Ensemble learning with dynamic weighting for response modeling in direct marketing

Xin Zhang[a], Yalan Zhou[b], Zhibin Lin[c],Yu Wang[d,*]

[a] Department of Nursing, Children's Hospital of Chongqing Medical University, National Clinical Research Center for Child Health and Diseases, Ministry of Education Key Laboratory of Child Development and Disorders, Chongqing Key Laboratory of Pediatrics, Chongqing, 400010, China

[b] Army Logistics Academy, Chongqing, 401311, China

[c] Durham University Business School, Durham University, Durham, UK

[d] School of Economics and Business Administration, Chongqing University, Chongqing, 400030, China

* Corresponding author.
Tel: 86-23- 65105761.
Email: yuwang@cqu.edu.cn.

Abstract: Response modeling, a key to successful direct marketing, has become increasingly prevalent in recent years. However, it practically suffers from the difficulty of class imbalance, i.e., the number of responding (target) customers is often much smaller than that of the non-responding customers. This issue would result in a response model that is biased to the majority class, leading to the low prediction accuracy on the responding customers. In this study, we develop an Ensemble Learning with Dynamic Weighting (ELDW) approach to address the above problem. The proposed ELDW includes two stages. In the first stage, all the minority class instances are combined with different majority class instances to form a number of training subsets, and a base classifiers is trained in each subset. In the second stage, the results of the base classifiers are dynamically integrated, in which two factors are considered. The first factor is the cross entropy of neighbors in each subset, and the second factor is the feature similarity to the minority class instances. In order to evaluate the performance of ELDW, we conduct experimental studies on 10 imbalanced benchmark datasets. The results show that compared with other state-of-the-art imbalance classification algorithms, ELDW achieves higher accuracy on the minority class. Last, we apply the ELDW to a direct marketing activity of an insurance company to identify the target customers under a limited budget.
Keywords: direct marketing; response modeling; imbalance classification; dynamic ensemble learning.

1. Introduction

For product/service advertising and promotions, there are mainly two tactics used in practice: massive marketing and direct marketing. Massive marketing employs mass media, such as television, radio, magazines and newspapers, to deliver uniform product/service information to customers. As the market competition becomes severe, more and more firms are interested in direct marketing, and devote their resources to profiling and identifying the customers that may be open to specific marketing messages or campaigns (Shah and Murthi, 2021). Compared with massive marketing that provides uniform product/service information to customers without differentiations, direct marketing delivers different marketing messages to different customers. Owing to the growing popularity of direct marketing in various industries, academic interest and research in direct marketing are flourishing.

In direct marketing, a firm targets a customer with a marketing solicitation such as a catalog, a direct solicitation, or a coupon, and the customer decides whether or not to respond. Since soliciting a customer unlikely to respond is unprofitable, while not soliciting a potentially profitable customer means leaving money on the table, a key problem in direct marketing is the so-called response modeling, which refers to identifying the customers who will respond to a specific marketing program with high probabilities. Well-developed and accurate response models used for targeting specific customers contribute to firms by not only increasing the revenues, but also lowering the marketing costs (Kang et al. 2012; Trusov et al. 2016; Bradlow et al. 2017). Baesens et al. (2002) argue that an increase of only one percentage in response can result in substantial profit increases, which is illustrated by a real-life mail-order company. Knott et al. (2002) also point out that for a retail service bank, 0.7 percent of extra correct targets would increase 20% revenue for each customer.

In direct marketing, customers may choose to buy the products or not after receiving the marketing solicitations. Therefore, the customers' response to the marketing activities can be regarded as a binary classification problem. The higher a customer’s response probability is, the more likely she/he is to purchase. Due to the limitation of marketing budget, it is impossible to carry out marketing activities for all customers. Therefore, identifying those customers with high response probabilities can help the marketing manager to target more responding customers under the limited budget.

Since it is profitable to identify the target customers, the ability of predicting customers' response has long been a crucial endeavor for both practitioners and academics (Chun 2012; Liu et al. 2022). Conventional statistical analysis methods and machine learning algorithms, such as Logistic Regression (LR) (De Caigny et al. 2018; Dumitrescu et al. 2022), Genetic Algorithm (GA) (Stripling et al. 2018; Chan et al. 2020), Random Forest (RF) (Ładyżyński et al. 2019; Rao et al. 2020), and Artificial Neural Network (ANN) (Chaudhuri et al. 2021; Kim et al. 2021), have been applied to construct the response model based on the information of customers' behavior (Roy et al. 2018; Ng et al. 2021). However, when the number of customers responding to the marketing activities is much smaller than that of customers who do not respond, the prediction of customer's response becomes an imbalance classification problem, bringing a critical challenge for the performance of response model.

In imbalance classification problems, the data has a skewed class distribution such that the majority class contains a large number of instances while the minority class contains a small number of instances (Chen et al. 2021). Moreover, the minority instances are often the class of interest in practice. Conventional classifiers minimizing the overall training error may force the decision boundary to move closer to the minority class (Zheng et al. 2021). In this situation, the constructed model generally biases towards the majority instances with ignorance on the minority instances, leading to the low prediction accuracy of minority instances (Liu et al. 2022). In this paper, the minority class instances represent the customers who respond to the marketing activities and bring profits to the firm.

Until recently, there have been a great number of approaches to solving the imbalanced classification problem. These approaches can be generally divided into two categories: data-level and algorithm-level approaches (Fu et al. 2021). The former reduces the imbalance level of dataset by adding or removing instances as a pre-processing step (Hppner et al. 2021). Among them, the over-sampling methods generate more minority instances, while the under-sampling methods discard some of the majority instances, to balance the dataset (Gao et al. 2020). The algorithm-level approaches mainly include cost-sensitive learning methods and ensemble learning methods. Cost-sensitive learning methods assume that the misclassification cost of minority class instances is often much more expensive than that of majority class instances when aiming at minimizing the cost of misclassification (Bose & Chen, 2009; Fu et al. 2021). Ensemble learning methods solve imbalanced classification problem by highlighting the instances incorrectly classified in each iteration and combining the classification results from different base classifiers (Van den Poel & Buckinx, 2005). However, data-level approaches have long been criticized for adding too much noise or removing useful information in original dataset, while cost-sensitive learning methods may be difficult to implement since the prior information about misclassified cost is usually not available (Liu et al. 2022).

To sum up, identifying the minority responding customers based on the information of customers is of critical importance to successful direct marketing. However, the imbalance problem existing in the data makes the classification result biased towards the majority instances with ignorance on minority instances, leading to the low prediction accuracy of minority instances. To address this issue, we develop an Ensemble Learning with Dynamic Weighting (ELDW) approach to predict the probability of customers' response to the marketing activities. In the ELDW, there are two stages. The first stage is to train a number of diverse base classifiers, which is the premise of effective ensemble learning. In this stage, all the minority class instances are combined with different majority class instances, and a number of diverse subsets are constructed to train the base classifiers. In the second stage, the results of base classifiers for an unknown instance are dynamically integrated considering two factors are considered for each unknown instance. The first factor is the cross entropy of its neighbors in each subset for dynamically integrating the result of each base classifier, and the second factor is the feature similarity to the minority class instance that is helpful for strengthening the identification of instance from the minority class. After integrating the results in the above two factors, the probability of a customer responding to the marketing activities is obtained. In order to verify the effectiveness of ELDW, we compare its performance with some other state-of-the-art methods on 10 imbalanced benchmark datasets, and apply it to a real direct marketing activity to identify the target customers under a limited budget.

The rest of this paper is structured as follows. Section 2 reviews the related work. Section 3 introduces the proposed ELDW approach. Section 4 reports the experimental studies and comparison results. The ELDW approach is applied to the direct marketing in section 5. Finally, we end this paper with conclusions in section 6.

2. Related work

The work in this paper is related to three streams of research, i.e., target customer identification, imbalance classification and instance-based learning. In this section, we provide a comprehensive review of the related literature, and summarize our contributions.

2.1 Target customer identification

Target customer identification can be realized by building a response model. According to Bose & Chen (2009) and the approaches developed in recent years, we divide the techniques of identifying target customers into two mainly categories, i.e., statistical techniques and machine learning techniques, as shown in Fig. 1.

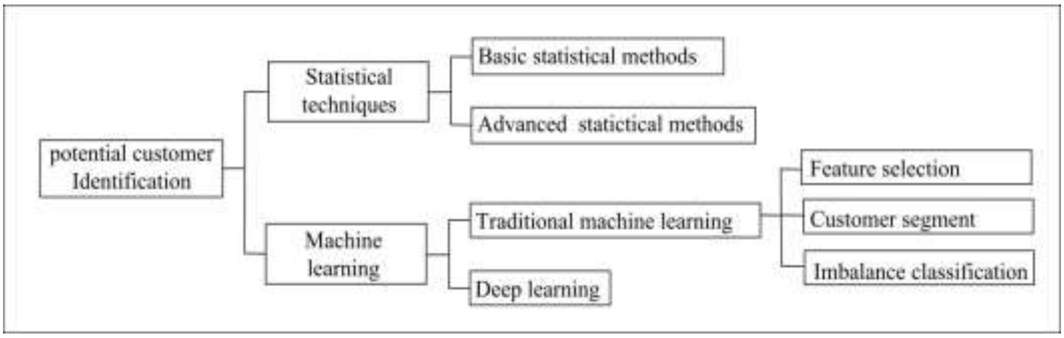

Fig. 1 Categories of techniques for identifying target customers

In Fig. 1, statistical techniques include the basic statistical techniques and advanced statistical techniques. On the other hand, machine learning techniques include conventional machine learning algorithms (such as logistic regression, decision tree, support vector machines and so on) and deep learning algorithms. In conventional machine learning algorithms, some researchers use feature selection methods to identify the key features of customers, while other researchers adopt clustering methods to segment the customers, and the target customers are identified in each segment. The representative literature of different categories of techniques is summarized in Table 1.

Table 1 Representative literature on identifying customers

Category	Complete name	Reference	Techniques	Contribution
Statistic technique	Predicting Online-purchasing Behavior	Van den Poel & Buckinx (2005)	Basic statistical technique: Logit	It identifies the impact of different factors on purchasing behavior. And it uses logit modelling to predict whether or not a purchase is made during the next visit.
	Allocation of Catalogs to Collective Customers Based on Semiparametric Response Models	Baumgartner and Hruschka (2005)	Advance statistic technique: Probit + Non-linear	It proposes a profit maximization customers' response model based on purchase, purchase value, sending back goods and the value of returned goods, to maximize the firm's profit.
Machine learning	Mutual Information and Sensitivity Analysis for Feature Selection in Customer Targeting: A Comparative Study	Nestor et al. (2019)	Feature selection	It focuses on analyzing the advantages and disadvantages of using mutual information and sensitivity analysis for feature selection in the classification problems of bank telemarketing.
	Predicting the Response of Segmented Customers for the Promotion Using Data Mining	Hong and Kim (2010)	Customer segment	It proposes a method that segments customers by utilizing self-organizing map and predicts the customers' response in each customer segment by integrating logistic regression, neural networks and support vector machines.
	Predicting Direct Marketing Response in Banking:	Miguéis et al. (2017)	Imbalance classification	It explores imbalance classification methods in the bank telemarketing. The performance of an under-

Category	Complete name	Reference	Techniques	Contribution
	Comparison of Class Imbalance Methods			sampling method is compared with that of an oversampling method in order to determine the most appropriate specification. And the importance of features is also explored.
	Direct Marketing Campaigns in Retail Banking with the Use of Deep Learning and Random Forests	Ładyżyński et al. (2019)	Deep learning	This paper proposes a model based on random forests and deep neural networks to identify customers interested in credit products. And this model can extract significant patterns from customers historical data and predict the probability of credit purchase.

In statistical technique, regression model is the most commonly used basic statistical technique, where the coefficients represent the influence of explanatory variables. The advanced statistical techniques combine two or more basic models together and leverage the strength of each model. However, different statistical techniques require different assumptions, and violation of those assumptions would cause inaccurate estimation of the parameters. In machine learning technique, feature selection can select most relevant features to distinguish the responding customers form no-responding customers. Whereas influenced by consumer psychological factors, the customers with the same feature values may make different decisions, reducing the accuracy of prediction model.

2.2 Imbalance Classification

There has been a large number of approaches proposed for imbalanced classification problem. Generally, these approaches can be divided into three categories: data-level approaches, algorithm-level approaches and hybrid approach (Gao et al. 2020). The framework of approaches to imbalance classification is shown in Fig. 2.

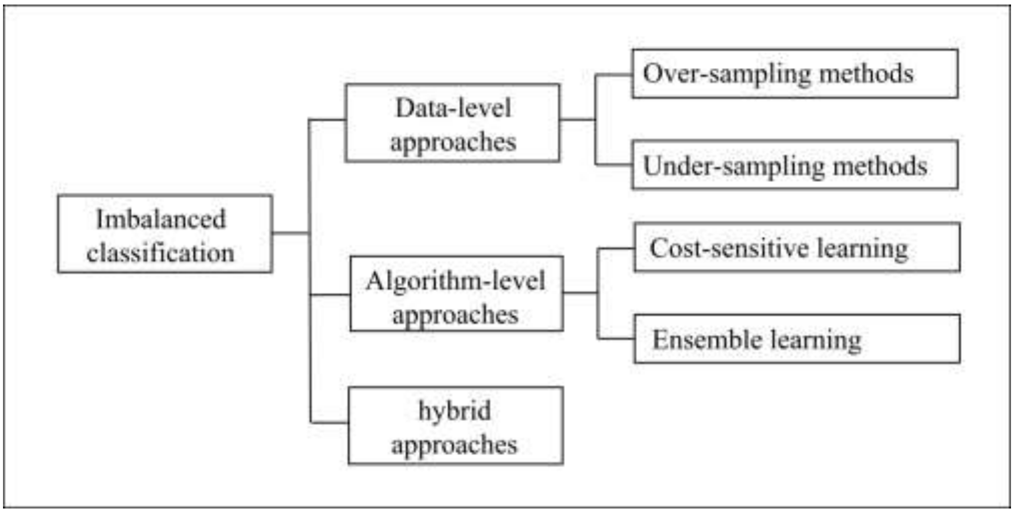

Fig. 2 Categories of approaches in imbalance classification

In Fig. 2, the data-level approaches utilize the data preprocessing techniques, such as oversampling the minority class instances or under-sampling the majority class instances, to balance the original dataset (Liu et al. 2022). The algorithm-level approaches include cost-sensitive learning and ensemble learning. The cost-sensitive learning penalizes higher in the misclassification of minority class than that in the majority class when minimizing the overall training error (Zhang 2020; Ng et al. 2021). The ensemble learning solves the imbalance classification problem by combining the results of base classifiers. It can be divided into the bagging-based and the boosting-based method (Gao et al. 2020). To gain a comprehensive review, we summarize some representative literature on imbalance classification in Table 2.

