Information Fusion 36 (2017) 80–89

Contents lists available at ScienceDirect

Information Fusion

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

One versus one multi-class classification fusion using optimizing decision directed acyclic graph for predicting listing status of companies

Ligang Zhou[a,*], Qingyang Wang[b], Hamido Fujita[c]

[a] School of Business, Macau University of Science and Technology, Taipa, Macau
[b] Asia-Pacific Academy of Economics and Management, University of Macau, Macau
[c] Faculty of Software and Information Science, Iwate Prefectural University, Iwate, Japan

ARTICLE INFO

Article history:
Received 31 December 2015
Revised 10 November 2016
Accepted 10 November 2016
Available online 11 November 2016

Keywords:
Multi-class classification
One versus one
Listing status
Prediction
Decision directed acyclic graph
Optimizing

ABSTRACT

Most existing research has demonstrated the success of different decomposition and ensemble strategies for solving multi-class classification problems. This study proposes a new ensemble strategy for One-vs-One (OVO) scheme that uses optimizing decision directed acyclic graph (ODDAG) whose structure is determined by maximizing the fitness on the training set instead of by predefined rules. It makes an attempt to reduce the effect of non-competent classifiers in OVO scheme like decision directed acyclic graph (DDAG) but in another way. We test the proposed method on some public data sets and compare it to some other widely used methods to select the proper candidates and related settings for a problem with practical concern from financial industry in China, i.e. the prediction of listing status of companies. The experimental result shows that our model can outperform the benchmarked methods on this real problem. In addition, the ODDAG combined with decision tree is a white box model whose internal rules can be viewed and checked by decision makers.

1. Introduction

Due to the wide existence of multi-class classification problems in different areas, many different methods have been developed to solve such problems. The strategies behind these methods can be roughly categorized into two categories [1]. One category of extensible algorithm, is to extend some binary classification techniques by special formulations to make them applicable for multi-class classification problems, such as discriminant analysis [2–4], decision trees [5,6], *k*-nearest neighbors [7,8], Naive Bayes [9,10], neural networks [11–13], and support vector machines [14–17]. Another category of decomposition and ensemble methods (DEM), is to decompose a multi-class classification problem into a set of binary classification problems that can be solved by binary classifiers (BCs), and then classify a new observation by applying an aggregative strategy on the binary classifiers' predictions.

A wide variety of empirical studies have reported the decomposition and ensemble methods can increase the performance on multi-class classification problems. Garcia et al. [18] demonstrated the proposed DEM increases the performance of the noise filters studied. Sesmero [19] formalized and evaluated an ensemble of classifiers by using a specific attribute subset to train the base learners and demonstrated their model is as accurate as some well-known classification methods in most cases. Galar et al. [20–22] conducted a comprehensive investigation into different ensemble strategies to combine the outputs of the binary classifiers generated from different decomposition strategies in solving multi-class classification problems. Their empirical study showed that the performance of DEM is dependent on the selection of decomposition strategy, ensemble strategy, the binary classifier, and the characteristics of the problem.

Most existing research shows that the design or selection of decomposition and ensemble strategies play an important role in the performance of DEMs. With regard to decomposition strategies, OVO [23,24], One-vs-All (OVA) [20,25], and error-correcting output coding (ECOC) [26] are the most widely used. Lorena [27] provided a comprehensive review on different decomposition strategies for multi-class problems. Galar et al. [20] demonstrated that OVO and OVA strategies are simple but powerful.

However, one inherent flaw of OVO decomposition strategy is the problem of non-competent classifiers [28]. A binary classifier can only distinguish the observations from the two classes used

[*] Corresponding author.
E-mail addresses: mrlgzhou@gmail.com (L. Zhou), qywang@umac.mo (Q. Wang), issam@iwate-pu.ac.jp (H. Fujita).

http://dx.doi.org/10.1016/j.inffus.2016.11.009
1566-2535/© 2016 Elsevier B.V. All rights reserved.

in training set, and therefore the binary classifier has no capability to discriminate the observations from other classes. For a new observation whose class is unknown, some binary classifiers from OVO decomposition are competent and some are non-competent. However, which binary classifiers are competent and which binary classifiers are non-competent for the new observation are unknown as well. If all outputs from both the competent and non-competent classifiers are taken into account equivalently in the ensemble stage, the non-competent classifiers may mislead the correct classification of the new observation.

