8	10	9	3	2	1
Abalone 19	7	6	8	5	3	10	9	2	4	1
Ave. Rank	5.34	5.64	7.30	6.31	6.09	8.92	8.42	3.53	2.16	**1.28**
Difference	4.06	4.36	6.02	5.03	4.81	7.64	7.14	2.25	0.88	N/A

and

$$F_F = \frac{(32-1)\times 199.79}{32\times(10-1)-199.79} = \frac{6193.49}{88.21} = 70.21,$$

F_F is calculated from the F-distribution with $(k-1) = (10-1) = 9$ and degrees of freedom $(k-1)(n-1) = (10-1)(32-1) = 279$. The critical value at $\alpha = 0.05$ of $F(9, 279)$ is 1.914, so we can reject the null hypothesis and claim that the considered algorithms are not equivalent at $\alpha = 0.05$ because $F_F = 70.21$ is much bigger than 1.914.

Due to rejected the null-hypothesis, we utilize the Bonferroni–Dunn test [180] to evaluate the imbalanced-learning algorithms. In this test, if the average ranks are different by at least the critical difference (CD), we can say the chosen algorithm is significantly different from another algorithm. We compute the critical difference, which defined in the following equation:

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6n}}. \tag{19}$$

where q_α (critical values) are taken from [181]. We can compute the critical difference as $CD = 2.773\sqrt{\frac{10(10+1)}{6\times 32}} = 2.10$.

Based on Table 11 one can see the first rank belongs to the IFTSVM-ID, and the worst rank belongs to the AdaBoost technique. The difference between the average ranks of the best IFTSVM-ID and DLSSVM-CIL (resp. 1-NN, AdaBoost, EasyEnsemble, SVM-RUS, SVM-OSS, SVM-SMOTE, and EFSVM) is $3.53 - 1.28 = 2.25$ (resp. $8.42 - 1.28 = 7.14$, $8.92 - 1.28 = 7.64$, $6.09 - 1.28 = 4.81$, $6.31 - 1.28 = 5.03$, $7.30 - 1.28 = 6.02$, $5.64 - 1.28 = 4.36$, and $5.34 - 1.25 = 4.06$), which is greater than the $CD = 2.10$. Therefore, there is a significant difference between the best IFTSVM-ID and DLSSVM-CIL (resp. 1-NN, AdaBoost, EasyEnsemble, SVM-RUS, SVM-OSS, SVM-SMOTE, and EFSVM). But the difference between the average rank of the best IFTSVM-ID and the best F.ART-IFTSVM is $2.16 - 1.28 = 0.88$, which is less than $CD = 2.10$. So, we cannot conclude that the best IFTSVM-ID is significantly different from the best F.ART-IFTSVM. But almost for all datasets, the best IFTSVM-ID performs better than best F.ART-IFTSVM (see the outcomes in Table 10).

6. Software and open source for imbalanced classification

Two of the most famous software tools for data management purposes are KEEL [104] and WEKA [182]. The reason for their success is basically the inclusion of some of the most significant state-of-the-art algorithms and their ease of use. The major difference between KEEL and WEKA is that the first one provides a complete module for imbalanced classification, whereas the second one is limited to cost-sensitive and simple re-sampling.

The first tool for doing experiments is KEEL (short for Knowledge Extraction based on Evolutionary Learning). This is an open-source tool that can be used for different knowledge data discovery tasks. KEEL design experiments with different datasets and computational intelligence algorithms in order to assess the behavior of the algorithms. It contains a wide variety of classical knowledge extraction algorithms, preprocessing techniques (training set selection, feature selection, imputation methods for missing values, among others), computational intelligence based learning algorithms, hybrid models, statistical methodologies for contrasting experiments and so forth. It allows to perform a complete analysis of new computational intelligence proposals in comparison to existing ones. Keel tools have the following benefits:

(i) Data management,
(ii) Design of experiments,
(iii) Design of imbalanced experiments,
(iv) Statistical tests.

It focuses on imbalanced classification and comprises almost 167 benchmark problems under different scenarios:

(i) 22 datasets with an imbalance ratio between 1.5 and 9. We can consider them as low imbalanced problems.

(ii) 100 datasets with an imbalance ratio higher than 9. These can be divided into three different parts, depending on the research papers in which they have been used. We consider all of them to be highly imbalanced problems. They have an additional difficulty with the classification task.

(iii) 15 Multi class imbalanced problems. We use them to extend the studies in imbalanced classification when several classes are involved.

(iv) 30 Noisy and Borderline Examples. These are synthetic problems to analyze the behavior with both imbalance and noise.

For the imbalanced data, WEKA presents two solutions. First, to carry out a training data re-balancing. Second, perform a cost-sensitive classification. We selected the re-balancing procedure via instance filtering methods. Besides these, we can include the SMOTE preprocessing via the package manager. We have basically two ways to process cost-sensitive learning. The first one is via instance weighting using the "ClassBalancer" filtering. The second one is by importing a user derived cost-matrix. The cost-matrix is based on meta learning, which is comprise two parts. First part is to use "Cost-Sensitive Classifier" approach, and the second one is the "MetaCost" scheme.

(i) The goal of "ClassBalancer" is re-weighting the samples in each class to get a same total class weight.

(ii) We can use two additional methods to use "Cost-Sensitive Classifier": re-weighting training samples according to the total cost assigned to each class; or predicting the class with minimum expected misclassification cost (rather than the most likely class).

(iii) The study of the cost matrix in MetaCost is more intuitive. This classifier should generate similar results to one created by passing the base learner to bagging. The difference is that MetaCost generates a single cost-sensitive classifier of the base learner, giving the benefits of fast classification and interpretable output.

7. Summary of the results

In this section, we bring up several questions for future research directions in the imbalance domain. These critical questions follow straightly from the discussed methods to address some difficulties in the imbalanced datasets. Our aim for writing this survey is to discuss and mention open challenges for future research in this area. We are sure the following fundamental question should be considered strongly in both empirical and theoretical cases to understand the nature of imbalanced learning issues.