Table 2 Representative literature on imbalance classification

Category	Complete name	Reference	Techniques	Contribution
Data-level approach	Smote: Synthetic Minority Over-Sampling Technique	Chawla et al. (2002)	Over-sampling methods	For each minority class instance, SMOTE generates k synthetic instances on the lines between this minority instance and its k nearest minority class neighbors.
	Local Distribution-based Adaptive Minority Oversampling for Imbalanced Data Classification	Wang et al. (2021)	Over-sampling methods	This method is proposed to deal with two issues: where and how many synthetic instances should be generated. It first identifies the informative borderline minority instances as sampling seeds. Then, it captures the local distribution of each seed and generates synthetic instances around seeds via a Gaussian mixture model.

Category	Complete name	Reference	Techniques	Contribution
	Neighborhood-based Undersampling Approach for Handling Imbalanced and Overlapped Data	Vuttipittayamongkol & Elyan (2020)	Under-sampling methods	This method handles imbalance problem by removing potential overlapped instances. Four undersampling methods are proposed. They are basic neighborhood search, modified tomek link search, common nearest neighbors Search and Recursive Search. The methods vary in terms of local search criteria and negative instances elimination.
	Uffdfr: Undersampling Framework with Denoising, Fuzzy C-means Clustering, and Representative Instance Selection for Imbalanced Data Classification	Zheng et al. (2021)	Under-sampling methods	A novel three-stage undersampling framework is proposed. The first stage is to eliminate noisy, boundary and redundant majority class instances. The second stage uses the fuzzy c-means clustering algorithm to cluster majority class instances into multiple clusters. The third stage selects the representative majority class instances vital clustering-based undersampling methods to screen the representative instances and filter out the unrepresentative instances from each cluster.
Algorithm-level approach	Cost-sensitive KNN Classification	Zhang (2020)	Cost-sensitive learning	Two efficient Cost-Sensitive (CS) KNN classification models, referred to Direct-CS-KNN classifier and Distance-CS-KNN classifier, are designed for making KNN classifier sensitive to misclassification costs in imbalanced classification. And the two CS-KNN classifiers are further improved with extant strategies, such as smoothing, minimum-cost k-value selection, feature selection and ensemble selection.
	CS-ResNet: Cost-sensitive Residual Convolutional Neural Network for PCB Cosmetic Defect Detection	Zhang et al. (2021)	Cost-sensitive learning	In cosmetic defect detection, a novel model called cost-sensitive residual convolutional neural network (CS-ResNet) is proposed by adding a cost-sensitive layer in the standard ResNet. Specifically, larger weights to minority real defects are assigned based on the class-imbalance degree and then CS-ResNet is optimized by minimizing the weighted cross-entropy loss function.
	A Dynamic Ensemble	Guo et al. (2021)	Ensemble learning	An ensemble learning algorithm based on K-means sampling and

Category	Complete name	Reference	Techniques	Contribution
	Learning Algorithm based on K-means for ICU Mortality Prediction			distance-based dynamic ensemble is proposed. K-means sampling achieves the diversity of base classifiers and the distance-based dynamic ensemble is a flexible fusion method which creates a personalized combination of results from base classifiers for each testing instance.
	Adaptive Ensemble of Classifiers with Regularization for Imbalanced Data Classification	Wang et al. (2021)	Ensemble learning	First, it leverages the properties of stochastic gradient descent to obtain the solution with the minimum norm to achieve regularization; furthermore, it interpolates the ensemble weights by exploiting the global geometry of data to further prevent overfitting.
Hybrid approach	An Ensemble Imbalanced Classification Method based on Model Dynamic Selection Driven by Data Partition Hybrid Sampling	Gao et al. (2020)	Ensemble+Sampling	First, the data space is divided into four regions according to the majority class proportion in minority class neighborhoods. Then a boundary minority class weighted over-sampling method is applied. Next, three ensemble learning models are built to be selected adaptively according to the imbalance degree of neighbors.
	A Hybrid Data-level Ensemble to Enable Learning from Highly Imbalanced Dataset	Chen et al. (2021)	Hybrid data-level	It filters out unrepresentative majority instances based on an unsupervised margin definition, while the oversampling method generates diverse minority instances according to the behavior of ensemble learning. The combination of the two data-level approaches serves a twofold purpose of balancing the data distribution and optimizing the fundamental properties of the ensemble.

Although the data-level approaches balance the original dataset, they ignore useful information or alter the original data distribution. The oversampling methods may expand the minority class boundary and worsen the class imbalance problem by generating too many similar instances (Chen et al. 2021). On the other hand, the under-sampling methods tend to miss much useful information, especially when the imbalanced ratio is high (Gao et al. 2020). In algorithm-level approaches, the cost-sensitive learning method may be difficult to implement, since the setting of the misclassification cost is usually not available. It is highly domain related and often given by experts (Gao et al. 2020; Chen et al. 2021). In ensemble learning methods, static ensemble learning combines the result of base classifier in a global way, while the dynamic ensemble learning can integrate the result of base classifier in a personalized way (Guo et al. 2021). And thus, dynamic ensemble learning methods are the main research direction of scholars in recent years. Hybrid approaches combine the data-level approaches and algorithm-level approaches to avoid the disadvantages caused by a single method.

2.3 Instance-based Learning

Instance-based learning, a typical lazy learning mechanism, is a commonly used nonparametric learning algorithm (Kang and Cho, 2008). It only stores training instances before the arrival of a new instance, whose target value is determined by compared this instance with the stored instances. The instance-based learning has the following advantages: (1) flexibility: It can deal with complex and diverse data without prior assumptions on data. (2) Adaptability: it can learn and adapt to the changes in data, which makes it perform well in dealing with non-stationary data and online data. (3) Interpretability: its prediction is based on similarity comparisons with existing instances, making results easier to understand and interpret. (4) Robustness: it works relatively well with noisy data because it can reduce the impact of a single instance by comparing the similarity of multiple instances.

Due to the above advantages, the instance-based algorithms are widely used in labeling data, filling data, and improving other machine learning algorithms. In recent years, multi-instance learning by comparing similarity at the packet level has been widely used in the fields of target detection, image labeling and so on. The representative literature on instance-based learning is shown in Table 3 below.

Table 3 Representative literature about imbalance classification

Category	Complete name	Reference	Contribution
Label/Impute the unknown instances	Online state-of-health prediction of lithium-ion batteries with	Yu et al. (2020)	A novel semi-supervised learning framework is proposed to estimate the capacity of batteries. Among them, an improved locally linear

Category	Complete name	Reference	Contribution
	limited labeled data		reconstruction method, that is, an instance-based learning method, is used to determine the capacity distributions of the unlabeled data.
	K-nearest neighbor imputation based on sparse coding	Su et al. (2015)	K-nearest neighbor based on sparse coding algorithm is present to impute the missing values. This method reconstructed each missing instance with the training instances, fully considering the correlation between instances in the reconstruction process. And it used an e_1 norm to ensure each missing instance imputed by different number of training instances.
Improving other algorithms	Lazy Learning for Norparametric Locally Weighted Regression	Roh et al. (2020)	A newly designed local model is proposed to predict the output of new data. The model exploits the concept of the nearest neighbor, and constructs the weighted least square estimation once a new query is provided given.
	A case based method to predict optimal k value for k-NN algorithm	Yang et al. (2017)	Instance-based learning method is used to determine the optimal K value. A local complexity was computed for each example and a complexity profile was constructed by sorting these local complexity values. And then, a feature vector was built by combing the local complexity profile and some statistic features of a data set, and the label is optimum k value. The feature vector and the label form the meta-data. A predict model was trained on meta-data set to predict optimum k value.
Multi-instance learning methods	A linear programming approach to multiple instance learning	Kucukasci et al. (2021)	A linear programming framework is proposed to learn instance level contributions to bag label without exposing the standard assumption. Each instance of a bag is mapped to a pseudo-class membership estimate and these estimates are aggregated to obtain the bag-level class membership in an optimization framework. A simple linear mapping enables handling various MIL assumptions with adjusting instance contributions.
	A Transfer Learning-Based	Xiao et al. (2022)	A new framework called transfer learning-based multiple instance

Category	Complete name	Reference	Contribution
	Multi-Instance Learning Method With Weak Labels		learning framework is proposed to address the problem of multiple instance transfer learning in which both the source task and the target task contain the weak labels. First, an iterative framework to solve the transfer learning model with weak labels is put forward to update the label of the bag to improve the performance of multiple instance learning. And then the convergence analysis of the proposed method is introduced.

Since instance-based learning compares the similarity between unknown instance and its neighbors to obtain local information for classification or prediction, it is widely used in industry, medicine, and other fields. In this paper, instance-based learning is also used to explore the local information of an unknown instance to solve the imbalance classification problem, in which the minority class instances are difficult to be recognized accurately.

2.4 Our Contributions

Identifying the target customers in direct marketing is an imbalance classification problem. The customers who respond to the marketing activities are regarded as the minority class instances, and the customers who do not respond are the majority class. In this study, we propose a customer response model based on Ensemble Learning with Dynamic Weighting (ELDW). Compared with previous works, our main contributions are as follows.

- An important assumption of ensemble learning to solve imbalance classification problem is that the base classifiers are diverse, i.e., the feature spaces of training subsets are different from each other. In this paper, we combine the minority class instances with different majority class instances to form the diverse subsets and generate the base classifiers. Consequently, the base classifiers capture the different patterns in each subset. Besides, the minority class instances in each subset are similar with each other, making the base classifier learn more information from the minority class instances over all the subsets.

- When predicting the class of the unknown instance, ELDW uses the cross entropy of its neighbors in each subset to integrate the result of each base classifier dynamically. This dynamic ensemble process considers local information in each subset to improve the overall performance and robustness. The reason for using cross entropy to integrate the result of each base classifier is that if a base classifier can predict the neighbors of an unknown instance accurately, it can also predict the unknown instance accurately. When the base classifier predicts neighbors accurately, the neighbors' cross entropy is not high. Therefore, the lower the cross entropy is, the more accurate the prediction of corresponding base classifier is.
- In order to enhance the identification of minority class instance, ELDW uses instance-based learning to explore the feature similarity of unknown instance to the minority class neighbors in each subset. If an unknown instance always has a high feature similarity with minority neighbors in each subset, it is more likely to be the minority class instance. Finally, the customers' response probability is determined by combining the dynamic ensemble learning and instance-based learning.

3. The proposed method

The Ensemble Learning with Dynamic Weighting (ELDW) proposed in this study is designed to build the customers' response model on imbalanced dataset. As shown in Fig. 3, there are two primary stages in this ensemble learning model. One is to generate a number of diverse base classifiers, and the other is to integrate the result of base classifier dynamically. At last, each unknown instance will get a weighted predicted probability, that is, the probability of customers' response to marketing activities.

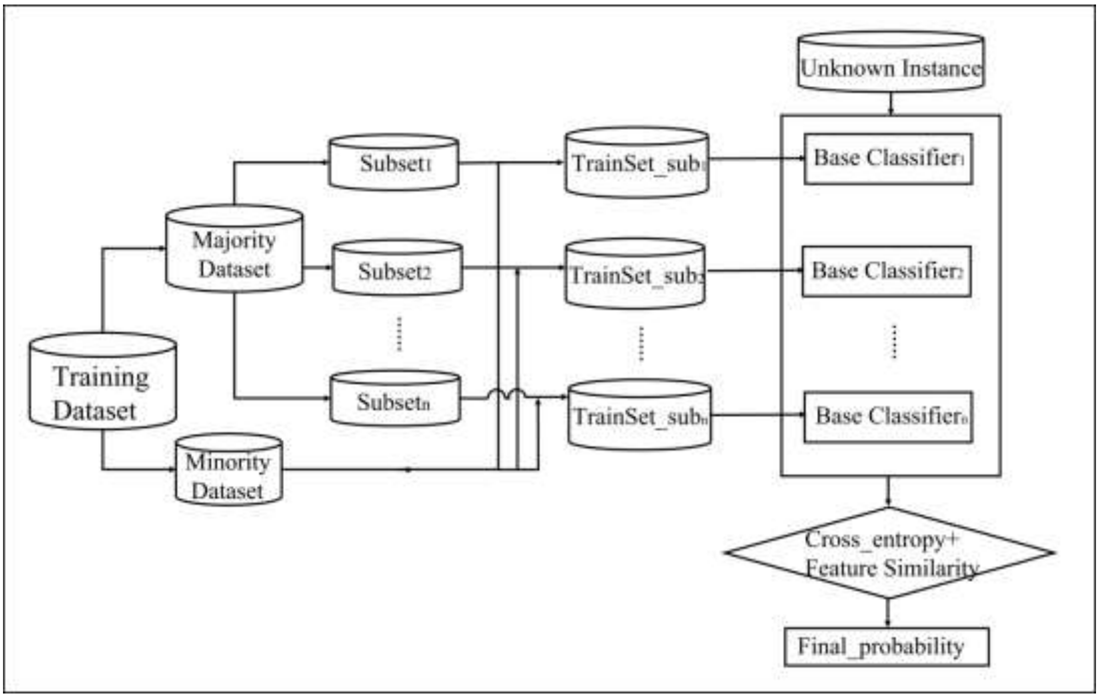

Fig. 3 The framework of the proposed ELDW

3.1 Base classifier generation

The first stage of our approach is to train a number of diverse base classifiers. In this stage, the majority class instances are divided into several subsets according to whether they are in the neighborhood of the minority class instances or not. And then, the majority instances in each subset are combined with all the minority class instances to form a new subset to train the base classifier.