Many efforts have been made in developing ensemble strategy to reduce the effect of non-competent classifiers. Weighted voting (WV), an straightforward and widely used strategy, is to classify a new observation into the class with the maximum total confidence which is the sum of confidence from all binary classifiers on the class. DDAG [29] has been proved to be another effective strategy [14]. The basic idea of DDAG is to start at the top node of the hierarchy and successively discard certain classes by nodes that have been accessed until reaching the bottom of the structure, where the final leaf node returns the class of the estimated observation. Each node in the structure corresponds to a binary classifier. Dynamic classifier selection (DCS) [28], an effective and emerging strategy, can reduce the number of non-competent classifiers in the ensemble process. The main idea of DCS is to consider a reduced subset of binary classifiers by analyzing the classes of the observation's neighbours.

The ensemble strategies of WV, DDAG and DCS are predetermined and heuristic. They need not check their performance on training set or validation set and therefore have no feedback mechanism for adjusting the decision structure. Takahashi and Abe [30] demonstrated that the generalization ability of the decision acyclic graph support vector machines (DAGSVM) depends on the decision acyclic graph structure. They proposed to optimize the structure with the estimate of the generalization error defined as the ratio of the number of support vectors to the number of observations in training data set for the pairwise classes so that the class pairs with higher generalization abilities are put in the upper nodes of the tree. The advantage of the DAGSVM is that the optimized tree can be obtained by one-pass scan and therefore it is time-saving. However, it is difficult to extend the optimization of generalization error due to it dependency on the number of support vectors in support vector machines. Therefore, this would lead to hinder the extension of such optimization strategy to other DDAG with different binary classifiers. This study is to develop an ensemble strategy for OVO by optimizing the DDAG structure and apply it for predicting listing status of companies which is an import problem in Chinese stock markets.

In general, listed companies in China exhibit four different listing statuses: (1) normal status without any risk warning, (2) abnormal status with delisting risk warning, (3) abnormal status with other risk warning, and (4) delisted status. These four statuses are denoted as “A” “D”, “B” and “X”, respectively. A company can switch from one listing status to another with the exception that delisted status cannot switch back to any other statuses.

Since different listing statuses indicates different levels of overall risk, correct prediction of the listing status of a listed company is helpful for the company's investors, creditors and other stakeholders to assess the company's risk. Although predicting the listing status of Chinese listed companies is a topic with practical significance, existing studies tend to consider it as a financial distress prediction problem which simplifies the listing status prediction problem as a binary classification problem [31–33].

Zhou et al. [34] introduced OVO and OVA based methods to predict the listing status of Chinese listed company. Although their experimental results demonstrate the efficiency of the proposed methods, but they did not explore the improvement on the aggregative strategies. This paper is to present a new learning architecture based on DDAG for predicting the listing status of Chinese listed companies and compare it to DDAG, DCS and other ensemble methods.

The rest of this paper is organized as follows. Section 2 presents the related decomposition and ensemble strategies. The new learning architecture based on DDAG is given in Section 3. The empirical studies of the proposed learning architecture on some public data sets and the big-scale data set of listing status prediction are presented in Section 4. Section 5 draws conclusions and discusses future research directions.

2. Related decomposition and ensemble strategies

Multi-class classification models aim at assigning a class label for each input observation. Given a training data set $\{(\boldsymbol{X}_1, y_1), \ldots, (\boldsymbol{X}_N, y_N)\}$, where $\boldsymbol{X}_i \in \mathbb{R}^m$ denotes the ith observation feature vector, and $y_i \in \{1, \ldots, K\}$ is the class label of the ith observation. A multi-class classification model is a map function F: $\boldsymbol{X} \to \{1, \ldots, K\}$ inferred from the labeled training data set through a training process.

2.1. Decomposition strategies

2.1.1. One-vs-one decomposition strategy

OVO [20] approach is to divide the multi-class problem with K classes into $C_K^2 = K \times (K-1)/2$ binary classification problems. One binary classifier is constructed for each binary classification problem for discriminating each pair of classes. Let the binary classifier that discriminates the pairwise classes of i and j be denoted by B_{ij}, $i < j$, the output of binary classifier B_{ij}, denoted by p_{ij}, is the posterior probability defined as follows.

$$p_{ij} = f_{ij}(y = i|\boldsymbol{X}), \quad i < j, \quad p_{ij} \in [0, 1], \tag{1}$$

where f_{ij}: $\boldsymbol{X} \to \{i, j\}$ is the map function of binary classifier B_{ij}.