7.1. Data pre-processing techniques in class imbalanced learning

Data pre-processing techniques have consisted of strategies that pre-process the given imbalanced dataset and modify the data distribution to make standard algorithms for the users. We think the following questions have to be investigated very carefully:

(1) It is hard to connect the data distribution with the target loss function. That is why mapping the data distribution into an optimal new distribution is difficult. So, the question is; How to map the data distribution into the desirable new one?

(2) We have a lot of pre-processing techniques for dealing with the problem of imbalanced datasets. Finding the perfect method that can be used to handle all imbalanced datasets is problematic. So, how do we identify appropriate pre-processing techniques for handling imbalanced datasets?

(3) One way to do pre-processing for handling imbalanced data is by generating synthetic data. It is as good as production data and capable of improving data quality. But finding the perfect method to create synthetic data is hard. So, how to produce perfect new synthetic samples?

(4) For the generation of new synthetic samples, the target value of each synthetic sample was calculated as a weighted average of the target variable values of the two samples. So, it is tough to determine the value of the target variable in the synthetic samples. So, how to set the number of the target variable in the synthetic samples?

(5) Having large datasets reduces the chances of making inaccurate predictions. But instead, it causes finding weak patterns in data. However, the idea of storing and handling massive numbers of data produces challenges in the world of these intellectual algorithms. So, how do we deal with computational complexity in large-scale datasets?

(6) Noisy data is data with a large number of additional meaningless information in it called noise. Noisy data can negatively affect the results of any data analysis and skew conclusions if not handled properly. So, how to find a method to deal with noisy datasets?

To handle those problems I suggest the following solutions:

(*i*) Data Cleaning One of the most significant aspects of the data preprocessing phase is finding and fixing bad and inaccurate observations from your dataset in order to improve its quality. This technique refers to identifying incomplete, inaccurate, duplicated, irrelevant or null values in the data.

(*ii*) Deep generative models such as Variational Autoencoder (VAE) and Generative Adversarial Network (GAN) can generate synthetic data. VAE is an unsupervised method where the encoder compresses the original dataset into a more compact structure and transmits data to the decoder. Then the decoder generates an output which is a representation of the original dataset. The system is trained by optimizing the correlation between input and output data.

(*iii*) We can use speed-up techniques for handling computational complexity. One of those techniques is a Coordinate descent. This is an optimization algorithm that successively minimizes along coordinate directions to discover the minimum of a function. At each iteration, the algorithm detects a coordinate or coordinate block via a coordinate selection rule, then exactly or inexactly minimizes over the corresponding coordinate hyperplane while fixing all other coordinates or coordinate blocks. We can perform a line search along the coordinate direction at the current iterate to determine the appropriate step size.

(iv) One of the effective techniques to reduce the negative effect of noise is a fuzzy set. For example, in fuzzy SVM, we assign a fuzzy membership to each sample and reformulate the SVMs such that different input samples can make different contributions to the learning of the decision surface. This technique enhances the SVM in reducing the negative effect of outliers and noises in data points. Also, it is good enough for applications in which data points have un-modeled characteristics.

7.2. Algorithmic structures techniques in class imbalanced learning

Algorithmic structure approaches create new algorithms or modify existing algorithms to tackle imbalanced problems. The following questions need to be considered for future research in imbalance area:

(1) We have different techniques with different strategies to deal with noisy datasets. Finding the best techniques is challenging. So, how to compare methods that could handle noisy datasets?

(2) Classification problems having multiple classes with an imbalanced dataset show a different challenge than a binary classification problem. And finding the best multi-class method for algorithmic structures techniques is a hard task. So, how to deal with classification in multi-label datasets?

(3) Cost-sensitive learning takes the costs of prediction errors into account when training a machine learning model. Many approaches and methods developed and used for cost-sensitive learning can be adopted for imbalanced classification problems. So, how do we find the cost-sensitive method to deal with imbalance issues?

(4) How to use the existing technique with a different learning system?

(5) How to deal with the model if the target loss function is changed? The model must be re-learned or, is it necessary to introduce further modifications?

(6) How to deal with computational complexity in large-scale datasets?

I have some suggestions to solve the aforementioned problems:

(i) From binary learners to multi-class classifiers: the decomposition methods One-vs-All (OVA), also known as "One-against-all", is a relatively simple decomposition strategy.

(ii) Same as pre-processing technique, speed-up techniques can be suggested for handling computational complexity. The Coordinate descent and gradient descent are algorithms that are suitable for computational complexity.

(iii) Cost-sensitive learning for imbalanced classification concentrates on assigning different costs to the misclassification errors. Among all the classifiers, the induction of cost-sensitive decision trees has probably gained the most attention.

(iv) We can consider a simple unbiased estimator of any loss and get performance bounds for empirical utility maximization in the presence of data with noisy labels. Also, we can reduce risk minimization under noisy labels to classification with a weighted 0–1 loss. It means we can use a simple weighted surrogate loss for which we can get strong utility bounds.

7.3. Hybrid techniques in class imbalanced learning

Hybrid methods combine approaches of different types to take advantage of their best characteristics for handling imbalanced distributions. The following questions need to consider for future research in imbalance learning domains:

(1) How to determine the right number of over/under-sample to apply for a modification of datasets?

(2) How to estimate an effective strategy for parameters and different kernels?

(3) How to deal with computational complexity in large-scale datasets?

(4) How to deal with the noisy datasets?

(5) How to deal with classification in multi-label datasets?

We can use the suggestions of pre-processing and algorithmic structure to solve the problems of hybrid techniques. We believe that all of the mentioned questions are critical, not just for theoretical research promotion but also for several practical application fields.

8. Conclusion

In this survey, we discussed some techniques and critical issues in the imbalance learning area. Several methods are considered for evaluation and predictive modeling in an imbalance area. We tried to concentrate on both regression and classification tasks. Besides, we proposed a new taxonomy for class imbalanced learning techniques where each method can be described depending on the same strategy in which it is based. We classified existing imbalance techniques into three parts: (1) Data preprocessing, (2) Algorithmic structures, and (3) Hybrid techniques. Finally, some issues which are highly associated with imbalanced data distributions are described and discovered the relationship between those issues with imbalanced datasets. Hopefully, our debates on the substantial nature of the imbalanced learning issue will be used in feature research to evaluate this area.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Salim Rezvani reports was provided by Department of Computer Science, Ryerson University, Toronto,Canada.