Suppose $T = \{<\mathbf{X}_1, y_1>, <\mathbf{X}_2, y_2>, \cdots, <\mathbf{X}_i, y_i>, \cdots, <\mathbf{X}_N, y_N>\}$ is the training set, where N represents the number of training instances. Among them, $\mathbf{X}_i = \{x_i^1, x_i^2, \cdots, x_i^d\} \in R^d, (i = 1, 2, \cdots N)$ is a d-dimension instance, and $y_i \in \{1, 0\}$ represents the class label, where 1 is the label of minority class and 0 is the label of majority class.

Step 1: the training set is divided into a minority class set and majority class set according to the label of instances. The two sets are denoted as $minSet$ and $majSet$, respectively;

Step 2: the imbalance ratio between $majSet$ and $minSet$ is calculated, denoted as IR, as shown in Eq. (1):

$$IR = \mathrm{round}(\frac{\#majInstances}{\#minInstances}) \quad (1)$$

In Eq. (1), $\#majInstances$ and $\#minInstances$ represent the number of instances in $majSet$ and $minSet$, respectively.

Step 3: for instances in $minSet$, their IR nearest neighbors in $majSet$, are denoted as $neighborSet_{maj}$;

Step 4: other majority class instances not included in $neighborSet_{maj}$ form another dataset, denoted as $outsideSet_{maj}$;

Step 5: $outsideSet_{maj}$ and $minSet$ form a new subset to train a base classifier;

Step 6: $neighborSet_{maj}$ is regarded as a new $majSet$. Repeat step 2-5 until IR=1, and the new $majSet$ and $minSet$ form the last subset to train a base classifier;

The steps of base classifiers generation is shown in Fig. 4.

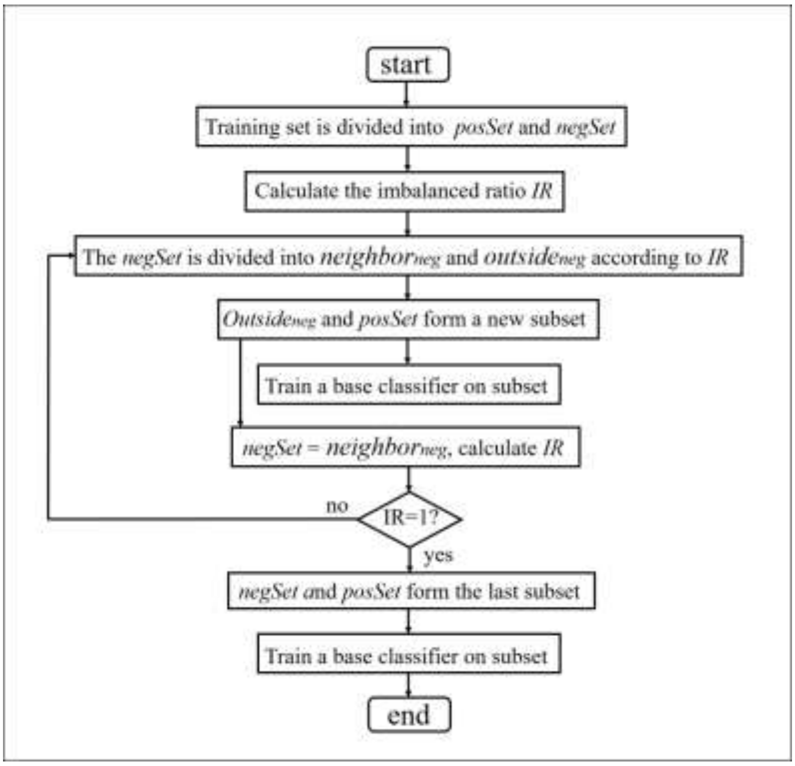

Fig. 4 The steps of base classifiers generation

A two-dimension imbalanced classification problem, as shown in Fig. 5, is used to illustrate the main idea of base classifiers generation. The circular instances represent the majority class instances, and the triangular instances represent the minority class instances.

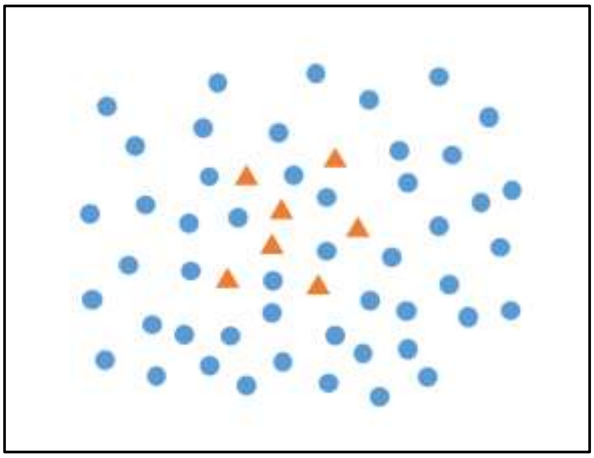

Fig. 5 An illustration of a two-dimension imbalanced classification problem

For the classification problem illustrated in Fig. 5, we show the detailed steps of base classifiers generation in Fig. 6.

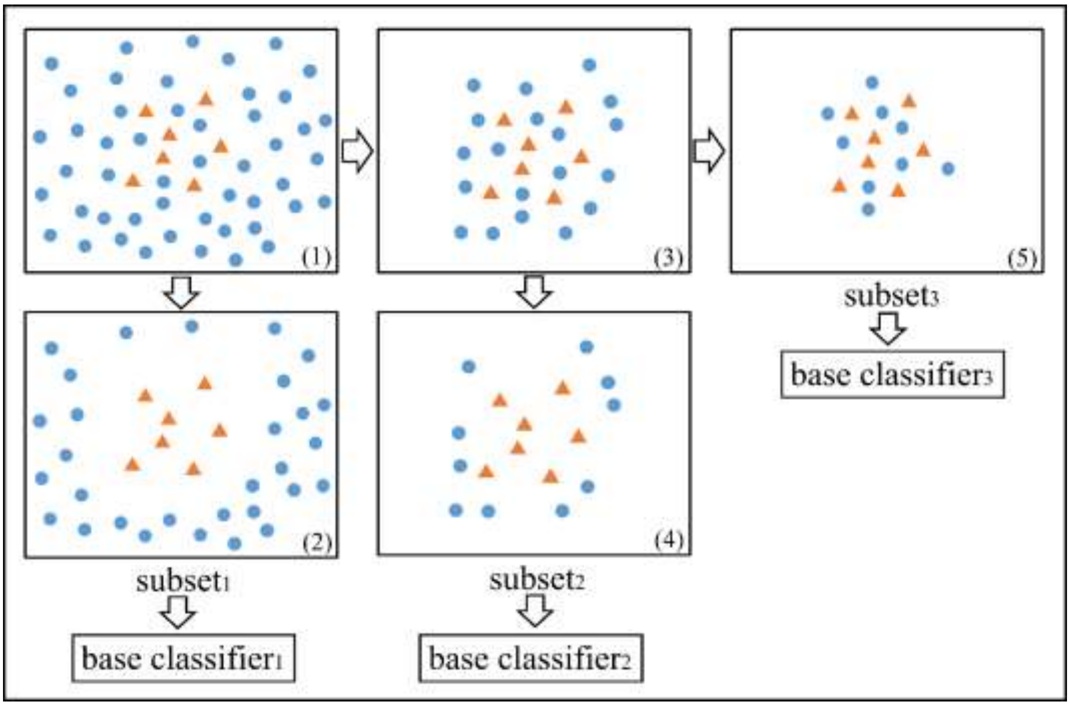

Fig. 6 The detailed steps of generating base classifiers

In Fig. 6(1), the imbalance ratio of original dataset is calculated as IR= 7 (majority class instances: minority class instances = 49:7) according to step 2. For each minority class instance, 7 nearest neighbors are obtained according to step 3.

In Fig. 6(2), according to steps 4-5, the majority class instances not in the 7 nearest neighbors of minority class instances are combined with all the minority class instances to form $subset_1$.

In Fig. 6(3), the majority class instances that appear in the 7 nearest neighbors of the minority class instances and all the minority class instances form the new dataset, according to step 6. The imbalance ratio of this new dataset is obtained, i.e., IR=3 (majority class instances: minority class instances = 19:7). According to step 3, for each minority class instance, 3 nearest neighbors are obtained.

In Fig. 6(4), according to steps 4-5, the majority class instances, which never appear in the 3 nearest neighbors of minority class instances are obtained, are combined with all the minority class instances form $subset_2$.

In Fig. 6(5), the majority class instances that appear in the 3 nearest neighbors of the minority class instances and all the minority class instances form the new dataset. The imbalance ratio of this new dataset is obtained IR=1 (majority class instances: minority class instances = 9:7). And thus, this new dataset forms the $subset_3$, and each obtained subset can be used to train a base classifier.

It can be seen from Fig. 6 that each subset contains all the minority instances and different majority instances. On one hand, the way that all minority class instances participate in the training of each base classifier can enhance the learning on minority class instances. On the other hand, the majority class instances and the class boundary in each subset are different, ensuring the diverse of subset in ensemble learning. The clear decision boundary between the majority and minority classes in each subset is helpful to reduce the over-fitting issues.

In summary, the pseudo codes of base classifier generation of the proposed ELDW approach are shown in Algorithm 1.

Algorithm 1:

Input: *TR*: training dataset;

Output: *classifierList*: a list of base classifiers; *subsetList*: a list of subsets

Phase1: generate diverse base classifiers

```
minTR = Ø; majTR = Ø; outsideSet_maj = Ø; classifierList = Ø; subsetList = Ø;
for <X_i, y_i> ∈ TR, do:
    if (y_i ==1):
        minTR .add(<X_i, y_i>);
    else:
        majTR.add (<X_i, y_i>);
end for
while (1>0): {
    IR=round(len(majTR)/ len(minTR)); #imbalance ratio determines the number of neighbors
    if (IR==1): #when dataset is balanced, the last base classifier is trained
        subset= majTR + minTR #form a new subset
        subsetList.add (subset);
        classifier.train (subset); #train a new classifier on the subset
        classifierList.add (classifier);
        break;
    else:
        neighborSet_maj = Ø; #store the majority neighbors of each minority instance
        for <X_i, y_i> ∈ minTR, do :
            neighbors=getNeighbors (<X_i, y_i>, TR, IR); #find neighbors in TR
            for <X_j, y_j>∈ neighbors, do:
                if (y_j==0): #find majority neighbors
                    neighborSet_maj.add(<X_j, y_j>); #store majority neighbors
            end for
        end for
        neighborSet_maj.unique(); # remove duplicate instances
        for <X_i, y_i> ∈ majTR, do:
            if <X_i, y_i> not in neighborSet_maj:
                outsideSet_maj.add (<X_i, y_i>); #store the majority instances not in the neighbors
            end if
        end for
        subset= outsideSet_maj + minTR #form a new subset
        subsetList.add (subset);
        classifier.train (subset); #train a new classifier on the subset
        classifierList.add (classifier);
        negTR= neighborSet_maj; #remaining majority instances form a new majority dataset
}
```

3.2 Ensemble with dynamic weighting

The second stage of our approach is to integrate the result of each base classifier dynamically. In this stage, two factors are considered for each unknown instance (instance whose class label is unknown). The first factor is the cross entropy of its neighborhood in each subset. It is assumed that if the base classifier could predict the class label of an unknown instance accurately, it can also predict its neighbors accurately as well. In this situation, the cross entropy of neighborhood will be low. Therefore, the result of each base classifier can be integrated based on its cross entropy. The second factor is the average feature similarity to the minority class instances, which is used to enhance the ability of minority class instances identification. Since the minority class instances in each subset are the same, if an unknown instance always has a high feature similarity with the minority class instances in each subset, it is more likely to be the minority class. Through the combining the above two aspects, the minority instances can finally obtain a higher predicted probability.

Each unknown instance gets its final predicted probability as shown in Fig. 7. First, each base classifier outputs its predicted probability. And then, the cross entropy of neighborhood and the feature similarity to the minority instances in each subset are obtained. At last, the final predicted probability is output based on the above two factors.

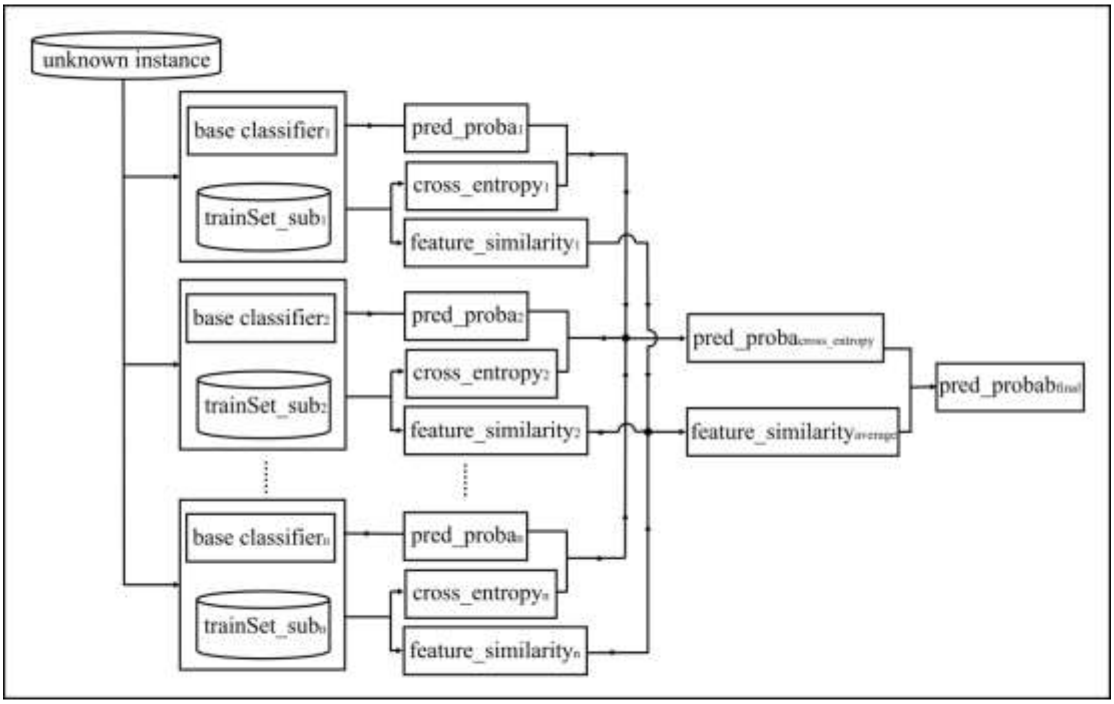

Fig. 7 The prediction of unknown instance

3.2.1 Integrating base classifiers based on cross entropy

For an unknown instance, its cross entropy of neighbors in each subset can be obtained. The low cross entropy indicates that the base classifier can predict accurately. In this case, the result of base classifier should be given a high weight when it is integrated.