The p_{ij} can be taken as the confidence of binary classifier B_{ij} classifying an observation with feature vector $\boldsymbol{X}$ as class i. Since classifier B_{ij} is only used to discriminate two classes, if the classifier classes an observation into class i with probability p_{ij}, it classes the observation into another class j with probability $1 - p_{ij}$. The outputs of all $K \times (K-1)/2$ binary classifiers can be represented by following score matrix P:

$$P = \begin{pmatrix} - & p_{12} & \cdots & p_{1K} \\ 1 - p_{12} & - & \cdots & p_{2K} \\ \vdots & \vdots & \vdots & p_{iK} \\ 1 - p_{1K} & 1 - p_{2K} & \cdots & - \end{pmatrix} \tag{2}$$

2.1.2. One-vs-all decomposition strategy

OVA approach is to divide the multi-class problem with K classes into K binary classification problems. Each binary classifier is constructed to discriminate one class from all other classes. The binary classifier B_i is trained by all training samples in which the observations of non class i are relabeled as 0.

The output of binary classifier B_i, denoted by p_i, $p_i \in [0, 1]$, is defined similar to Eq. (1). The value of p_i measures the confidence of the classifier B_i classifying an observation as class i. The outputs of all K binary classifiers can be represented by the following vector P:

$$P = (p_1, p_2, \ldots, p_K) \tag{3}$$

2.2. Ensemble strategies

2.2.1. Maximum confidence (MC) strategy

The class selected by the MC for OVA is the class voted by a classifier with the maximum confidence and is defined as follows:

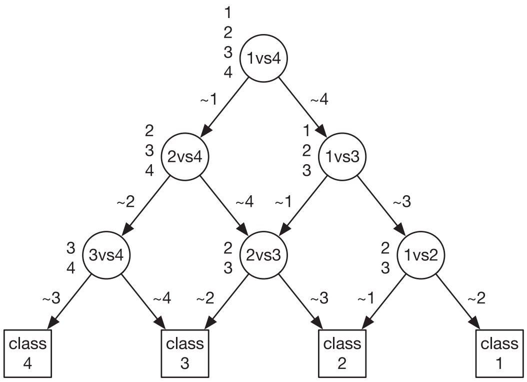


Fig. 1. The DDAG-based pairwise classification for four classes.

$$class = \arg\max_{i=1,\dots,K} p_i \tag{4}$$

2.2.2. Weighted voting (WV) strategy

The class selected by the WV for OVO is the class with the largest total confidence from all binary classifiers and is defined as [35]:

$$class = \arg\max_{i=1,\dots,K} \sum_{1\le j\ne i\le K} p_{ij}. \tag{5}$$

2.2.3. Dynamic classier selection strategy

The procedure of dynamic classifier selection strategy is as follows [28]:

1. Set $k = 3K$, where K is the number of classes;
2. Analyze the class of k nearest neighbours of the estimated observation;
3. If all k nearest neighbours belong to a unique class and $k \le 6K$, set $k = k + 1$ and goto Step 2;
4. The classes in score matrix defined in Eq. (2) that do not appear in the k nearest neighbours will be removed; Any aggregation strategy based on score matrix can be applied on the reduced score matrix.

2.2.4. Decision directed acyclic graph

For the listing prediction problem, there are four classes, i.e. $K = 4$. The DDAG contains $C_4^2 = 4 \times 3/2 = 6$ binary classifiers, one for each pair of classes. The outcome y of an observation $\boldsymbol{x}$ classified by a binary classifier is determined by the value of $f_{ij}(\boldsymbol{x})$ and a threshold value c as follows:

$$y = \begin{cases} i, & \text{if } f_{ij}(\boldsymbol{x}) \ge c \\ j, & \text{otherwise} \end{cases} \tag{6}$$

Fig. 1 shows the decision directed acyclic graph for the four classes. $\sim i$ denotes that $\boldsymbol{x}$ does not belong to class i. According to Fig. 1, to evaluate an observation with input $\boldsymbol{x} \in \mathbb{R}^m$, starting at the root node, the map function$f_{14}(\boldsymbol{x})$ of the binary classifier is evaluated. If $f_{14}(\boldsymbol{x}) \ge c$, $\boldsymbol{x}$ does not belong to class 4, the node is then exited via the right edge to node "1 vs 3". If $f_{14}(x) < c$, $\boldsymbol{x}$ does not belong to class 1, the node is then exited via the left edge to node "2 vs 4". After moving to the next node, the next node's map function is then evaluated. The above processes repeat until a final leaf node reached. The path taken through this processes from the root node to the final leaf node is known as the decision path.