Data availability

Data will be made available on request.

References

[1] R. Akbani, S. Kwek, N. Japkowicz, Applying support vector machines to imbalanced datasets, in: Proceedings of the 15th European Conference on Machine Learning, 2004, pp. 39–50.
[2] H. Haibo, M. Yunqian, Class imbalance learning methods for support vector machines, in: Imbalanced Learning: Foundations, Algorithms, and Applications, IEEE, 2013, pp. 83–99, http://dx.doi.org/10.1002/9781118646106.ch5, URL https://ieeexplore.ieee.org/document/6542402.
[3] H. Haibo, B. Yang, A.G. Edwardo, L. hutao, Adasyn: Adaptive synthetic sampling approach for imbalanced learning, Adv. Knowl. Discov. Data Min. (2008) 1–26.
[4] H. Hiabo, E.A. Garcia, Learning from imbalanced data, IEEE Trans. Knowl. Data Eng. 21 (9) (2009) 1263–1284.
[5] M. Galar, A. Fernandez, E. Barrenechea, F. Herrera, A review on ensembles for the class imbalance problem: Bagging, boosting, and hybrid-based approaches, IEEE Trans. Syst., Man, Cybern., Part C: Appl. Rev. 42 (2012) 463–484.
[6] P. Branco, L. Torgo, R.P. Ribeiro, A survey of predictive modelling under imbalanced distributions, 2015, pp. 43–48, CoRR URL http://arxiv.org/abs/1505.01658.
[7] S. Rezvani, X. Wang, Class imbalance learning using fuzzy ART and intuitionistic fuzzy twin support vector machines, Inform. Sci. 578 (2021) 659–682, http://dx.doi.org/10.1016/j.ins.2021.07.010.
[8] B. Krawczyk, Learning from imbalanced data: Open challenges and future directions, Progress Artif. Intell. 5 (4) (2016) 221–232, http://dx.doi.org/10.1007/s13748-016-0094-0.
[9] N. Rout, D. Mishra, M.K. Mallick, Handling imbalanced data: A survey, in: M.S. Reddy, K. Viswanath, S.P. K.M. (Eds.), International Proceedings on Advances in Soft Computing, Intelligent Systems and Applications, Springer Singapore, Singapore, ISBN: 978-981-10-5272-9, 2018, pp. 431–443.
[10] S. Tyagi, S. Mittal, Sampling approaches for imbalanced data classification problem in machine learning, in: Proceedings of ICRIC 2019, 2019, pp. 209–221.
[11] J. Yao, Y. Zheng, H. Jiang, An ensemble model for fake online review detection based on data resampling, feature pruning, and parameter optimization, IEEE Access 9 (2021) 16914–16927, http://dx.doi.org/10.1109/ACCESS.2021.3051174.
[12] M. Singla, D. Ghosh, K. Shukla, A survey of robust optimization based machine learning with special reference to support vector machines, Int. J. Mach. Learn. Cybern. 11 (2020) 1359–1385, http://dx.doi.org/10.1007/s13042-019-01044-y.
[13] A.N. Tarekegn, M. Giacobini, K. Michalak, A review of methods for imbalanced multi-label classification, Pattern Recognit. 118 (2021) 107965, http://dx.doi.org/10.1016/j.patcog.2021.107965.
[14] D. Devi, S.K. Biswas, B. Purkayastha, A review on solution to class imbalance problem: Undersampling approaches, in: 2020 International Conference on Computational Performance Evaluation, ComPE, 2020, pp. 626–631, http://dx.doi.org/10.1109/ComPE49325.2020.9200087.
[15] J. Van Pulse, T. hi Jehoshaphat, Knowledge discovery from imbalanced and noisy data, Data Knowl. Eng. 68 (12) (2009) 1513–1542.
[16] T.J. Lakshmi, C.S.R. Prasad, A study on classifying imbalanced datasets, in: 2014 First International Conference on Networks Soft Computing, ICNSC2014, 2014, pp. 141–145, http://dx.doi.org/10.1109/CNSC.2014.6906652.
[17] A. Azaria, A. Richardson, S. Kraus, V.S. Subrahmanian, Behavioral analysis of insider threat: A survey and bootstrapped prediction in imbalanced data, IEEE Trans. Comput. Soc. Syst. 1 (2) (2014) 135–155, http://dx.doi.org/10.1109/TCSS.2014.2377811.
[18] M. Woniak, M. Graña, E. Corchado, A survey of multiple classifier systems as hybrid systems, Inf. Fusion 16 (2014) 3–17.