The calculation of cross entropy in binary classification is shown as follows.

$$BinaryCrossEntropy = \sum_{j=1}^{k} -P_j \log \tilde{P}_j - (1-P_j)\log(1-\tilde{P}_j) \tag{2}$$

In Eq. (2), j is the index of the neighbor, P_j is the class label of the jth neighbor, and $\tilde{P}_j$ is the predicted probability of the jth neighbor by the classifier.

The detailed steps of integrating the results of base classifiers are as follows:

Step 1: for an unknown instance X, the predicted probability by each base classifier and the cross entropy of its neighbors in each subset are obtained, which are denoted as $predict\_proba_i$ and $crossEntropy_i$, respectively, and i is the index of the subset;

Step 2: the weight of each classifier is obtained. We use the reciprocal of cross entropy as the weight of base classifier, which is calculated by Eq. (3).

$$entropy\_weight_i = (1/crossentropy_i) \Big/ (\sum_{i=1}^{n} 1/crossentropy_i) \tag{3}$$

In Eq. (3), n is the number of base classifiers.

Step 3: the result of each base classifier is integrated by Eq. (4).

$$pred\_proba_{integration} = \sum_{i=1}^{n} predict\_proba_i * entropy\_weight_i \tag{4}$$

In this way, the predicted probability of unknown instance is integrated based on the cross entropy.

3.2.2 Enhancing the identification of minority class instances based on feature similarity

Inspired by the idea that "if an unknown instance is described well with its local neighbors, its class label can be predicted or estimated by the label of neighbors as well" (Kang and Cho, 2008), we use the feature similarity to the minority class neighbors to enhance the identification of minority instances. Recall that according to the base classifier generation, the minority class instances in different subsets are the same, while the majority class instances in different subsets are different. If an unknown instance has a high feature similarity with minority neighbors in each subset, it is more likely to be of the minority class.

The feature similarity to the minority class neighbors is obtained by Local Liner Reconstruction (LLR), which has been introduced in detail in Kang and Cho (2008). The main idea of LLR is to use local neighbors to reconstruct the unknown instance in a structured way, and the main steps are described as follows:

Step 1: when an unknown instance X arrives, its k nearest neighbors are found, as shown in Fig. 8(2);

Step 2: the unknown instance X is described by its neighbors. To find the best linear combination and the weights of neighbors, the following reconstruction error is $E(w)$ should be minimized:

$$\text{Min} \quad E(w) = \frac{1}{2}\left| X - \sum_{j=1}^{k} w_j X_j \right|^2$$

(5)

s. t.

$$\sum_j w_j = 1 \qquad (6)$$

$$w_j \geq 0 \qquad (7)$$

In Eq. (5), X_j is the feature of *j*th neighbor in a subset and w_j is the weight of *j*th neighbor.

Step 3: The minimization problem can be solved by any algorithm developed for Quadratic Programming (QP) to obtain the value of w_j, as shown in Fig. 8 (3).

Step 4: In each subset, the feature similarity between unknown instance and positive neighbors is calculated as Eq. (8):

$$feature\_similarity_i = \sum_{j=1}^{k} w_j y_j \qquad (8)$$

In Eq. (8), *i* is the index of corresponding subset. Since the class labels of positive and negative instances are 1 and 0, respectively, $feature\_similarity_i$ can be regarded as the feature similarity between unknown instance and the positive neighbors.

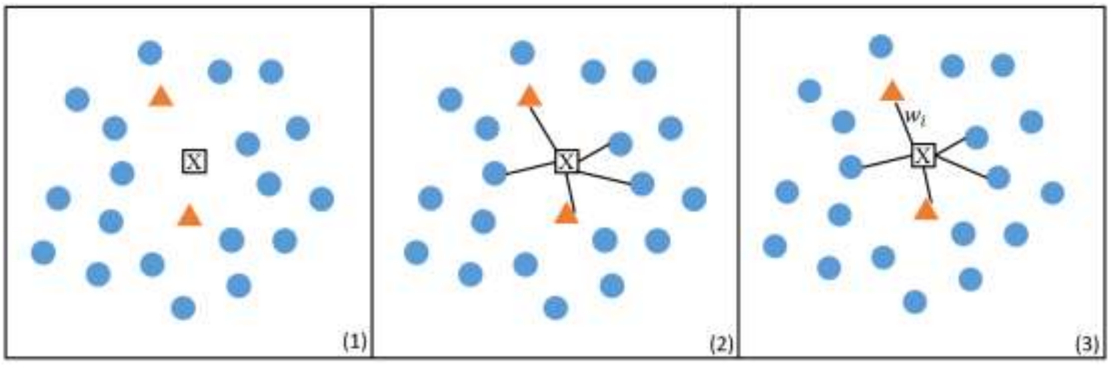

Fig. 8 Illustration of the LLR

In this way, the weight of each neighbor is obtained based on LLR, and this weight can be used to describe the feature similarity to be the positive instance. In addition, Kang and Cho (2008) has proved that this method is robust to the value of *k*, and when the *k* is set to a sufficiently large number, a small reconstruction error can be guaranteed.

Since the positive instances in different subsets are same, the average similarity to positive instance can be obtained by combining feature similarity in each subset as:

$$feature\_similarity\_ava = \left(\sum_{i=1}^{n} feature\_similarity_i\right) \Big/ n \tag{9}$$

The $feature\_similarity\_ava$ can be used to strengthen the identification of positive instance. So far, combined with the dynamic weight based on cross entropy and feature similarity based on LLR, the final predicted probability of the unknown instance can be calculated as Eq. (10).

$$final\_pred\_proba = pred\_proba\_ava_{entropy} * feature\_similarity\_ava \tag{10}$$

In summary, the pseudo codes of ensemble learning with dynamic weighting are shown in Algorithm 2.

Algorithm 2:

Input: *classifierList*; *subsetList*; unknown instance X;

Output: *final_pred_proba*: the final predicted probability of X

entropyWeightList = Ø; *similarityList* = Ø; *predProbaList* = Ø

for $subset_i \in subsetList$, do:

    neighbors = getNeighbors (X, $subset_i$, *k*); #get neighbors in each subset

    neighbors_pred = *classifierList*[i].pred(*neighbors*) #neighbors' predicted probability

    $crossentropy_i$= getEntropy (*neighbors*, *neighbors_pred*) # the cross entropy of neighbors

    entropyWeightList.append(1/$crossentropy_i$) # the weight based on cross entropy

    $w_1, \dots w_j, \dots, w_k$ = getLLRWeights(*neighbors*, X); # the weight of neighbors

    $feature\_similarity_i = \sum_{j=1}^{k} w_j * y_j$ #feature similarity to the positive instances in a subset

    similarityList.append($feature\_similarity_i$)

    $pred\_proba_i$= *classifierList*[i].pred_proba(X); #base classifier makes prediction

    predProbaList.append($pred\_proba_i$)

end for

$pred\_proba\_ava_{entropy}$=0

for $1/crossentropy_i \in entropyWeightList$, do: #weighted based on cross entropy

    $entropy\_weight_i = (1/crossentropy_i) \Big/ \left(\sum_{i=1}^{n} 1/crossentropy_i\right)$

    $pred\_proba\_ava_{entropy}$ += *predProbaList*[i]*$entropy\_weight_i$

end for

feature_similarity_ava= *similarityList*.average() # obtain average feature similarity

$final\_pred\_proba = pred\_proba\_ava_{entropy} * feature\_similarity\_ava$

4. Experiment study

In this section, we present the experimental studies on 10 benchmark datasets and compare the results. All the experiments are carried out on the Intel Core i5-8265U CPU, 1.6 GHz, 8 GB RAM PC and Windows 10 OS.

4.1 Data description

We select 10 benchmark datasets of binary classification from the KEEL dataset repository (https://sci2s.ugr.es/keel/datasets.php) that is usually used in the experimental study to compare the performance of machine learning algorithms. These datasets come from various fields in the real world and have different numbers of instances. It is worth noting that the imbalance ratio of most datasets is larger than 5. Usually, a more imbalanced dataset would cause larger effect on the classification results (Zhu et al. 2018). A brief description of these datasets is shown in Table 4.

Table 4 Description of the benchmark datasets

No.	Name	#Feature	#Instance	#Instances in each class	Imbalance ratio
1	Abalone9_18	8	731	42:689	16
2	Car_good	6	1728	69:1659	24
3	Ecoli2	7	336	77:259	3
4	Ecoli8	7	336	52:284	5
5	Glass6	9	214	29:185	6
6	Kr-vs-k-zero-one_vs_draw	6	2901	105:2796	27
7	Page-blocks0	10	5472	559:4913	9
8	Satimage	36	6435	626:5809	9
9	Vowel	13	990	90:900	10
10	Yeast1	8	1484	244:1240	5

To standardize the data, the min-max normalization method is adopted to map the feature values into the range from 0 to 1. To evaluate the classification performance, the standard 10-fold cross validation is adopted in the experiments. Final results are obtained by averaging the results over 10 runs.

4.2 Competing methods

We select five algorithms proposed in recent years for imbalanced classification problems as the competing methods. These five algorithms include an oversampling method, an undersampling method, a dynamic ensemble method and two hybrid methods that combine both the sampling and ensemble mechanisms. Namely, they are Local distribution-based Adaptive Minority Oversampling (LAMO) (Wang et al., 2021), Undersampling Framework with Denoising, Fuzzy c-means clustering, and Representative instance selection (UFDFR) (Zheng et al., 2021), Dynamic Ensemble Learning Algorithm based on K-means (DELAK) (Guo et al., 2021), ensemble imbalanced classification method based on model dynamic selection driven by Data Partition Hybrid Sampling (DPHS) (Gao et al., 2020) and Hybrid Data-level ensemble for highly imbalanced dataset (HD-ensemble) (Chen et al. 2021). All these methods have been briefly introduced in the literature review section. In the experiments, these methods and our ELDW adopt Logistic Regression (LR), Support Vector Machine (SVM), Naive Bayes (NB), and Multi-Layer Perceptron (MLP) as the base classifiers.

Since there is no parameter to be adjusted in our proposed ELDW, for the parameters in comparison methods, we use the general parameter setting or parameter setting recommended by the references. For the base classifier used in proposed ELDW and other competitors, such as LR, SVM, MLP and NB, the parameters in these base classifiers are set as default value in scikit-learn software.

4.3 Performance evaluation metrics

The confusion matrix is the basis of performance evaluation in binary classification. A typical confusion matrix is shown in Table 5.

Table 5 The confusion matrix

		Actual class label	
		positive	negative
Predicted class label	positive	TP	FP
	negative	FN	TN

In the confusion matrix, the rows denote the actual class labels of the instances and the columns denote the predicted classes. In a typical confusion matrix, True Positive (TP) is the number of positive instances that are correctly classified, whereas

False Positive (FP) is the number of negative instances that are incorrectly classified as positive. True negative (TN) is the number of negative instances that are correctly classified, and False Negative (FN) is the number of positive instances that are incorrectly classified as negative.

All performance evaluation metrics for binary classification can be derived from the confusion matrix. In this paper, the Receiver Operating Characteristic (ROC) curve, Area Under the Curve (AUC) are adopted as the performance evaluation metrics. The reason we do not adopt the overall classification accuracy as the evaluation metric is that in imbalance classification, when all the instances are predicted to be the majority class, the overall accuracy is still high, which has little meaning to the performance.

The ROC curve reveals the relationship between False Positive Rate (FPR) and True Positive Rate (TPR). It calculates a series of FPR and TPR by setting different thresholds, and then draws these values with FPR as the abscissa and TPR as the ordinate. FPR and TPR are calculated by Eq. (11) and Eq. (12), respectively.

$$\mathrm{FPR} = \mathrm{FP}/(\mathrm{FP}+\mathrm{TN}) \tag{11}$$

$$\mathrm{TPR} = \mathrm{TP}/(\mathrm{TP}+\mathrm{FN}) \tag{12}$$

A typical ROC curve is shown in Fig. 9. For each point on the ROC curve, its abscissa and ordinate represent the FPR and TPR under the corresponding threshold, respectively. With the decrease of threshold, TPR and FPR are increased. The larger the area under the ROC curve, the better the performance of the model. That is to say, the model can obtain a larger TPR value with a smaller FPR value.