The classification by DDAG is equivalent to operating on a list [29]. The equivalent list statuses for each node is shown near to that node as shown in Fig. 1. The list is initialized with a list of all classes, such as {1, 2, 3, 4} list for the root node "1 vs 4" which corresponds to the first and last element of the list. Then, the input $\boldsymbol{x}$ is evaluated by the binary classifier for class 1 and class 4 denoted by the node "1 vs 4". If the node prefers one of the two classes, the other class is eliminated from the list, and the DDAG proceeds to evaluate the input $\boldsymbol{x}$ by the node with the first and the last elements of the new list. The above procedure for evaluating $\boldsymbol{x}$ in terms of a list is repeated until only one element is left. Then the observation $\boldsymbol{x}$ will be classified into the class that corresponds to the element number. For Fig. 1, suppose the node "1 vs 4" classify $\boldsymbol{x}$ as 1, the element 4 is then removed from the list, the new list is {1, 2, 3}. Then if the node "1 vs 3" classify $\boldsymbol{x}$ as 1, the element 3 is removed from the list. Since only element 1 is left in the list, observation with input $\boldsymbol{x}$ is classified into class 1.

Therefore, the algorithm to determine the DDAG structure for K classes problem and classify observation $\boldsymbol{x}$ is as follows [29,30]:

Algorithm DDAG

Input: $K(K-1)/2$ binary classifiers inferred from training set, $\boldsymbol{x}$.

Output: the structure of DDAG and the predicted class of $\boldsymbol{x}$.

1. Generate the initial list: {1, 2,..., K}.
2. Select the first element and last element in the list to form the class pair (i, j). If $\boldsymbol{x}$ is classified into Class i by the binary classifier for pair (i, j), remove j from the list. Otherwise, remove i.
3. If the number of elements in the list selected at Step 2 is more than one, go to Step 2. Otherwise, classify $\boldsymbol{x}$ into the class associated with the only element in the list and stop.

Each decision node with pairwise classes in DDAG can be any binary classification model trained by training data set. For a problem with K classes, K-1 decision nodes will be evaluated in order to derive an answer.

3. Optimizing decision directed graph

3.1. Generalization error based directed acyclic graph

Takahashi and Abe [30] optimized DDAG by putting the class pairs that are easily separated in the upper nodes. Their estimate of generalization error is based on the number of support vectors. We estimate the generalization error with classification error on the pairwise training set from the binary classifier. The algorithm to determine the DDAG structure and classify $\boldsymbol{x}$ is given as follows:

Algorithm GDDAG:

Input: $K(K$-1)/2 binary classifiers inferred from training set, $\boldsymbol{x}$.

Output: the structure of DDAG and the predicted class of $\boldsymbol{x}$

1. Generate the initial list: $\{1, 2, \dots, \}$.
2. Select the class pair (i, j) with the minimum generalization error from the list. If $\boldsymbol{x}$ is classified into Class i by the binary classifier for pair (i, j), remove j from the list. Otherwise, remove i.
3. If the number of elements in the list selected at Step 2 is more than one, go to Step 2. Otherwise, classify $\boldsymbol{x}$ into the class associated with the only element in the list and stop.

The difference between GDDAG and DDAG is the strategy to select the class pairs from the list. The definition of generalization error and the input of binary classifiers will affect the performance of GDDAG. For example, suppose classification errors (CE) of six binary classifiers shown in Fig. 1 on their corresponding training sets are as Table 1, then 1vs2 with the minimum CE will be selected in the root node. If the observation is not be classified as class 1 by the 1vs2 binary classifier, then the list will be 2, 3, 4, 2vs3 will be selected because it has the minimum CE among the three classifiers 2vs3, 2vs4 and 3vs4. According to GDDAG, the whole decision directed graph as Fig. 2 will be obtained.

Table 1
The classification errors of different binary classifiers.

BCs	1vs2	1vs3	1vs4	2vs3	2vs4	3vs4
CEs	0.15	0.20	0.18	0.25	0.30	0.32

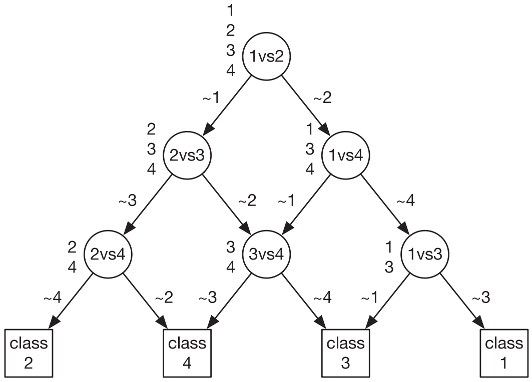


Fig. 2. Example of decision graph by GDDAG.

Table 2
A typical resulting confusion matrix.