[19] G.M. Weiss, K. McCarthy, B. Zabar, Cost-sensitive learning vs. Sampling: Which is best for handling unbalanced classes with unequal error costs? in: Proceedings of the International Conference on Data Mining, 2007, pp. 25–28.
[20] C.R. Milaré, G.E.A.P.A. Batista, A.C.P.L.F. Carvalho, A hybrid approach to learn with imbalanced classes using evolutionary algorithms, Logic J. IGPL 19 (2) (2011) 293–303, http://dx.doi.org/10.1093/jigpal/jzq027.
[21] V. Ganganwar, An overview of classification algorithms for imbalanced datasets, Int. J. Emerg. Technol. Adv. Eng. 2 (4) (2012) 42–47.
[22] D. Ramyachitra, P. Manikandan, Imbalance dataset classification and solutions: A review, Int. J. Comput. Bus. Res. 5 (4) (2014).
[23] G. Wang, J.Y.-C. Teoh, J. Lu, K.S. Cho, Least squares support vector machines with fast leave-one-out AUC optimization on imbalanced prostate cancer data, Int. J. Mach. Learn. Cybern. 11 (2020) 1909–1922, http://dx.doi.org/10.1007/s13042-020-01081-y.
[24] W. Xue, P. Zhong, W. Zhang, G. Yu, Y. Chen, Sample-based online learning for bi-regular hinge loss, Int. J. Mach. Learn. Cybern. 12 (2021) 1753–1768, http://dx.doi.org/10.1007/s13042-020-01272-7.
[25] C. Zhang, Y. Zhou, J. Guo, G. Wang, X. Wang, Research on classification method of high-dimensional class-imbalanced datasets based on SVM, Int. J. Mach. Learn. Cybern. 10 (2019) 1765–1778, http://dx.doi.org/10.1007/s13042-018-0853-2.
[26] D.R. Don, I.E. Iacob, DCSVM: Fast multi-class classification using support vector machines, Int. J. Mach. Learn. Cybern. 11 (2020) 433–447, http://dx.doi.org/10.1007/s13042-019-00984-9.
[27] S. Rezvani, X. Wang, F. Pourpanah, Intuitionistic fuzzy twin support vector machines, IEEE Trans. Fuzzy Syst. 27 (11) (2019) 2140–2151, http://dx.doi.org/10.1109/TFUZZ.2019.2893863.
[28] S. Rezvani, Ranking method of trapezoidal intuitionistic fuzzy numbers, Ann. Fuzzy Math. Inform. 5 (3) (2013) 515–523.
[29] H. He, Y. Ma, Imbalanced Learning: Foundations, Algorithms, and Applications, first ed., Wiley-IEEE Press, ISBN: 1118074629, 2013.
[30] M.A. Ganaie, M. Tanveer, Alzheimer's Disease Neuroimaging Initiative, KNN weighted reduced universum twin SVM for class imbalance learning, Knowl.-Based Syst. 245 (7) (2022) 108578.
[31] A. Barbado, Ó. Corcho, R. Benjamins, Rule extraction in unsupervised anomaly detection for model explainability: Application to OneClass SVM, Expert Syst. Appl. 189 (2022) 116100.
[32] M. Akpinar, M. FatihAdak, G. Guvenc, SVM-based anomaly detection in remote working: Intelligent software SmartRadar, Appl. Soft Comput. 109 (2021) 107457.
[33] Y. Ji, H. Lee, Event-based anomaly detection using a one-class SVM for a hybrid electric vehicle, IEEE Trans. Veh. Technol. 71 (6) (2022) 6032–6043.
[34] H. Han, W. Wen-Yuan, M. Bing-Huan, Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning, in: Advances in Intelligent Computing, Springer, Berlin, 2005, pp. 878–887.
[35] C.V. Rijsbergen, Information Retrieval, second ed., Dept. of Computer Science, University of Glasgow, 1979.
[36] G. Myatt, W. Johnson, Making Sense of Data II, John Wiley and Sons, Ltd, pp. 111–163, http://dx.doi.org/10.1002/9780470417409.ch4, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470417409.ch4.
[37] Y. Tang, S. Krasser, P. Judge, Y. Zhang, Fast and effective spam sender detection with granular SVM on highly imbalanced mail server behavior data, in: Proceedings of 2nd International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborativeCom), 2006, pp. 1–6.
[38] V.N. Vapnik, Statistical Learning Theory, first ed., John Wiley and Sons, New York, 1998.
[39] M. Kubat, R.C. Holte, S. Matwin, Machine learning for the detection of oil spills in satellite radar images, Mach. Learn. 30 (1998) 195–215.
[40] C.E. Metz, Basic principles of roc analysis, in: Seminars in Nuclear Medicine, vol. 8, 1978, pp. 283–298.
[41] F.J. Provost, T. Fawcett, R. Kohavi, The case against accuracy estimation for comparing induction algorithms, in: ICML'98: Proc. of the 15th Int. Conf. on Machine Learning, 1998, pp. 445–453.
[42] V. Vapnik, The Nature of Statistical Learning Theory, Pringer-Verlag New York, Inc, 1995.
[43] R. Batuwita, V. Palade, Efficient resampling methods for training support vector machines with imbalanced datasets, in: Proceedings of the International Joint Conference on Neural Networks, 2010, pp. 1–8.
[44] A. Estabrooks, T. Jo, N. Japkowicz, A multiple resampling method for learning from imbalanced data sets, Comput. Intell. 20 (2004) 18–36.
[45] A. Fernandez, S. Garcia, M.J. del Jesus, F. Herrera, A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data-sets, Fuzzy Sets and Systems 159 (2008) 2378–2398.
[46] A. Fernandez, M.J. del Jesus, F. Herrera, On the 2-tuples based genetic tuning performance for fuzzy rule based classification systems in imbalanced data-sets, Inform. Sci. 180 (2010) 1268–1291.
[47] L. Torgo, R.P. Ribeiro, B. Pfahringer, P. Branco, Smote for regression, in: Progress in Artificial Intelligence, Springer, 2013, pp. 378–389.