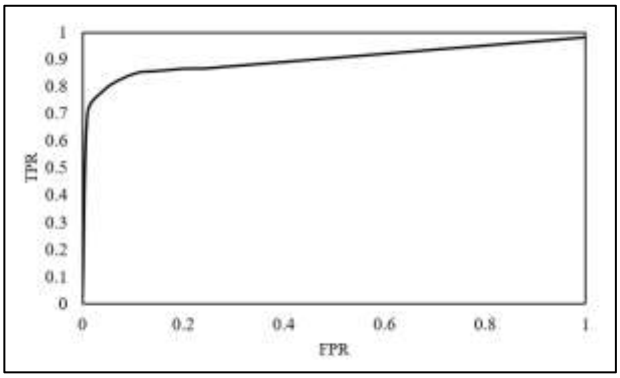

Fig. 9 A typical ROC curve diagram

The area under the ROC curve is the other metric, i.e., the AUC. The larger the value of AUC is, the better the performance of the model is. Obviously, the value of AUC cannot be greater than 1.

4.4 Results on benchmark datasets

In this section, we compare our ELDW with five competing algorithms on the 10 benchmark datasets. In the experiments, we use LR, SVM, NB and MLP as base classifiers.

(1) LR as the base classifier

When the LR is adopted as the base classifier, the ROC curve of each algorithm on each dataset is shown in Fig. 10. Among them, the red line in each subgraph is the ROC curve of ELDW.

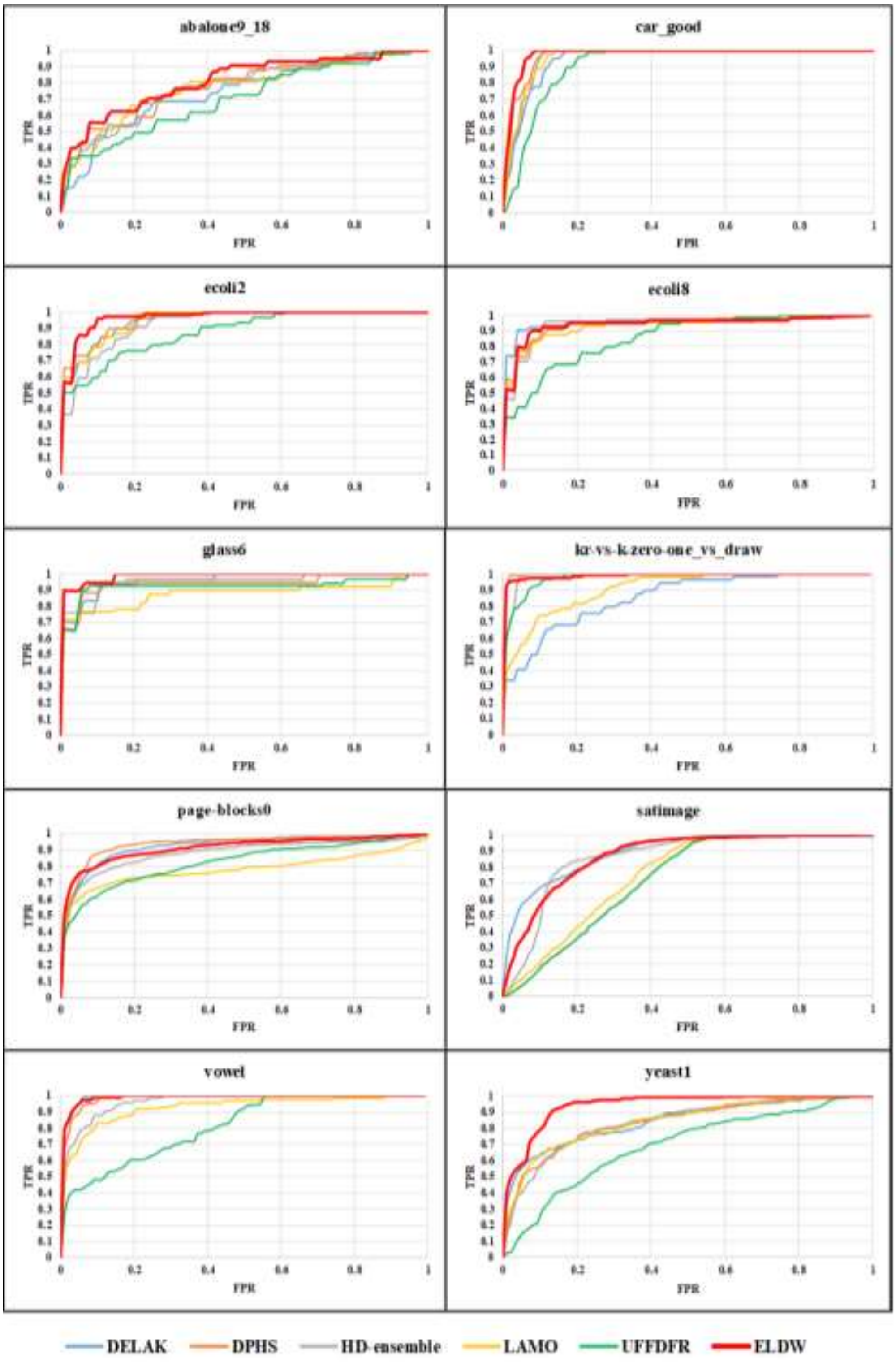

Fig. 10 ROC curves of algorithms on benchmark datasets (LR as base classifier)

It can be seen from Fig. 10 that ELDW can achieve the highest ROC curve on most datasets. In other words, ELDW can achieve a higher TPR value than other five algorithms when the FPR value is same. In addition, the performance of other algorithms on each data set is unstable. For example, as shown by the green line in each subgraph, UFFDFR algorithm performs well on the Glass6 and Kr-vs-k-zero-one_vs_draw datasets, but performs poorly on other datasets.

To present a quantitative comparison, we report the AUC value of each algorithm over each dataset in Table 6, where the highest AUC value for each dataset is bolded and unlined, and the average AUC value (mean AUC) and AUC variance (vari AUC) of each algorithm over all datasets are shown in the last two rows. The last column shows the ranking of our ELDW when the AUC values of different methods on each dataset is ordered.

Table 6 AUC value of each algorithm on each dataset (LR as base classifier)

	DELAK	DPHS	Hd_ensemble	LAMO	UFFDFR	ELDW	Rank
Abalone9_18	0.748	0.783	0.776	0.777	0.738	**<u>0.813</u>**	1
Car_good	0.947	0.970	0.957	0.956	0.910	**<u>0.976</u>**	1
Ecoli2	0.951	0.956	0.926	0.952	0.877	**<u>0.970</u>**	1
Ecoli8	**<u>0.954</u>**	0.946	0.943	0.934	0.857	0.943	3
Glass6	0.961	0.945	0.954	0.882	0.963	**<u>0.971</u>**	1
Kr-vs-k-zero-one_vs_draw	0.993	**<u>0.997</u>**	0.986	0.911	0.977	0.992	3
Page-blocks0	0.924	**<u>0.937</u>**	0.882	0.783	0.826	0.914	3
Satimage	**<u>0.893</u>**	0.863	0.726	0.754	0.632	0.873	2
Vowel	0.989	0.986	0.964	0.929	0.813	**<u>0.992</u>**	1
Yeast1	0.845	0.839	0.834	0.844	0.694	**<u>0.944</u>**	1
mean AUC	0.920	0.922	0.895	0.872	0.829	**<u>0.939</u>**	1
vari AUC	0.075	0.070	0.088	0.077	0.113	**<u>0.057</u>**	1

It can be seen from Table 6 that ELDW obtains the highest AUC values on 6 out of 10 benchmark datasets when the LR is adopted as base classifier. As for the datasets where AUC of ELDW is not the highest, the performance of ELDW is also competitive. For examples, on the Satimage dataset, the ranking of ELDW is 2, and on the remaining datasets, the ranking of ELDW is 3. Considering the performance overall datasets, ELDW obtains the highest average AUC value and the lowest variance of AUC, indicating that the results of ELDW are more stable.

We apply the Wilcoxon's signed-rank test to estimate the statistical differences of these algorithms over all the datasets. The p-values of Wilcoxon's signed-rank test are reported in Table 7.

Table 7 The p-values of Wilcoxon's signed-rank test (LR as base classifier)

	DELAK	DPHS	Hd_ensemble	LAMO	UFFDFR
ELDW	0.275	0.083	0.007	0.002	0.002

The results in Table 7 illustrate that the differences between ELDW and other four competitors (DPHS, Hd_ensemble, LAMO, UFFDFR) is significant at the 0.1 significance level, expect for DELAK algorithm.

(2) SVM as the base classifier

When the SVM is adopted as the base classifier, the ROC curve of each algorithm on each dataset is shown in Fig. 11. Among them, the red line in each subgraph is the ROC curve of ELDW.

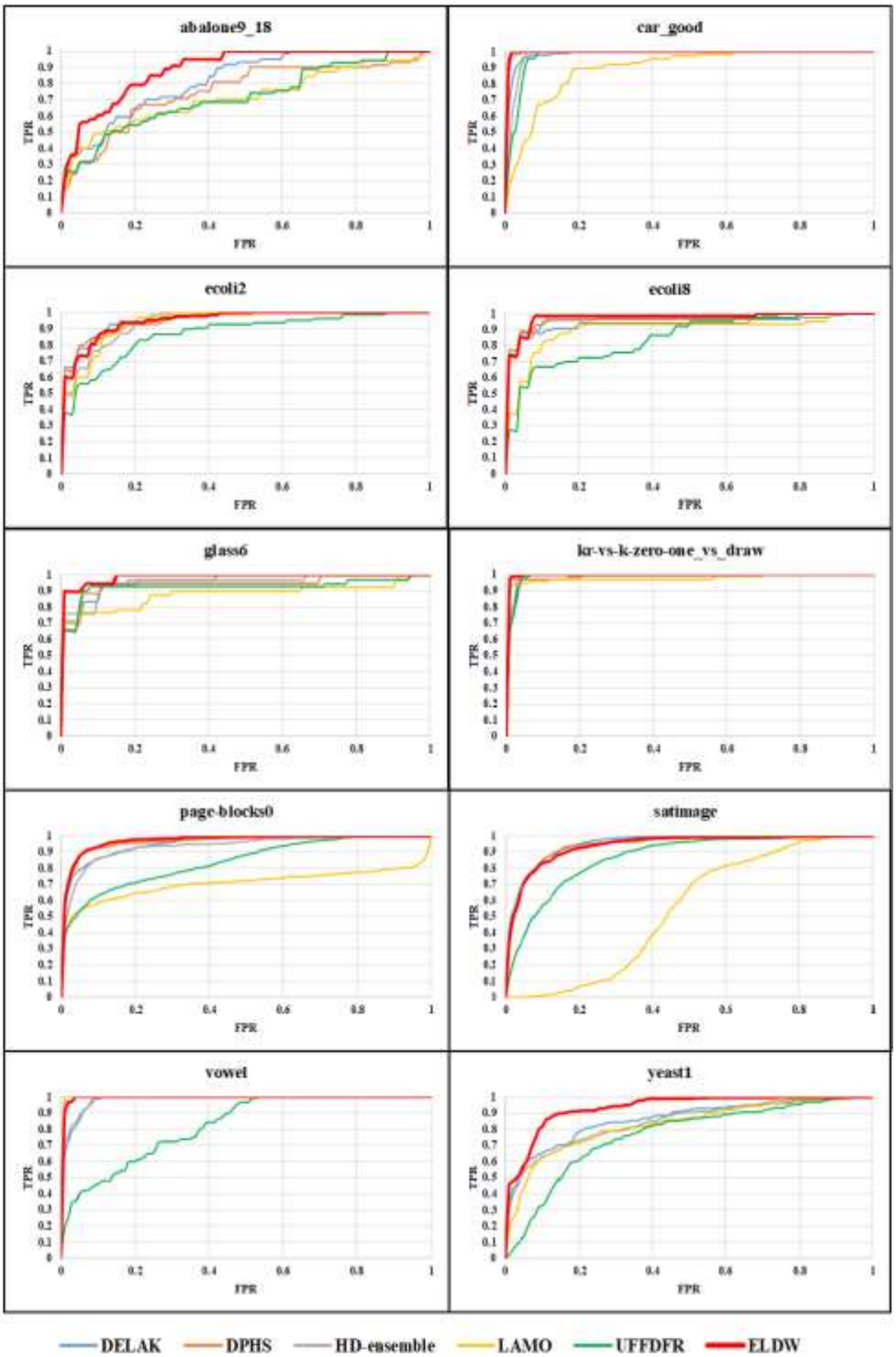

Fig. 11 ROC curves of algorithms on benchmark datasets (SVM as base classifier)

It can be seen from Fig. 11 that ELDW also achieves the highest ROC curve on most datasets. And the performance of other algorithms is unstable. For example, as shown by the blue line in each subgraph, DELAK algorithm performs the best on Ecoli2 and Satimage datasets, but performs poorly on other datasets. The AUC values are shown in Table 8.

Table 8 AUC value of each algorithm on each dataset (SVM as base classifier)

	DELAK	DPHS	Hd_ensemble	LAMO	UFFDFR	ELDW	Rank
Abalone9_18	0.821	0.749	0.722	0.735	0.612	**0.887**	1
Car_good	0.987	0.994	0.980	0.895	0.973	**0.997**	1
Ecoli2	**0.956**	0.954	0.942	0.944	0.870	0.952	3
Ecoli8	0.941	0.962	0.958	0.899	0.847	**0.975**	1
Glass6	0.951	0.976	0.970	0.953	0.917	**0.981**	1
Kr-vs-k-zero-one_vs_draw	0.982	0.990	0.992	0.977	0.988	**0.999**	1
Page-blocks0	0.954	0.967	0.937	0.705	0.841	**0.974**	1
Satimage	**0.953**	0.943	0.923	0.540	0.867	0.941	3
Vowel	0.984	0.999	0.986	**1**	0.822	0.998	3
Yeast1	0.854	0.862	0.845	0.826	0.765	**0.936**	1
mean AUC	0.938	0.940	0.926	0.847	0.850	**0.964**	1
vari AUC	0.055	0.077	0.083	0.146	0.107	**0.035**	1

It can be observed from Table 8 that our ELDW obtains the highest AUC values on 7 benchmark datasets when the SVM is adopted as base classifier. As for the datasets where the AUC of ELDW is not the highest, the ranking of ELDW is 3. These results indicate that the performance of ELDW is also competitive. Similarly, considering the performance over all datasets, ELDW obtains the highest average AUC value and the lowest variance of AUC.