Predicted class	Actual class				
	C_1	C_2	$\cdots$	C_K	Total
C_1	n^{11}	n^{21}		n^{K1}	$\hat{n}^1$
C_2	n^{12}	n^{22}		n^{K2}	$\hat{n}^2$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
C_K	n^{1K}	n^{2K}		n^{KK}	$\hat{n}^K$
Total	n^1	n^2	$\ldots$	n^K	n

3.2. Optimizing decision directed graph by minimizing errors on training set

Both DDAG and GDDAG determine the structure in terms of a predefined strategy for selecting the class pairs. Different sequence of class pairs determines different structures. Although the goal to optimize the DDAG is to minimize generalization error, GDDAG implement the goal in a heuristic way. The decision graphs are actually predetermined without being optimized in terms of their performance on training or validation set. It is natural to introduce the strategies in decision tree construction to optimize the DDAG structure by minimizing errors on the training or validation set. Since DDAG naturally generalizes the class of decision trees [29] and implement the same class of functions as that of decision trees, to optimize DDAG with binary classifier in each node is actually to optimize the structure of a decision tree [30].

For an observation n with feature vectors $\boldsymbol{x}_n$ and label y_n, the value of function f_{ij} defined in Eq. (1) for observation n is denoted by f_{ij}^n. Let $\boldsymbol{f}^n = (f_{12}^n, f_{13}^n, \ldots, f_{1K}^n, f_{23}^n, \ldots, f_{K(K-1)}^n)$ be the vector of values of all $K(K-1)/2$ binary classifiers' functions, each observation in the training set for optimizing the decision tree is $(\boldsymbol{f}^n, y_n)$. The training sample set can be denoted by $S' = \{(\boldsymbol{f}^1, y_1), \ldots, (\boldsymbol{f}^N, y_N)\}$, where N is sample size of the training set. The algorithm to optimize the structure of DDAG with objective function of minimizing generalization error on training set is as follows [36]:

Algorithm ODDAG
Input: training set S', feature set $f = \{f_{12}, f_{13}, \ldots, f_{K(K-1)}\}$
Output: the optimized structure of DDAG

1. Examine all possible binary splits on every feature with regard to all observations in training set S'.
2. Select a split of the feature with best optimization criterion.
3. If the split leads to a child node having too few observations, select a split with the best optimization criterion subject to the minimum leaf size constraint.
4. Impose the split.
5. Repeat recursively for the two child nodes until a stopping criterion is triggered.

The selected optimization criteria used to split nodes is the Twoing rule [6,36]. Let $L(i)$ denote the fraction of the members of class i in the left child node after a split, and $R(i)$ denote the fraction of members of class i in the right child node after a split. Twoing rule is to choose the split criterion to maximize

$$P(L)P(R)\left(\sum_i |L(i) - R(i)|\right)^2, \tag{7}$$

where P(L) and P(R) are the fraction of observations that split to the left and right respectively. If the expression in Eq. (7) is large, the split made each child node purer. If the expression is small, the split did not increase node purity.

The stopping rules are as follows [36]:

(1) All observations in the training set belong to a single value of y.
(2) The maximum tree depth has been reached.
(3) The number of cases in the terminal node is less than the minimum number of cases for parent nodes.
(4) If the node were split, the number of cases in one or more child nodes would be less than the minimum number of cases for child nodes.

4. Empirical study

This section reports the results of the experiments for evaluating the proposed methods. First, the experimental framework is presented. Then a general study of the proposed methods tested on public data sets is conducted, which helps to identify the promising methods and their related settings. Finally, the application of the proposed methods on a real problem of listing status prediction are presented.

4.1. Experimental framework

Fig. 3 shows the framework of training and testing ensemble strategies based multi-class classification models. The raw training set splits into different subsets in terms of pairs of classes. The employment of features selection and random sampling is dependent on characteristics of the data sets. If there are a large number of features, feature selection method will be introduced. If the data set is highly imbalanced, random sampling method will be introduced.

4.2. Performance measures

Micro-average (MiA) and Macro-average (MaA) measures are commonly used in evaluating performance on multi-class problems. When the data set is highly imbalanced, MiA does not provide a meaningful measure of performance. However, MaA gives significant performance measure in spite of the level of imbalance in a data set. Therefore, MaA is used to measure the multi-class classifier performance in this study.

Table 2 gives a typical resulting confusion matrix for a problem with K classes, where n^{ij} denotes the number of observations of a class with actual list status i which are predicted as a class with listing status j. $(i = 1, \ldots, K,\ j = 1, \ldots, K)$.

Table 3
Summary for the UCI data sets used in the experiments.