[48] N. Chawla, K. Bowyer, L. Hall, W. Kegelmeyer, SMOTE: Synthetic minority over-sampling technique, J. Artificial Intelligence Res. 16 (2002) 321–357.
[49] G.M. Weiss, F.J. Provost, Learning when training data are costly: The effect of class distribution on tree induction, J. Artif. Intell. Res 19 (2003) 315–354.
[50] N.V. Chawla, L.O. Hall, A. Joshi, Wrapper-based computation and evaluation of sampling methods for imbalanced datasets, in: Proceedings of the 1st International Workshop on Utility-Based Data Mining, 2005, pp. 24–33.
[51] N.V. Chawla, D.A. Cieslak, L.O. Hall, A. Joshi, Automatically countering imbalance and its empirical relationship to cost, Data Min. Knowl. Discov. 17 (2008) 225–252.
[52] L. Breiman, J.H. Friedman, R.A. Olshen, C.J. Stone, Classification and Regression Trees, Wadsworth and Brooks. Monterey, CA, 1984.
[53] C. Drummond, R.C. Holte, C4. 5, class imbalance, and cost sensitivity: Why under-sampling beats over-sampling, in: Workshop on Learning from Imbalanced Datasets II, Vol. 11, 2003.
[54] S. Wang, X. Yao, Diversity analysis on imbalanced data sets by using ensemble models, in: 2009 IEEE Symposium on Computational Intelligence and Data Mining, 2009, pp. 324–331.
[55] E.Y. Chang, B. Li, G. Wu, K. Goh, Statistical learning for effective visual information retrieval, in: IEEE International Conference on Image Processing, 2003.
[56] D. Tao, X. Tang, X. Li, X. Wu, Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 28 (2006) 1088–1099.
[57] C. Chen, A. Liaw, L. Breiman, Using Random Forest to Learn Imbalanced Data, University of California, Berkeley, 2004.
[58] I. Mani, J. Zhang, Knn approach to unbalanced data distributions: A case study involving information extraction, in: Proceedings of Workshop on Learning from Imbalanced Datasets, 2003.
[59] S. Garcia, J.R. Cano, A. Fernandez, F. Herrera, A proposal of evolutionary prototype selection for class imbalance problems, Intell. Data Eng. Automat. Learn., IDEAL (2006) 1415–1423.
[60] S. Garcia, F. Herrera, Evolutionary undersampling for classification with imbalanced datasets: Proposals and taxonomy, Evol. Comput. 17 (2009) 275–306.
[61] G. Menardi, N. Torelli, Training and assessing classification rules with imbalanced data, Data Min. Knowl. Discov. 28 (2014) 92.122.
[62] Z.E.A. Elassad, H. Mousannif, H.A. Moatassime, Class-imbalanced crash prediction based on real-time traffic and weather data: A driving simulator study, Traffic Inj. Prev. 21 (3) (2020) 201–208.
[63] W. A.Rivera, P. Xanthopoulos, A priori synthetic over-sampling methods for increasing classification sensitivity in imbalanced data sets, Expert Syst. Appl. 66 (30) (2016) 124–135.
[64] B. Makond, K.J. Wang, K.M. Wang, Benchmarking prognosis methods for survivability. A case study for patients with contingent primary cancers, Comput. Biol. Med. 138 (2021) 104888.
[65] L. Lian, X. Luo, C. Pan, J. Huang, W. Hong, Z. Xu, Lung image segmentation based on DRD U-Net and combined WGAN with deep neural network, Comput. Methods Programs Biomed. (2022) 107097, http://dx.doi.org/10.1016/j.cmpb.2022.107097.
[66] M. Hammad, N. Hewahi, W. Elmedany, MMM-RF: A novel high accuracy multinomial mixture model for network intrusion detection systems, Comput. Secur. 120 (2022) 102777, http://dx.doi.org/10.1016/j.cose.2022.102777.
[67] N. Chawla, A. Lazarevic, L. Hall, K. Bowyer, Smoteboost: Improving prediction of the minority class in boosting, in: Proceedings of the Principles of Knowledge Discovery in Databases, 2003, pp. 107–119.
[68] T. Maciejewski, J. Stefanowski, Local neighbourhood extension of smote for mining imbalanced data, in: IEEE Symposium on Computational Intelligence and Data Mining, CIDM, 2011, pp. 104–111.
[69] S. Barua, M.I.X. Yao, K. Murase, Mwmote-majority weighted minority oversampling technique for imbalanced data set learning, IEEE Trans. Knowl. Data Eng. 26 (2014) 405–425.
[70] C. Bunkhumpornpat, K. Sinapiromsaran, C. Lursinsap, Safelevel-smote: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem, Adv. Knowl. Discov. Data Min. (2009) 475–482.
[71] E. Ramentol, N. Verbiest, R. Bello, Y. Canallero, C. Cornelis, F. Herrera, Smote-first: A new resampling method using fuzzy rough set theory, in: World Scientific Proceedings Series on Computer Engineering and Information Science Uncertainty Modeling in Knowledge Engineering and Decision Making, 2012, pp. 800–805.
[72] N. Verbiest, E. Ramentol, C. Cornelis, F. Herrera, Improving smote with fuzzy rough prototype selection to detect noise in imbalanced classification data, Adv. Artif. Intell. IBERAMIA (2012) 169–178.
[73] J. Stefanowski, S. Wilk, Improving rule based classifiers induced by modlem by selective pre-processing of imbalanced data, in: Proc. of the RSKD Workshop at ECML/PKDD, Warsaw, 2007, pp. 54–65.
[74] S.S. Lee, Regularization in skewed binary classification, Comput. Statist. 14 (1999) 277–292.