We apply the Wilcoxon's signed-rank test to estimate the statistical differences of these algorithms over all the datasets. The p-values of Wilcoxon's signed-rank test are reported in Table 9.

Table 9 The p-values of Wilcoxon's signed-rank test (SVM as base classifier)

	DELAK	DPHS	Hd_ensemble	LAMO	UFFDFR
ELDW	0.013	0.02	0.001	0.003	0.001

The results in Table 9 illustrate that the differences between ELDW and these five competitors is significant at the 0.05 significance level.

(3) NB as the base classifier

When the NB is adopted as the base classifier, the ROC curve and the AUC values of each algorithm on each dataset are shown in Fig. 12 and Table 10, respectively.

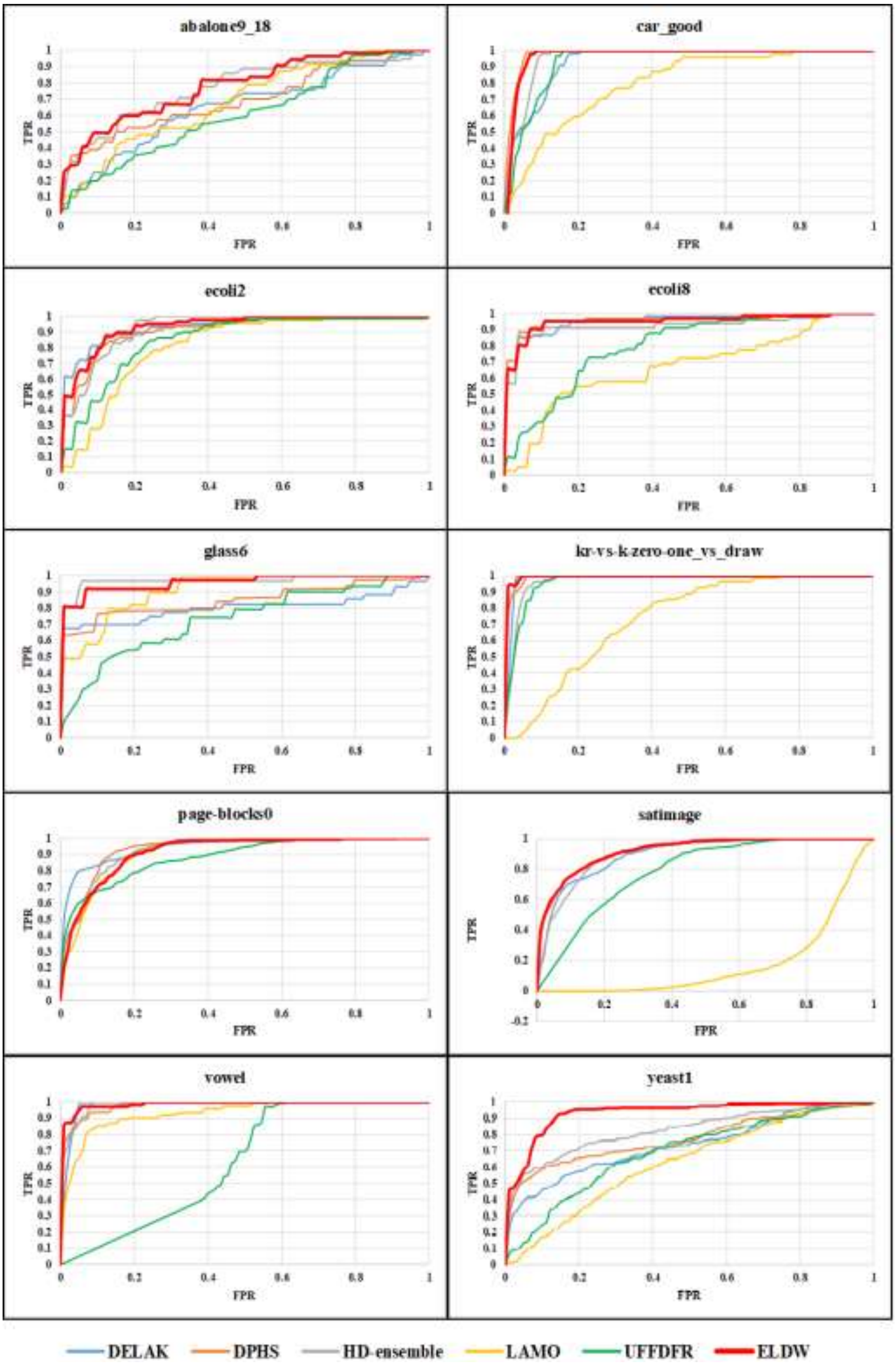

Fig. 12 ROC curves of algorithms on benchmark datasets (NB as base classifier)

The results in Fig. 12 show that ELDW can also achieve the highest ROC curve on most datasets, and the performance of other algorithms on datasets is unstable. For example, as shown by the yellow line in each subgraph, LAMO algorithm performs better on Abolone9_18, Glass6 and Page-blocks0 datasets, but performs poor on other datasets.

Table 10 AUC value of each algorithm on each dataset (NB as base classifier)

	DELAK	DPHS	Hd_ensemble	LAMO	UFFDFR	ELDW	Rank
Abalone9_18	0.650	0.697	0.770	0.691	0.606	**0.882**	1
Car_good	0.935	0.979	0.957	0.803	0.933	**0.981**	1
Ecoli2	0.930	0.913	0.928	0.812	0.852	**0.939**	1
Ecoli8	0.954	**0.961**	0.927	0.670	0.798	0.952	3
Glass6	0.803	0.850	**0.970**	0.912	0.729	0.962	2
Kr-vs-k-zero-one_vs_draw	0.985	**0.991**	0.971	0.739	0.965	0.983	3
Page-blocks0	**0.947**	0.934	0.929	0.915	0.885	0.915	4
Satimage	0.901	0.922	0.903	0.177	0.790	**0.924**	1
Vowel	0.985	0.981	0.986	0.932	0.627	**0.990**	1
Yeast1	0.734	0.775	0.825	0.627	0.691	**0.933**	1
mean AUC	0.882	0.900	0.916	0.728	0.788	**0.946**	1
vari AUC	0.114	0.097	0.068	0.221	0.123	**0.034**	1

Table 10 indicates that our ELDW obtains the highest AUC values on 6 benchmark datasets when the NB is adopted as base classifier. As for the datasets where the AUC of ELDW is not the highest, the ranking of ELDW is 2 or 3. Besides, ELDW also obtains the highest average AUC value and the lowest variance of AUC.

We apply the Wilcoxon's signed-rank test to estimate the statistical differences of these algorithms over all the datasets. The p-values of Wilcoxon's signed-rank test are reported in Table 11.

Table 11 The p-values of Wilcoxon's signed-rank test (NB as base classifier)

	DELAK	DPHS	Hd_ensemble	LAMO	UFFDFR
ELDW	0.06	0.16	0.03	0.007	0.001

The results in Table 11 illustrate that the differences between ELDW and four competitors (DELAK Hd_ensemble, LAMO, UFFDFR) are significant at the 0.1 significance level, expect for DPHS.

(4) MLP as the base classifier

When the MLP is adopted as the base classifier, the ROC curve of each algorithm on each dataset is shown in Fig. 13. Among them, the red line in each subgraph is the ROC curve of ELDW.

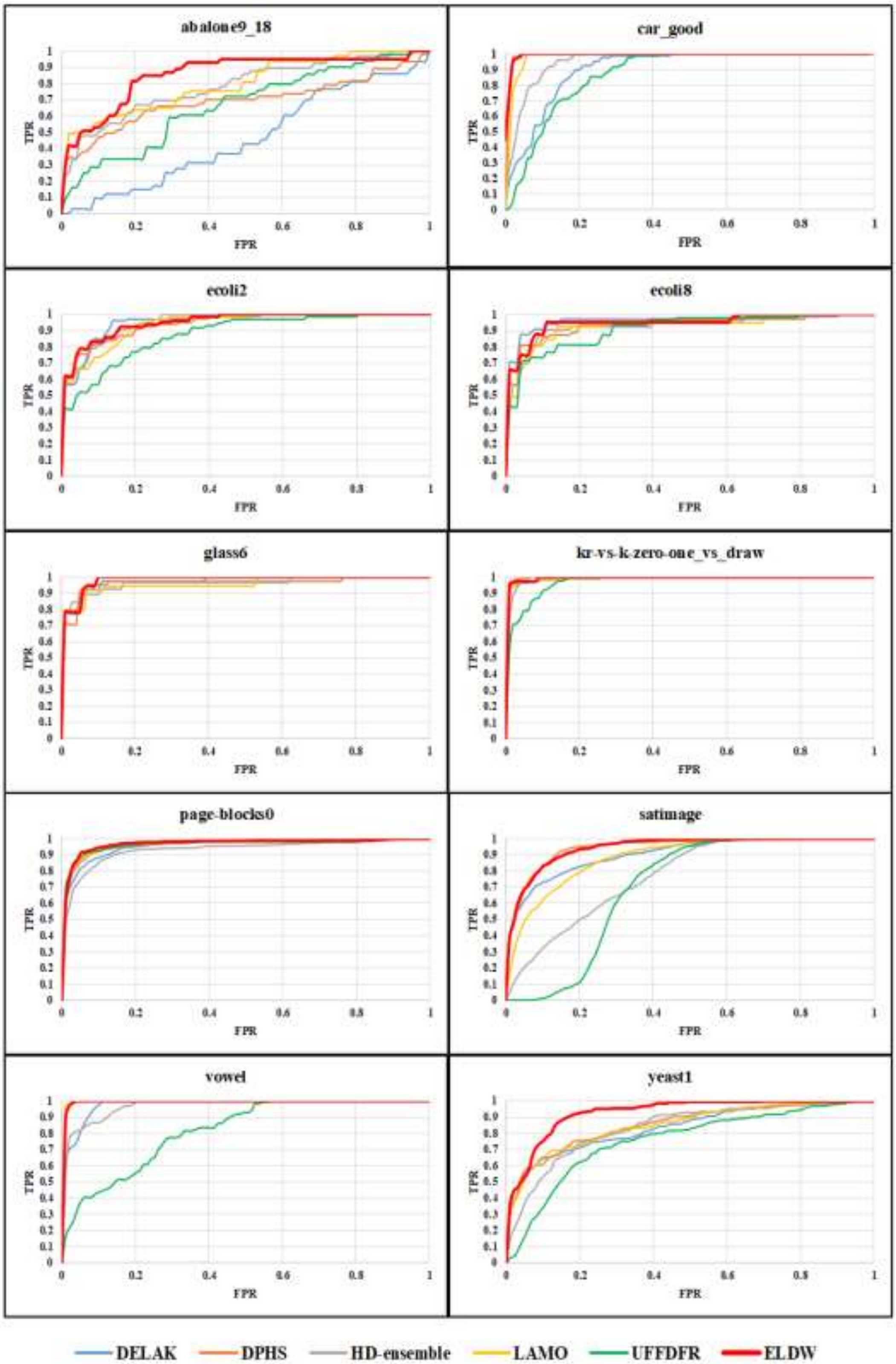

Fig. 13 ROC curves of algorithms on benchmark datasets (MLP as base classifier)

The results in Fig. 13 show that ELDW can also achieve the highest ROC curve on most datasets. And the performance of other algorithms on datasets is unstable. For example, as shown by the orange line in each subgraph, DPHS algorithm performs better on Car_good and Kr-vs-k-zero-one_vs_draw datasets, but performs poor on other datasets. In detail, we show the AUC value of each algorithm on datasets in Table 12 to give a quantitative comparison.

Table 12 AUC value of each algorithm on each dataset (MLP as base classifier)

	DELAK	DPHS	Hd_ensemble	LAMO	UFFDFR	ELDW	Rank
Abalone9_18	0.484	0.694	0.786	0.802	0.690	**0.858**	1
Car_good	0.906	**0.993**	0.954	0.982	0.871	0.987	2
Ecoli2	**0.954**	0.945	0.950	0.939	0.878	0.951	2
Ecoli8	**0.959**	0.937	0.930	0.926	0.911	0.951	2
Glass6	**0.977**	0.961	0.964	0.959	0.950	0.968	2
Kr-vs-k-zero-one_vs_draw	0.994	**0.997**	0.991	0.992	0.971	**0.997**	1
Page-blocks0	0.958	0.970	0.933	**0.972**	0.843	0.970	2
Satimage	0.905	0.944	0.771	0.886	0.701	**0.945**	1
Vowel	0.980	0.997	0.973	**0.999**	0.816	0.998	2
Yeast1	0.839	0.851	0.837	0.851	0.757	**0.931**	1
mean AUC	0.896	0.943	0.909	0.931	0.839	**0.956**	1
vari AUC	0.151	0.067	0.080	0.065	0.097	**0.041**	1

It can be observed from Table 12 that our ELDW obtains 5 highest AUC values on 10 benchmark datasets when the MLP is adopted as base classifier. As for the datasets where AUC of ELDW is not the highest, the performance of ELDW is also competitive. For these datasets, the ranking of ELDW is 2. What is more, considering the performance overall datasets, ELDW also obtains the highest average AUC value and the lowest variance of AUC.

We apply the Wilcoxon's signed-rank test to estimate the statistical differences of these algorithms over all the datasets. The p-values of Wilcoxon's signed-rank test are reported in Table 13.

Table 13 The p-values of Wilcoxon's signed-rank test (MLP as base classifier)

	DELAK	DPHS	Hd_ensemble	LAMO	UFFDFR
ELDW	0.064	0.041	0.001	0.009	0.001

The results in Table 13 illustrate that the differences between ELDW and five competitors is significant at the 0.1 significance level.