Data set	#Cl	#Fea	# Obs	Names in UCI
Car	4	6	1728	Car evaluation
Lymphography	4	18	148	Lymphography
Vehicle	4	18	846	Statlog (vehicle silhouettes)
Cleveland	5	13	297	Heart disease
Nursery	5	8	1296	Nursery
Pageblocks	5	10	548	Page blocks classification
Shuttle	5	9	2175	Statlog (shuttle)
Autos	6	25	159	Automobile
Dermatology	6	34	358	Dermatology
Glass	7	9	214	Glass identification
Satimage	7	36	643	Statlog (landsat satellite)
Segment	7	19	2310	Statlog (image segmentation)
Zoo	7	16	101	Zoo
Ecoli	8	7	336	Ecoli
Led7digit	10	7	500	LED display domain
Penbased	10	16	1100	Pen-based recognition of handwritten digits
Yeast	10	8	1484	Yeast
Vowel	11	13	990	Vowel recognition - deterding data[a]

[a] The updated vowel data set is retrieved from weka.

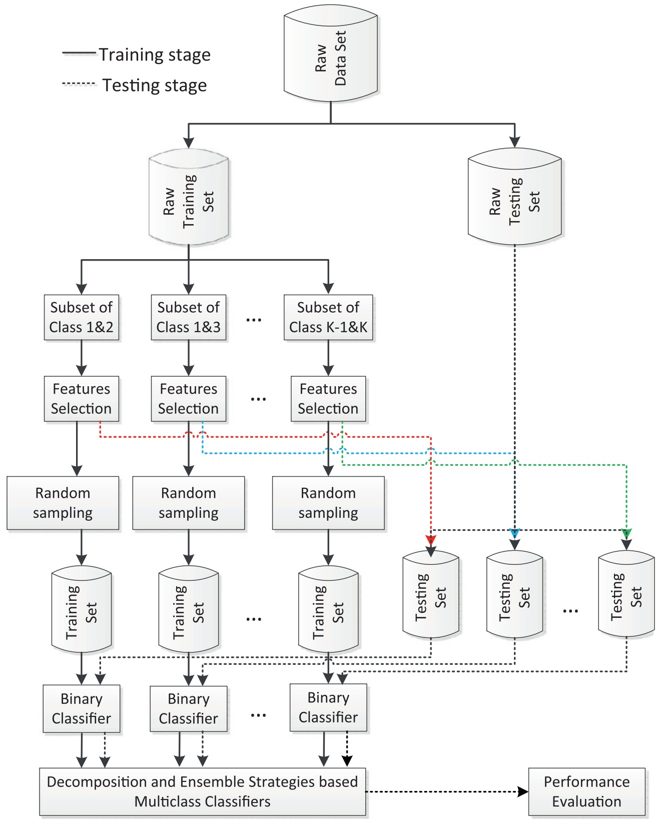


Fig. 3. The framework of training and testing decomposition and ensemble strategies based multi-class classifiers.

The Macro-average, the average of classification accuracy for each of the K classes, is defined as follows:

$$MaA = \frac{1}{K}\sum_{i=1}^{K}\frac{n^{ii}}{n^{i}} \tag{8}$$

The performance of a method on a data set is estimated by a 5-fold stratified cross-validation (SCV). In each of the five rounds of tests, 80% of observations are randomly selected by stratified sampling methods for the training set, the other 20% of observations are used for test. The random selected training set will be checked to assure there is a least one observation from each class. 5-fold SCV is more appropriate than a 10-fold SCV in this experimental framework [28].

4.3. General study

4.3.1. The data sets

This general study contains almost all data sets used in [28] except for the Flare data set, since the definition of predicted classes in the Flare data set is flexible. We retrieved the data sets from the UCI repository [37] and removed all observations with missing values from each data set and take the same number of observations by stratified sampling methods from the original UCI data sets as that used in [28]. The feature of unique identification of observations in each data set is removed. The summary of number of classes (#Cl), the number of features (#Fea), the number of observations (#Obs) and name in UCI of each data set is described in Table 3.

4.3.2. The results

To keep the interpretability of the models, decision tree C4.5 is selected to construct the binary classifier shown in Fig. 3. Since the number of features and observations is not too large, all features and observations in subset of each pairwise classes are used in constructing the binary classifiers. Table A.8 in Appendix shows the average of MaA measure of 5-fold SCV of six different DEMs. The best result of each data set is highlighted in bold-face.