[75] S. Lee, Noisy replication in skewed binary classification, Comput. Statist. Data Anal. 34 (2000) 165–191.
[76] S. Hu, Y. Liang, L. Ma, Y. He, Msmote: Improving classification performance when training data is imbalanced, in: Second International Workshop on Computer Science and Engineering, Vol. 2, 2009, pp. 13–17.
[77] D. Zhang, W. Liu, X. Gong, H. Jin, A novel improved smote resampling algorithm based on fractal, J. Comput. Inf. Syst. 7 (2011) 2204–2211.
[78] C. Bunkhumpornpat, S. Subpaiboonkit, Safe level graph for synthetic minority over-sampling techniques, in: 13th International Symposium on Communications and Information Technologies, ISCIT, 2013, pp. 570–575.
[79] C. Bunkhumpornpat, K. Sinapiromsaran, C. Lursinsap, Dbsmote: Density-based synthetic minority over-sampling technique, Appl. Intell. 36 (2012) 664–684.
[80] I. Tomek, Two modifications of CNN, IEEE Trans. Syst. Man Cybern. 11 (1976) 769–772.
[81] G.E. Batista, R.C. Prati, M.C. Monard, A study of the behavior of several methods for balancing machine learning training data, ACM SIGKDD Explor. Newsl. 6 (2004) 20–29.
[82] P.E. Hart, The condensed nearest neighbor rule, IEEE Trans. Inform. Theory 14 (1968) 515–516.
[83] M. Kubat, S. Matwin, Addressing the curse of imbalanced training sets: One-sided selection, in: Proc. of the 14th Int. Conf. on Machine Learning, 1997, pp. 179–186.
[84] E. Ramentol, Y. Canallero, R. Bello, F. Herrera, Smote-rsb: A hybrid preprocessing approach based on oversampling and undersampling for high imbalanced data-sets using smote and rough sets theory, Knowl. Inf. Syst. 33 (2012) 245–265.
[85] J. Chen, M. Casique, M. Karakoy, Classification of lung data by sampling and support vector machine, in: In Proceedings of the 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Vol. 2, 2004, pp. 3194–3197.
[86] Y. Fu, S. Ruixiang, Q. Yang, H. Simin, C. Wang, H. Wang, S. Shan, J. Liu, W. Gao, A block-based support vector machine approach to the protein homology prediction task in kdd cup 2004, SIGKDD Explor. Newslett. 6 (2004) 120–124.
[87] S. Lessmann, Solving imbalanced classification problems with support vector machines, in: Proceedings of the International Conference on Artificial Intelligence, 2004, pp. 214–220.
[88] R. Batuwita, V. Palade, An improved non-comparative classification method for human microrna gene prediction, in: Proceedings of the International Conference on Bioinformatics and Bioengineering, 2008, pp. 1–6.
[89] R. Batuwita, V. Palade, Micropred: Effective classification of pre-mirnas for human mirna gene prediction, Bioinformatics 25 (2009) 989–995.
[90] C. Bellinger, S. Sharma, N. Japkowicz, One-class versus binary classification: Which and when? in: 2012 11th International Conference on Machine Learning and Applications, Vol. 2, 2012, pp. 102–106.
[91] L. Xuan, C. Zhigang, Y. Fan, Exploring of clustering algorithm on class-imbalanced data, in: 2013 8th International Conference on Computer Science and Education, 2013, pp. 89–93.
[92] R.C. Holte, L.E. Acker, B.W. Porter, Concept learning and the problem of small disjuncts, IJCAI 89 (1989) 813–818.
[93] T. Jo, N. Japkowicz, Class imbalances versus small disjuncts, ACM SIGKDD Explor. Newsl. 6 (2004) 40–49.
[94] S. Yen, Y.S. Lee, Under-sampling approaches for improving prediction of the minority class in an imbalanced dataset, in: Intelligent Control and Automation, 2006, pp. 731–740.
[95] S.J. Yen, Y.S. Lee, Cluster-based under-sampling approaches for imbalanced data distributions, Expert Syst. Appl. 36 (2009) 5718–5727.
[96] G. Cohen, M. Hilario, H. Sax, S. Hugonnet, A. Geissbuhler, Learning from imbalanced data in surveillance of nosocomial infection, Artif. Intell. Med. 37 (2006) 7–18.
[97] J. Yuan, J. Li, B. Zhang, Learning concepts from large scale imbalanced data sets using support cluster machines, in: Proceedings of the 14th Annual ACM International Conference on Multimedia, 2006, pp. 441–450.
[98] Z. Lin, Z. Hao, X. Yang, X. Liu, Several svm ensemble methods integrated with under-sampling for imbalanced data learning, in: Proceedings of the 5th International Conference on Advanced Data Mining and Applications, 2009, pp. 536–544.
[99] P. Kang, S. Cho, Eus svms: Eensemble of under-sampled svms for data imbalance problems, in: Proceedings of the 13th International Conference on Neural Information Processing, 2006, pp. 837–846.
[100] Y. Liu, A. An, X. Huang, Boosting prediction accuracy on imbalanced datasets with svm ensembles, in: Proceedings of the 10th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, 2006, pp. 107–118.
[101] B. Wang, N. Japkowicz, Boosting support vector machines for imbalanced data sets, Knowl. Inf. Syst. 25 (2010) 1–20.
[102] W. Fan, S. Stolfo, J. Zhang, P. Chan, Adacost: Misclassification cost-sensitive boosting, in: In Proceedings of the 16th International Conference on Machine Learning, 1999, pp. 97–105.