In summary, the above experimental results show that the proposed ELDW algorithm can achieve the highest AUC value on most benchmark datasets, indicating that ELDW algorithm can identify the minority class instances correctly with low false positive rate. On the other hand, the AUC variance of ELDW overall datasets is the smallest, indicating that the performance of ELDW is more stable than others. To sum up, ELDW algorithm can achieve accurate and stable classification accuracy on the minority class instances.

4.4 Discussion on results

It can be seen from the results in section 4.3 that ELDW performs well in ROC and AUC metrics on most benchmark datasets, no matter which base classifier is chosen. What is more, ELDW algorithm can achieve the highest AUC value on most benchmark datasets, indicating that it can identify the minority class instances accurately with low false positive rate. And the AUC variance of ELDW overall datasets is the smallest, indicating that the performance of ELDW is more stable than others. The reason why ELDW can perform well is that we combine the dynamic ensemble learning with the instance-based learning to improve the identification of minority class instances.

For dynamic ensemble learning process, in training phase, ELDW generates training subsets by combining all minority class instances with majority class neighbors that are at different distance, guaranteeing the diversity of the base classifiers. When an unknown instance comes, ELDW dynamically integrates the result of base classifier based on the cross entropy of neighbors in each training subset to improve the overall performance and robustness by exploring the local information of each unknown instance.

In instance-based learning, in order to enhance the identification of minority class instances, ELDW explores feature similarity of unknown instance to the minority class neighbors. Finally, for each unknown instance, ELDW outputs the probability of being minority class by combining the result of dynamic ensemble and instance-based learning. It is the feature similarity that further separates the instances from minority class and majority class on the basis of ensemble learning.

Nevertheless, there is some tradeoffs in ELDW algorithm. In terms of time efficiency, in the training phase, ELDW need search majority class neighbors for minority class instances iteratively to form the subsets. Besides, when the unknown instance arrives, ELDW also need to find its neighbors in each subset to integrate the result of base classifier dynamically. Both of the above processes are time-consuming due to finding the nearest neighbors, however, the accuracy of prediction model is improved. Therefore, how to improve the time efficiency of the algorithm while ensuring the accuracy is a direction worthy of future research. Furthermore, the final result is not highly interpretive. Another worth-mentioning advantage of the proposed ELDW is that it has high interpretability, since it is a type of instance-based learning. As Breiman et al. (2003) suggested, model accuracy comes before model interpretability when developing data modeling. We can observe from the experimental results that our ELDW not only achieves high accuracy on the minority class, but also provides highly interpretable results.

5. Application to response modeling in direct marketing

The above experimental results have verified the effectiveness of the proposed ELDW for imbalance classification. In this section, we apply the ELDW to a real-world direct marketing activity of an insurance company.

5.1 Data description

The dataset comes from an insurance company in Netherlands. This company is ready to sell motor vehicle insurance to European families who have brought vehicles. In this dataset, each household's record contains a target variable indicating whether they buy insurance and 93 predictor variables indicating information on both socio-demographic characteristics and ownership of various types of insurance policies. The number of customers in this dataset is 9629, where there are 586 responding customers and 9043 non-responding customers. The imbalanced ratio of this dataset is 1:15, which makes it an extremely imbalanced classification problem. The detailed information of each predictor variable is shown in Table 14.

Table 14 Description of the data set

Attribute ID	Attribute description
1	Number of houses owned by residents
2	Average size of households
3	Average age of residents
4-13	Psychographic segment: successful hedonists, driven growers, average family, career loners, living well, cruising seniors, retired and religious, family with grownups, conservative families, or farmers
14-17	Proportion of residents with Catholic, Protestant, other, and no religion
18-21	Proportion of residents of married, living together, other relation, and singles
22-23	Proportion of households without children and with children
24-26	Proportion of residents with high, medium, and lower education level
27	Proportion of residents in high status
28-32	Proportion of residents who are entrepreneur, farmer, middle management, skilled laborers, and unskilled laborers
33-37	Proportion of residents in social class A, B1, B2, C, and D
38-39	Proportion of residents who rented home and owned home
40-42	Proportion of residents who have 1, 2, and no car
43-44	Proportion of residents with national and private health service
45-50	Proportion of residents whose income level is <$30,000; $30,000–$45,000; $45,000–$75,000; $75,000–$123,000; >$123,000; and average
51	Proportion of residents in purchasing-power class
52-72	Scaled contribution to various types of insurance policies such as private third party, third-party firms, third-party agriculture, car, van, motorcycle/scooter, truck, trailer, tractor, agricultural M/C, moped, life, private accident, family accidents, disability, fire, surfboard, boat, bicycle, property, social security
73-93	Scaled number of households holding insurance policies for the same categories as in scaled contribution attributes

5.2 Performance evaluation metrics

In this application, the purpose is to identify the customers who really buy the insurance. In direct marketing, the managers are interested in the customers who are highly probable to respond. Therefore, we use the Cumulative Hit Rate (CHR) as the performance evaluation metric, in addition to the ROC and AUC. The CHR is defined as the ratio of the number of Real Customers (RC) identified at present out of the Total Real Customers (TRC) as shown in Eq. (13):

$$\mathrm{CHR} = \mathrm{RC}/(\mathrm{TRC}) \tag{13}$$

A typical CHR curve is shown in Fig. 14. Its abscissa is the marketing scale and its ordinate is the CHR. The so-called marketing scale is the ratio of the number of customers receiving marketing activities out of the total number of customers. As shown in Fig. 14, the point A indicates that when the 20% of customers will receive the information of marketing activities, and 95% of real customers can be recognized.

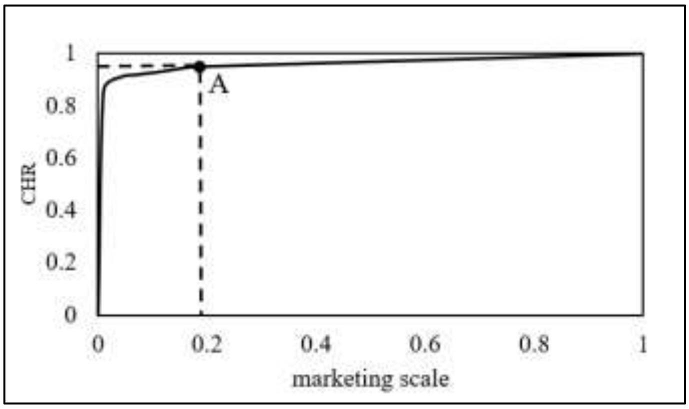

Fig. 14 A typical CHR curve

5.3 Results comparison

In the experiments, we also use LR, SVM, NB and MLP as base classifiers.

(1) Comparison of results with respect to the ROC curves

The ROC curves of ELDW and the competing methods on the dataset are shown in Fig. 15. Among them, the red line in each subgraph is the ROC curve of ELDW.

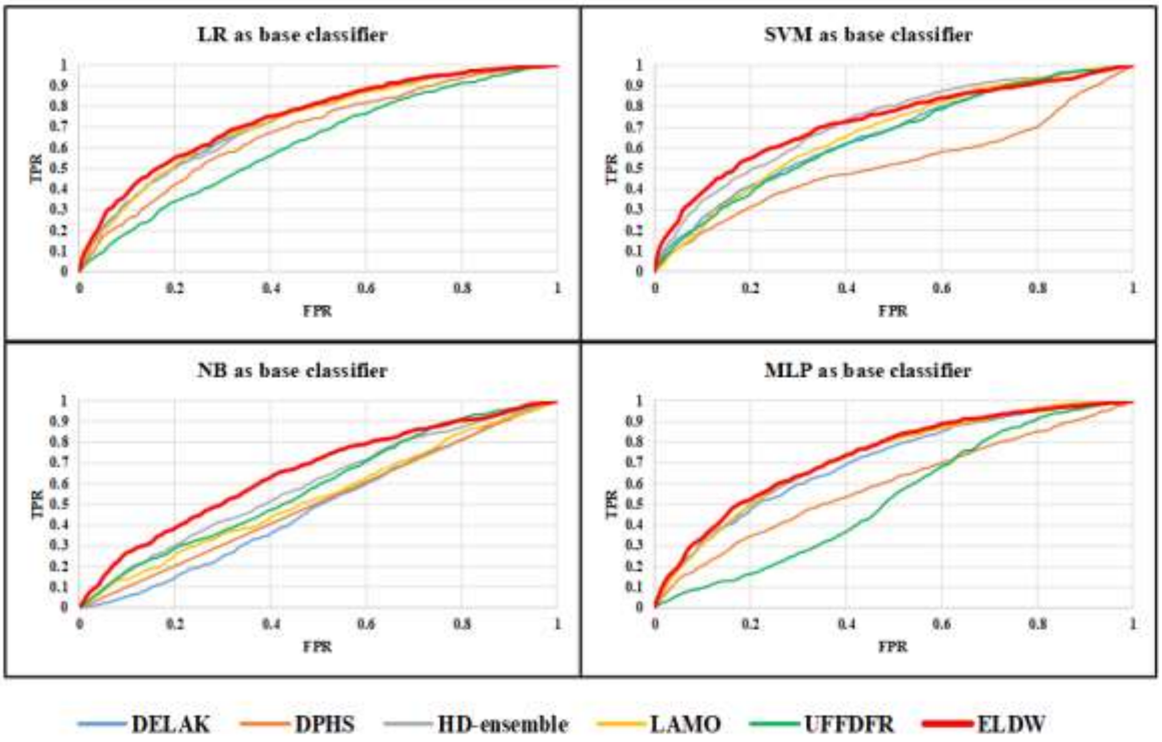

Fig. 15 ROC curves of different methods on the marketing dataset

Fig. 15 shows that the ELDW can achieve the highest ROC curve regardless of which base classifier it adopts. On the other hand, the performance of other algorithms is unstable when they adopt different base classifier. For example, as shown by the green line in each subgraph, UFFDFR algorithm performs relatively better when base classifier is SMV and NB, but it performs the worst when the base classifier is LR or MLP.

(2) Comparison of results with respect to the AUC value

The AUC value of each algorithm is shown in Table 15. In each row, the highest AUC value is bolded and unlined, and the last column shows the ranking of ELDW when the AUC value of each row is ordered.

Table 15 AUC value of each algorithm on real marketing dataset

base classifier	DELAK	DPHS	Hd_ensemble	LAMO	UFFDFR	ELDW	Rank
LR	0.732	0.680	0.728	0.729	0.622	**0.749**	1
SVM	0.663	0.521	0.724	0.674	0.655	**0.726**	1
NB	0.487	0.507	0.590	0.534	0.583	**0.654**	1
MLP	0.707	0.593	0.727	0.726	0.532	**0.737**	1

The AUC values in Table 15 show that ELDW obtains the highest AUC value no matter which base classifier is adopted. It can also be seen that when LR is adopted as base classifier, ELDW achieves the highest AUC value 0.749.

(3) Comparison of results with respect to the ROC curves CHR values

It is worth noting that due to the limited budget, it is impossible to carry out marketing activities to all customers in direct marketing. Thus, we need to select customers who are most likely to respond to marketing activities in direct marketing.

In Fig 16, when ELDW adopts LR, SVM, NB and MLP as base classifier respectively, the CHR of each algorithm is shown as follows. In each subgraph, the abscissa marketing scale indicates the proportion of customers receiving the marketing activity. And the ordinate is CHR.

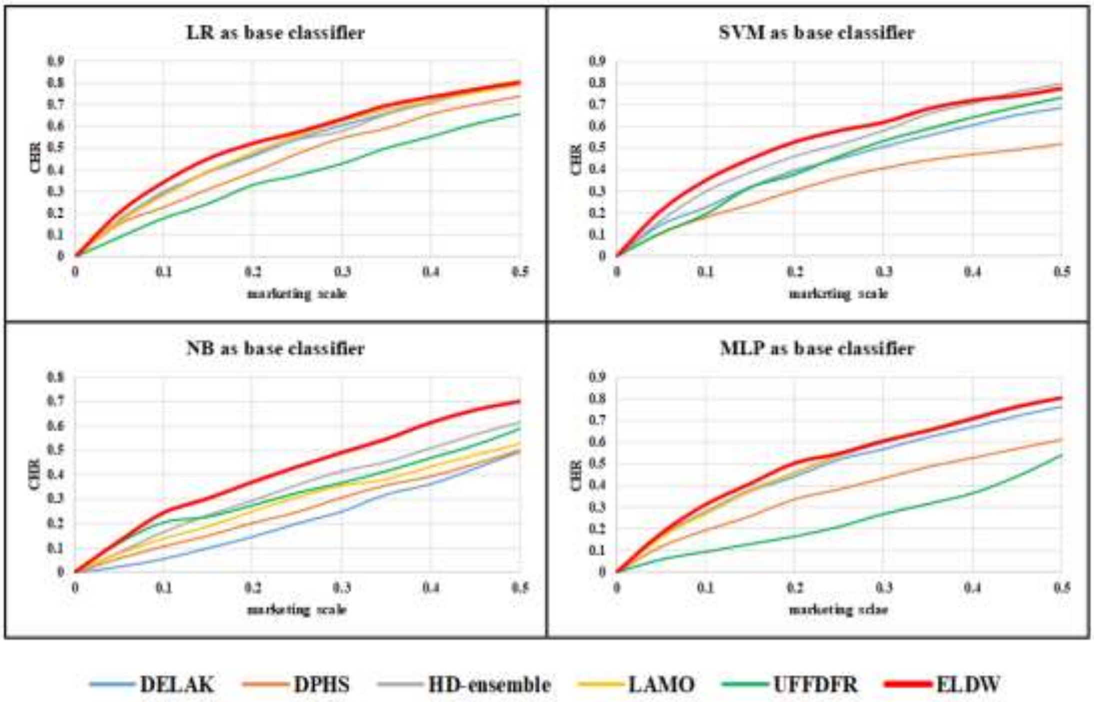

Fig. 16 CHR curves of different methods on the marketing dataset

It is obvious from Fig. 16 that ELDW can achieve highest CHR curve in each subgraph. In detail, no matter the marketing scale is 0.1, 0.2 or 0.5, ELDW can obtain the highest CHR value, which means that ELDW can identify much more real customers than other algorithms under the same marketing scale. In other words, ELDW can identify much more target customers and bring more profits to the company within the same marketing budget.