Table A.8 shows that none of the six DEMs can achieve the best performance on all eighteen data sets. The performance of all six DEMs fluctuates a lot on different data sets. For example, the six methods can achieve more than 96% accuracy on Dermatology data set, but none of them can achieve more than 40% on Cleveland data set. OVO decomposition integrated with dynamic classifier selection (OVO-DCS) can achieve the best performance for the most times, followed by OVO-ODDAG for four times and OVO-GDDAG, OVO-WV for three times.

4.4. Listing status prediction

4.4.1. The data set

The data are collected from China Stock Market and Accounting Research Database (CSMARD) provided by the GTA database. There are 23,521 company-year observations dating from 1999 to 2013. 24 observations whose company had disclosed financial statements but were delisted by the end of June in the observed year are removed since the delisting status cannot recover to any other listing statuses. The total number of observations in the training set and testing set is 23,497. Each observation in an observed year t contains the following variables: (1) 167 different financial ratios measuring various aspects of a company's financial status in the fiscal year $t-1$, such as short-term solvency, long-term solvency, asset management or turnover, profitability, capital structure, stock holder's earning profitability, cash management and development capability; (2) three market variables introduced by Shumway [38], including the excess return of the company's stock, relative market capitalization, and the standard deviation of a firm's stock return in the fiscal year $t-1$; (3) the stock market type (Shanghai or Shenzhen, A or B) and the listing status of the company in year t and $t+1$.

The listing rules require all listed companies must declare their financial statements within four months after the end of a fiscal year. China's Securities Regulatory Commission usually enforces the listing status adjustment for a company after the company's financial statements disclosure. Thus, most listing status adjustment on listed companies happened by the end of June. This study is to predict the listing status of a company at the end of June in the observed year t by the information disclosed in year $t-1$. Due to a delisted company cannot recover to any other listing status, it is meaningless to predict the status of a company that has just been delisted.

Most companies with normal listing status "A" retain their "A" listing status in the next year; very few normal companies with listing status "A" will transfer to other listing statuses. Most companies with listing status "B" retain "B" listing status in the following observed year, and some of them may regain listing status "A" because of their improvement in financial performance. In each year, The proportion of companies with listing status "B", "D" and "X" to companies with listing status "A" is very small. Only a very small proportion of companies will slip to "X".

Although there are a total of 172 features in the data set, most existing research [1,38,39] about financial distress prediction demonstrated that the number of features that are statistically significant to discriminate the normal company from the financial distressed company is not more than 10. Ten features will be selected by the feature selection methods for the binary classifiers for two reasons: (1) to make the knowledge from the model for listing status prediction interpretable and simple; (2) to reduce the computational time for modeling.

4.4.2. Experimental settings

A hybrid feature selection method that combines the filter method of two-sample t-test with variance inflation factor (tt-FVIF) [34], is used for features selection. Since the number of different classes is highly imbalanced for the listing status prediction problem, to avoid the bias on any class, random undersampling method is employed to make the final training set balanced. In the testing process, each training set has all observations in the raw training set, but has different features set to match the input of different binary classifier.

The training set consists of observations dating from 1999 to 2007, while the testing set contains observations dating from 2008 to 2013. Because the "A", "B", "D" and "X" classes are highly imbalanced in the training set, the random undersampling algorithm is employed to construct a balanced training sample for the binary classifiers as shown in Fig. 3. The number of selected observations N' for each class by RU is set to the size of minor class set in each pair of classes. The number of observations from different classes in the training set and testing set are listed in Table 4. $N_{training}$ is the number of observations in the data set for sampling training set. $N_{testing}$ is the number of observations in the testing set. Since the sample from the major class in each pair of classes are randomly sampled, the number of sample from class "A" varies and is dependent on the size of intersection set of three sample sets for three different pairs including "A" class. When the binary classifiers are ready, DDAG and GDDAG can be constructed directly without checking the performance of the structure on any validation set. However, ODDAG needs optimize the structure by checking the performance of the structure on a validation set which is the whole training set with a total of 12,960 observations in this study. All multi-class classification models are finally tested on the same testing set with a total of 10,537 observations.

Table 4
The number of observations for different classes and methods in the training set and testing set.

L_t	N_{traing}	N' in training set				$N_{testing}$
		Methods for OVO			OVA	
		B	D	X		
A	11,580	692/692	635/635	53/53	1380/1380	9846
B	692		635/635	53/53	692/692	257
D	635			53/53	635/635	419
X	53				53/53	15

[a] The union of sample sets for all pairs of binary classifiers [b] All samples in the training set are used.

Table 5
Average of MaA performance of DEMs under different ensemble strategies for listing status prediction problem.