[103] M. Joshi, V. Kumar, C. Agarwal, Evaluating boosting algorithms to classify rare classes: Comparison and improvements, in: Proceedings of the IEEE International Conference on Data Mining, 2001, pp. 257–264.
[104] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, F. Herrera, KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework, J. Multiple-Valued Logic Soft Comput 17 (2–3) (2011) 255–287.
[105] S. Kurin, A comparison of classification models for imbalanced datasets, Université Catholique de Louvain, 2017. Prom, 2017.
[106] A. Liu, J. Ghosh, C.E. Martin, Generative oversampling for mining imbalanced datasets, DMIN (2007) 66–72.
[107] J.M. Martinez-Garcia, C.P. Suarez-Araujo, P.G. Baaez, Sneom: A sanger network based extended over-sampling method. Application to imbalanced biomedical datasets, in: Neural Information Processing, 2012, pp. 584–592.
[108] B. Zadrozny, J. Langford, N. Abe, Cost-sensitive learning by cost-proportionate example weighting, in: ICDM'03 Proceedings of the Third IEEE International Conference on Data Mining, 2003, pp. 19–22.
[109] S. Rezvani, X. Wang, Erratum to entropy-based fuzzy support vector machine for imbalanced datasets" [Knowl.-Based Syst. 115 (2017) 87–99], Knowl.-Based Syst. 192 (2020) 105287, http://dx.doi.org/10.1016/j.knosys.2019.105287.
[110] G. Wu, E. Chang, Class-boundary alignment for imbalanced dataset learning, in: ICML 2003 Workshop on Learning from Imbalanced Data Sets II, Washington, DC, 2003.
[111] R. Batuwita, V. Palade, Fsvm-cil: Fuzzy support vector machines for class imbalance learning, IEEE Trans. Fuzzy Syst. 18 (2010) 558–571.
[112] C. Li, C. Jing, G. Xin-tao, An improved p-svm method used to deal with imbalanced data sets, in: IEEE International Conference on Intelligent Computing and Intelligent Systems, Vol. 1, 2009, pp. 118–122.
[113] R. Barandela, J.S. Sanchez, V. Garcia, E. Rangel, Strategies for learning in class imbalance problems, Pattern Recognit. 36 (2003) 849–851.
[114] G.M. Weiss, Mining with rarity: A unifying framework, SIGKDD Explor. Newslett. 6 (2004) 7–19.
[115] M.A. Maloof, Learning when data sets are imbalanced and when costs are unequal and unknown, in: ICML-2003 Workshop on Learning from Imbalanced Data Sets II, Vol. 2, 2003, pp. 1–2.
[116] K. Veropoulos, C. Campbell, N. Cristianini, Controlling the sensitivity of support vector machines, in: Proceedings of the International Joint Conference on AI, 1995, pp. 55–60.
[117] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and other kernel-based learning methods, Cambridge University Press, Cambridge, UK, 2000.
[118] H. Ma, L. Wang, B. Shen, A new fuzzy support vector machines for class imbalance learning, in: 2011 International Conference on Electrical and Control Engineering, 2011, pp. 3781–3784.
[119] T. Imam, K. Ting, J. Kamruzzaman, Z-svm: An svm for improved classification of imbalanced data, in: Proceedings of the 19th Australian Joint Conference on Artificial Intelligence: Advances in Artificial Intelligence, 2006, pp. 264–273.
[120] B. Raskutti, A. Kowalczyk, Extreme re-balancing for svms: A case study, SIGKDD Explor. Newslett. 6 (2004) 60–69.
[121] A. Kowalczyk, B. Raskutti, One class svm for yeast regulation prediction, SIGKDD Explor. Newslett. 4 (2002) 99–100.
[122] P. Domingos, Metacost: A general method for making classifiers cost-sensitive, in: KDD'99: Proceedings of the 5th International Conference on Knowledge Discovery and Data Mining, 1999, pp. 155–164.
[123] A.P. Sinha, J.H. May, Evaluating and tuning predictive data mining models using receiver operating characteristic curves, J. Manage. Inf. Syst. 21 (2004) 249–280.
[124] Y. Freund, R. Schapire, A decision-theoretic generalization of online learning and an application to boosting, in: Proceedings of the Second European Conference on Computational Learning Theory, 1995.
[125] Y. Sun, M.S. Kamel, A.K. Wong, Y. Wang, Cost-sensitive boosting for classification of imbalanced data, Pattern Recognit. 40 (2007) 3358–3378.
[126] J. Song, X. Lu, X. Wu, An improved adaboost algorithm for unbalanced classification data, in: FSKD'09 Proceedings of the 6th International Conference on Fuzzy Systems and Knowledge Discovery, Vol. 1, 2009, pp. 109–113.
[127] W. Liu, S. Chawla, D.A. Cieslak, N.V. Chawla, A robust decision tree algorithm for imbalanced data sets, SDM 10 (2010) 766–777.
[128] D.A. Cieslak, N.V. Chawla, Learning decision trees for unbalanced data, Mach. Learn. Knowl. Discov. Databases (2008) 241–256.
[129] D.A. Cieslak, T.R. Hoens, N.V. Chawla, W.P. Kegelmeyer, Hellinger distance decision trees are robust and skew-insensitive, Data Min. Knowl. Discov. 24 (2012) 136–158.
[130] Z.H. Zhou, X.Y. Liu, Training cost-sensitive neural networks with methods addressing the class imbalance problem, IEEE Trans. Knowl. Data Eng. 18 (2006) 63–77.