The CHR of ELDW under different marketing scale is shown in Table 16 when ELDW adopts different base classifier. The highest CHR in each column is bold and underlined.

Table 16 CHR of ELDW under different marketing scale

marketing scale	0.05	0.10	0.15	0.2	0.25	0.3	0.35	0.40	0.45	0.5
LR	0.204	0.344	**<u>0.451</u>**	0.521	0.572	**<u>0.632</u>**	**<u>0.694</u>**	**<u>0.733</u>**	**<u>0.770</u>**	0.802
SVM	**<u>0.211</u>**	**<u>0.351</u>**	0.449	**<u>0.528</u>**	**<u>0.580</u>**	0.619	0.681	0.721	0.741	0.775
NB	0.127	0.245	0.303	0.370	0.433	0.491	0.546	0.614	0.666	0.701
MLP	0.178	0.312	0.408	0.503	0.547	0.605	0.653	0.708	0.763	**<u>0.804</u>**

The results in Table 16 show that once the marketing scale is determined, we can choose the base classifier with the highest CHR. For example, when the marketing scale is 0.2, we can choose SVM as the base classifier for ELDW, since its CHR is the highest than other base classifiers. In this way, we can determine which base classifier to be chosen according to the size of marketing scale, so as to maximize the profit under the same budget.

5.4 Management Insights

Establishing customer response model in direct marketing can help companies better predict and manage customer response behavior. The managerial insights of this paper are shown as follows:

(1) The establishment of customer response model

In this paper, establishing customer response model is regarded as an imbalanced classification problem, since the number of responding customers is much smaller than that of non-responding customers. We combine ensemble learning with instance-based learning to recognize the target customers as many as possible in a limited budget. This model identifies the customer response probability to the direct marketing activities by mining their characteristics.

(2) The evaluation of customer response model

In this paper, we use ensemble learning method to build customer response model. And thus, we try different machine learning algorithms as base classifiers to compare their performance. In this process, the models are evaluated by appropriate metrics, such as AUC, cumulative hit rate, etc. to measure the ability of identifying the target customers. In addition, customer response model needs to be updated regularly to adapt to changes in customer behavior. In future work, we should establish a continuous data collection and model updating mechanism to ensure the predictive ability of the model.

(3) The application of customer response model

The customer response model is established for decision-making in direct marketing activities. Under the limited budget, the company conducts direct marketing activities to customers with a higher responding probability to maximize the profits as much as possible, while customers' decisions are influenced by many factors in actual application. That is the reason why customers with similar characteristics will respond differently. And thus, the company should combine the results of the model with the actual marketing strategy to hit target customers.

6. Conclusion

This paper proposes an imbalance classification algorithm based on ensemble learning with dynamic weighting (ELDW) to predict the response of customer to the marketing activities. The performance of ELDW is verified on benchmark datasets, indicating that ELDW can achieve higher accuracy on minority class instances. And then, ELDW is applied to the real marketing activity. The results show that ELDW can identify much more target customers within a limited budget, compared with other state-of-the-art imbalanced algorithms.

This paper comes up with a new dynamic ensemble learning algorithm. First, the majority class instances are divided into several subsets according to whether they are in the neighborhood of minority class instances or not, and then the majority instances in each subset combined with all the minority class instances form a new subset to train the base classifier. In this way, the diverse training subsets are formed. For each unknown instance, when the result of base classifier is integrated, we consider following two aspects. One is the cross entropy of neighbors in each subset, the other is the feature similarity to the minority class instance. After that, each unknown instance obtains the probability of being the minority class, that is, the probability of customers' response to marketing activities.

The main theoretical contribution of this paper is to propose a new dynamic ensemble learning method for solving imbalance classification problems, especially for datasets with high imbalance ratio. For each subset that generates base classifier, the minority class instances included in it are same with each other, while the majority class instances are different. In this way, the base classifier can enhance the recognition of minority class instances. Furthermore, the degree of separation between minority instances and majority instances in different subsets are different, allowing the diversity among the base classifiers to ensuring the performance of ensemble learning. For each unknown instance, the final prediction is obtained by combining the cross-entropy-based integration result with the similarity to the minority instances. In this way, ELDW can predict minority instances accurately, that is, improve the identification of minority customers who respond to the marketing activities.

Despite the contributions of this algorithm, there is still some work need to be improved in the future. For example, the subsets are generated by obtaining the neighbors using Euclidean distance, resulting in the shape of the subset tending to be spherical. It may not be suitable to datasets distributed in other shapes.

Acknowledgements

This research is supported by the China Postdoctoral Science Foundation (2023M740440), the Chongqing Natural Science Foundation Postdoctoral Science Fund Project (CSTB2023NSCQ-BHX0107) and the Chongqing Postdoctoral Program Special Foundation (Grant No. 2022CQBSHTB3065) .

References

Baesens, B., Viaene, S., Van den Poel, D., Vanthienen, J., & Dedene, G. (2002). Bayesian neural network learning for repeat purchase modelling in direct marketing. *European Journal of Operational Research*, 138 (1), 191-211.

Baumgartner, B. and H. Hruschka, Allocation of catalogs to collective customers based on semiparametric response models. *European Journal of Operational Research*, 2005. 162(3): p. 839-849.

Bose, I., & Chen, X. (2009). Quantitative models for direct marketing: A review from systems perspective. *European Journal of Operational Research*, 195(1): 1-16.

Bradlow, E. T., Gangwar, M., Kopalle, P. & Voleti, S. (2017). The role of big data and predictive analytics in retailing. *Journal of Retailing*, 93 (1): 79-95.

Breiman, L. (2003). Statistical modeling: The two cultures. *Quality Control and Applied Statistics*, 48(1): 81-82.

Chan, K. Y., Kwong C. K., & Kremer, G. E. (2020). Predicting customer satisfaction based on online reviews and hybrid ensemble genetic programming algorithms. *Engineering Applications of Artificial Intelligence*, 95: 103902.

Chaudhuri, N., Chaudhuri， N., Gupta， G., Vamsi， V., & Bose, I. (2021) On the platform but will they buy? Predicting customers' purchase behavior using deep learning. *Decision Support Systems*, 149: 113622.

Chawla, N., Bowyer, K., Hall, L., & Kegelmeyer, W. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *The Journal of Artificial Intelligence Research*, 16: 321-357.

Chen, Z., Duan, J., Kang, L., & Qiu, G. (2021). A hybrid data-level ensemble to enable learning from highly imbalanced dataset. *Information Sciences*, 554: 157-176.

Chun, Y. H. (2012). Monte Carlo analysis of estimation methods for the prediction of customer response patterns in direct marketing. *European Journal of Operational Research*, 217(3): 673-678.

De Caigny, A., Coussement, K. & De Bock, K.W. (2018). A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees. *European Journal of Operational Research*, 269(2): 760-772.

Dumitrescu, E., Hue, S., Hurlin, C., & Tokpavi, S. (2022). Machine learning for credit scoring: Improving logistic regression with non-linear decision-tree effects. *European Journal of Operational Research*, 297(3): 1178-1192.

Fu, Y. G., Ye, J. F, & Y, Z. F. (2021). Construction of EBRB classifier for imbalanced data based on Fuzzy C-Means clustering. *Knowledge-Based Systems*, 234: 107590.

Gao, X., Ren, B., Zhang, H., Sun, B., Li, J., Xu, J., He, Y., & Li, K. (2020). An ensemble imbalanced classification method based on model dynamic selection driven by data partition hybrid sampling. *Expert Systems with Applications*, 160: 113660.

Guo, C., Liu, M., & Lu, M. (2021). A Dynamic Ensemble Learning Algorithm based on K-means for ICU mortality prediction. *Applied Soft Computing*, 103: 107166.

Hong, T. H., & Kim, E. M. (2010). Predicting the Response of Segmented Customers for the Promotion Using Data Mining. *Information Systems Review*, 12(2): 75-88.

Hppner, S., Baesens, B., & Verbeke, W. (2021). Instance-dependent cost-sensitive learning for detecting transfer fraud. *European Journal of Operational Research*, 297(1): 291-300.

Kang, P., & Cho, S. (2008). Locally linear reconstruction for instance-based learning. *Pattern Recognition*, 41: 3507-3518.

Kang, P., Cho, S., & Maclachlan, D. L. (2012). Improved response modeling based on clustering, under-sampling, and ensemble. *Expert Systems with Applications*, 39 (8): 6738-6753.

Kim, J., Ji, H. G., Oh, S., Hwang, S., & Pobil, A. P. D. (2021). A deep hybrid learning model for customer repurchase behavior. *Journal of Retailing and Consumer Services*, 59: 102381.

Knott, A., Hayes, A., & Neslin, S. A. (2002). Next-product-to-buy models for cross selling applications. *Journal of Interactive Marketing*, 16 (3): 59-75.

Kucukasci, E. S., Baydogan, M., Taskin, Z. C. (2021). A linear programming approach to multiple instance learning. *Turkish Journal of Electrical Engineering and Computer Sciences*. 29(4):2186-2201.

Ładyżyński, P., Żbikowski, K., & Gawrysiak, P. (2019). Direct marketing campaigns in retail banking with the use of deep learning and random forests. *Expert systems with applications*, 134: 28-35.

Liu, B., Blekas, K., & Tsoumakas, G. (2022). Multi-label sampling based on local label imbalance. *Pattern Recognition*, 122: 108294.

Liu, Y., Cao, J., & Zhang, Q. (2022). The product marketing model of the economic zone by the sensor big data mining algorithm. *Sustainable Computing: Informatics and Systems*, 36:

100820.

Miguéis, V. L., Camanho, A. S., & Borges, J. (2017). Predicting direct marketing response in banking: comparison of class imbalance methods. *Service Business*, 2017. 11(4): 831-849.

Nestor, B., Sergio, M., & Marcelo, F. (2019). Mutual information and sensitivity analysis for feature selection in customer targeting: A comparative study. *Journal of Information Science*, 45(1): 53-67.

Ng, W. W. Y., Liu, Z., & Zhang, J. (2021). Maximizing minority accuracy for imbalanced pattern classification problems using cost-sensitive Localized Generalization Error Model. *Applied Soft Computing*, 104: 107178.

Rao, C., Liu, M., Goh, M, & Wen, J. (2020). 2-stage modified random forest model for credit risk assessment of P2P network lending to "Three Rurals" borrowers. *Applied Soft Computing*, 95: 106570.

Roh, S. B., Kim, Y. S., Ahn, T. C. (2020). Lazy Learning for Norparametric Locally Weighted Regression. *International Journal of Fuzzy Logic and Intelligent Systems*, 2(20): 145-155.

Roy, A., Cruz, R. M.O., Sabourin, R., & Cavalcanti, G. D. C. (2018). A study on combining dynamic selection and data preprocessing for imbalance learning. *Neurocomputing*, 286: 179-192.

Shah, D. & Murthi, B.P.S. (2021). Marketing in a data-driven digital world: Implications for the role and scope of marketing. *Journal of Business Research*, 125: 772-779.

Stripling, E., vanden Broucke, S., Antonio K., Baesens B., & Snoeck M. (2018). Profit maximizing logistic model for customer churn prediction using genetic algorithms. *Swarm and Evolutionary Computation*, 40: 116-130.

Su, Y., Cheng, D., Zong, M., Li, L, Zhu, Y. (2015). K-nearest neighbor imputation based on sparse coding. *Application Research of Computers*, 32(07):1942-1945.

Trusov, M., Ma, L., & Jamal, Z. (2016). Crumbs of the cookie: user profiling in customer-base analysis and behavioral targeting. *Marketing Science*, 35 (3): 405-426.

Van den Poel, D. & Buckinx, W. (2005). Predicting online-purchasing behaviour. *European Journal of Operational Research*, 166(2): 557-575.

Vuttipittayamongkol, P., & Elyan, E. (2020). Neighbourhood-based undersampling approach for handling imbalanced and overlapped data. *Information Sciences*, 509: 47-70.

Wang, C., Deng, C., Yu, Z., Hui, D., Gong, X., & Luo, R. (2021). Adaptive ensemble of classifiers with regularization for imbalanced data classification. *Information Fusion*, 69: 81-102.

Wang, X., Xu, J., Zeng, T., & Jing, L. (2021). Local distribution-based adaptive minority oversampling for imbalanced data classification. *Neurocomputing*, 422: 200-213.

Xiao, Y. S., Liang, F., Liu, B. (2022). A Transfer Learning-Based Multi-Instance Learning Method With Weak Labels. *IEEE Transactions on Cybernetics*, 52(1): 2168-2267.

Yang, Z.G., Li H. Q., Zhu L.P., Liu, Q., Ali, S. (2017). A case based method to predict optimal k value for k-NN algorithm. *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, 1(33): 55-65.

Yu, J., Yang, J., Wu, Y., Tang, D., Dai, J. (2020), Online state-of-health prediction of lithium-ion batteries with limited labeled data. *International Journal of Energy Research*, 14 (44): 11345-11353.

Zhang, S. (2020), Cost-sensitive KNN Classification. *Neurocomputing*, 391: 234-242.

Zhang, H., Jiang, L., & Li, C. (2021). CS-ResNet: Cost-sensitive residual convolutional neural network for PCB cosmetic defect detection. *Expert Systems with Applications*, 185: 115673.

Zheng, M., Li， T., Zheng， X., Yu， Q., Chen， C., Zhou， D., Lv， C., & Yang, W. (2021). UFFDFR: Undersampling framework with denoising, fuzzy c-means clustering, and representative sample selection for imbalanced data classification. *Information Sciences*, 576: 658-680.

Zhu, R., Wang, Z., Ma, Z., Wang, G., Xue, J. (2018). LRID: A new metric of multi-class imbalance degree based on likelihood-ratio test. *Pattern Recognition Letters*, 116: 36-42.