[131] R. Alejo, V. Garcia, J.M. Sotoca, R.A. Mollineda, J.S. Sanchez, Improving the performance of the rbf neural networks trained with imbalanced samples, Comput. Ambient Intell. (2007) 162–169.
[132] S.H. Oh, Error back-propagation algorithm for classification of imbalanced data, Neurocomputing 74 (2011) 1058–1061.
[133] C.L. Castro, A. de Padua Braga, Novel cost-sensitive approach to improve the multilayer perceptron performance on imbalanced data, IEEE Trans. Neural Netw. Learn. Syst. 24 (2013) 888–899.
[134] P. Cao, D. Zhao, O.R. Zaiane, A pso-based cost-sensitive neural network for imbalanced data classification, Trends Appl. Knowl. Discov. Data Min. (2013) 452–463.
[135] L. Torgo, R.P. Ribeiro, Predicting outliers, Knowl. Discov. Databases: PKDD (2003) 447–458.
[136] R.P. Ribeiro, L. Torgo, Predicting harmful algae blooms, in: Portuguese Conference on Artificial Intelligence EPIA 2003: Progress in Artificial Intelligence, 2003, pp. 308–312.
[137] R.P. Ribeiro, Utility-Based Regression, (Ph.D. thesis), Dep. Computer Science, Faculty of Sciences - University of Porto, 2011.
[138] G. Bansal, A.P. Sinha, H. Zhao, Tuning data mining methods for cost-sensitive regression: A study in loan charge-off forecasting, J. Manage. Inf. Syst. 25 (2008) 315–336.
[139] H. Zhao, A.P. Sinha, G. Bansal, An extended tuning method for cost-sensitive regression and forecasting, Decis. Support Syst. 51 (2011) 372–383.
[140] J. Hernandez-Orallo, Soft (Gaussian cde) regression models and loss functions, 2012, arXiv preprint arXiv:1211.1043.
[141] J. Hernandez-Orallo, Probabilistic reframing for cost-sensitive regression, ACM Trans. Knowl. Discov. Data 8 (2014) 1–17.
[142] A. Estabrooks, N. Japkowicz, A mixture-of-experts framework for learning from imbalanced data sets, in: Advances in Intelligent Data Analysis, 2001, pp. 34–43.
[143] C. Seiffert, T.M. Khoshgoftaar, J.V. Hulse, A. Napolitano, Rusboost: A hybrid approach to alleviating class imbalance, IEEE Trans. Syst. Man Cybern. A 40 (2010) 185–197.
[144] A. Tan, D. Gilbert, Y. Deville, Multi-class protein fold classification using a new ensemble machine learning approach, Genome Inform. 14 (2003) 206–217.
[145] J. Xiao, L. Xie, C. He, X. Jiang, Dynamic classifier ensemble model for customer classification with imbalanced class distribution, Expert Syst. Appl. 39 (2012) 3668–3675.
[146] S. Kotsiantis, P. Pintelas, Mixture of expert agents for handling imbalanced data sets, Ann. Math., Comput. Teleinform. 1 (2003) 46–55.
[147] C. Phua, D. Alahakoon, V. Lee, Minority report in fraud detection: Classification of skewed data, ACM SIGKDD Explor. Newsl. 6 (2004) 50–59.
[148] M. Moya, D. Hush, Network contraints and multiobjective optimization for one-class classification, Neural Netw. 9 (1996) 463–474.
[149] N. Japkowicz, C. Myers, M. Gluck, A novelty detection approach to classification, in: Proceedings of the Fourteenth Joint Conference on Artificial Intelligence, 1995, pp. 518–523.
[150] N. Japkowicz, Learning from imbalanced data sets: A comparison of various strategies, in: AAAI Workshop on Learning from Imbalanced Data Sets, Vol. 68, 2000, pp. 10–15.
[151] B. Scholkopf, J.C. Platt, J. Shawe-Taylor, A.J. Smola, R.C. Williamson, Estimating the support of a high-dimensional distribution, Neural Comput. 13 (2001) 1443–1471.
[152] L. Manevitz, M. Yousef, One-class svms for document classification, J. Mach. Learn. Res. 2 (2002) 139–154.
[153] L. Zhuang, H. Dai, Parameter estimation of one-class svm on imbalance text classification, in: Advances in Artificial Intelligence, 2006, pp. 538–549.
[154] H.J. Lee, S. Cho, The novelty detection approach for different degrees of class imbalance, in: Neural Information Processing, 2006, pp. 21–30.
[155] X.Y. Liu, J. Wu, Z.H. Zhou, Exploratory undersampling for class-imbalance learning, IEEE Trans. Syst. Man Cybern. B 39 (2009) 539–550.
[156] H.Y. Wang, Combination approach of smote and biased-svm for imbalanced datasets, in: International Joint Conference on Neural Networks, IJCNN 2008, 2008, pp. 228–231.
[157] G. Wu, E. Chang, Kba: Kernel boundary alignment considering imbalanced data distribution, IEEE Trans. Knowl. Data Eng. 17 (2005) 786–795.
[158] J. Doucette, M.I. Heywood, Gp classification under imbalanced data sets: Active sub-sampling and auc approximation, Genetic Programm. (2008) 266–277.
[159] S. Maheshwari, J. Agrawal, S. Sharma, A new approach for classification of highly imbalanced datasets using evolutionary algorithms, Intl. J. Sci. Eng. Res. 2 (2011) 1–5.
[160] Y. Yong, The research of imbalanced data set of sample sampling method based on k-means cluster and genetic algorithm, Energy Procedia 17 (2012) 164–170.
[161] M. Galar, A. Fernandez, E. Barrenechea, F. Herrera, Eusboost: Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling, Pattern Recognit. 46 (2013) 3460–3471.
[162] J. Stefanowski, S. Wilk, Selective pre-processing of imbalanced data for improving classification performance, in: DaWaK 2008: Data Warehousing and Knowledge Discovery, 2008, pp. 283–292.
[163] C. Bunkhumpornpat, K. Sinapiromsaran, C. Lursinsap, Mute: Majority under-sampling technique, in: 8th International Conference on Information, Communications and Signal Processing, 2011, pp. 1–4.
[164] P. Songwattanasiri, K. Sinapiromsaran, Smoute: Synthetics minority oversampling and under-sampling techniques for class imbalanced problem, in: Proceedings of the Annual International Conference on Computer Science Education: Innovation and Technology, Special Track: Knowledge Discovery, 2010, pp. 78–83.
[165] Z.Z. Yang, D. Gao, An active under-sampling approach for imbalanced data classification, in: Fifth International Symposium on Computational Intelligence and Design, Vol. 2, 2012, pp. 270–273.
[166] P. Jeatrakul, K.W. Wong, C.C. Fung, Classification of imbalanced data by combining the complementary neural network and smote algorithm, Neural Inf. Process.. Models Appl. (2010) 152–159.
[167] D. Mease, A. Wyner, A. Buja, Cost-weighted boosting with jittering and over/under-sampling: Jous-boost, J. Mach. Learn. Res. 8 (2007) 409–439.
[168] S. Chen, H. He, E.A. Garcia, Ramoboost: Ranked minority oversampling in boosting, IEEE Trans. Neural Netw. 21 (2010) 1624–1642.
[169] S. Ertekin, J. Huang, L. Giles, Active learning for class imbalance problem, in: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2007, pp. 823–824.
[170] J. Zhu, E.H. Hovy, Active learning for word sense disambiguation with methods for addressing the class imbalance problem, EMNLP-CoNLL 7 (2007) 783–790.
[171] S. Ertekin, Adaptive oversampling for imbalanced data classification, Inf. Sci. Syst. (2013) 261–269.
[172] Y. Mi, Imbalanced classification based on active learning smote, Res. J. Appl. Sci. Eng. Technol. 5 (2013) 944–949.
[173] J. Hu, Active learning for imbalance problem using l-gem of rbfnn, ICMLC (2012) 490–495.
[174] K. Madasamy, M. Ramaswami, Data imbalance and classifiers: Impact and solutions from a big data perspective, Int. J. Comput. Intell. Res. 13 (9) (2017) 2267–2281.
[175] D. Dua, C. Graff, UCI machine learning repository, 2017, URL http://archive.ics.uci.edu/ml.
[176] B. Efron, Bootstrap methods: Another look at the jackknife, Ann. Statist. 7 (1) (1979) 1–26.
[177] J. Huang, C.X. Ling, Using auc and accuracy in evaluating learning algorithms, IEEE Trans. Knowl. Data Eng. 17 (3) (2005) 299–310.
[178] S. Rezvani, X. Wang, Intuitionistic fuzzy twin support vector machines for imbalanced data, Neurocomputing 507 (2022) 16–25.
[179] M. Friedman, A comparison of alternative tests of significance for the problem of m rankings, Ann. Math. Stat. 11 (1) (1940) 86–92.
[180] O.J. Dunn, Multiple comparisons among means, J. Amer. Statist. Assoc. 56 (293) (1961) 52–64.
[181] J. Demar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (1) (2006) 1–30.
[182] E. Frank, M. Hall, I. Witten, The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", fourth ed., Morgan Kaufmann, Burlington, 2016.