DEM	OVA	OVO				
	MC	WV	DCS	DDAG	GDDAG	ODDAG
MaA	52.53 (2.60)	43.04 (4.74)	51.61 (0.83)	51.48 (2.69)	52.15 (3.12)	**52.93** (1.51)

OVA aggregates four binary classifiers, each of which is constructed on one class versus the rest classes. For example, to construct the binary classifier for one class "A" versus the rest class (the combination of classes "B", "D" and "X"), the total number of observations in the rest class is $692+635+53=1380$, 1380 observations from major class "A" are randomly sampled. Therefore, the binary classifier for "A" vs "rest" are trained by 1380 "A" observations and 1380 none "A" observations.

All samples in raw training set are used for the feature selection process shown in Fig. 2. For example, to select the features that can discriminate "A" and "B", all 11,580 "A" and 692 "B" observations are used.

4.4.3. The results

Table 5 shows the average MaA performance of 30 iterations of tests under six different ensemble methods with C4.5 binary classifiers. Since the binary classifiers incorporated in aggregative methods are trained by randomly selected samples with random under-sampling method, to reduce the bias from any one group of samples and demonstrate the robust of the models, 30 iterations of tests are conducted. In each iteration of test, the binary classifier for each pair of classes is trained by balanced sample set in which all samples in the minor class are included and the samples from the major class are randomly selected. The standard deviation of the 30 iterations of tests is reported in the bracket. The best performance among the six DEMs is marked in bold. The OVO-ODDAG can achieve the best MaA performance of 52.93%. The standard deviations of OVO-ODDAG in the 30 iterations of test on MaA measure is 1.51%, which demonstrates good robustness of OVO-ODDAG. All DEMs cannot achieve a high MaA performance because they fail to obtain a high classification accuracy on the minor classes.

Table 6
The average rank of each ensemble methods with C4.5 on MaA measure.

DEM	OVA	OVO				
	MC	WV	DCS	DDAG	GDDAG	ODDAG
MaA	2.73	5.73	3.57	3.50	3.03	**2.43**

To make a statistical comparison among the six DEMs, Nemenyi test is introduced. The Nemenyi test [14] shows that if the difference of average ranks between any two methods is less than 1.38 (significance level $\alpha = 0.05$, the number of methods is six), the difference is not statistically significant. Table 6 shows that OVO-ODDAG can achieve the top average rank on MaA performance in the 30 iterations of tests. However, among other five aggregative methods, OVA-MC, OVO-DCS, OVO-DDAG, and OVO-GDDAG have no significant difference from OVO-ODDAG on the MaA performance measure.

4.4.4. Misclassification cost sensitivity analysis for ODDAG

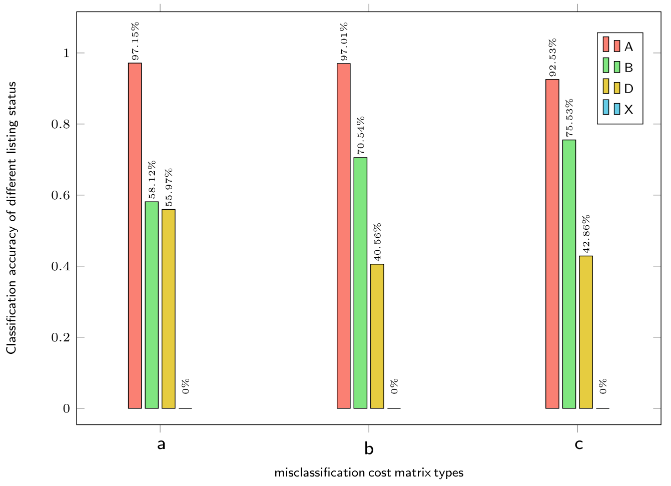
None of the employed models can achieve the correct classification rate on “X” observations as high as that on observations with other listing statues. The classification accuray on “X” from all models is almost zero or slightly greater than zero. It indicates that to predict whether a listed company will be delisted is very difficult. There are four possible reasons as follows:

(1) The data set is highly imbalanced. There are a total of 23,521 observations and only 68 “X” observations. 53 “X” observations are not enough for training the models and make the models catch the characteristics of delisted companies, therefore to correctly predict the 15 “X” observations among the 10,537 observations in the test set is extremely difficult.
(2) Some delisted companies given a delisting risk warning was not due to their financial performance but rather because of their operations risk and negative audits from accounting agencies, thus to improve prediction accuracy on “X” companies needs information about the company’s governance and business operations.
(3) Many listed companies in financial distress use the strategy of corporate restructuring to avoid delisting.
(4) In model construction, unsymmetrical misclassification cost has not been considered.

ODDAG methods can consider misclassification cost in optimizing the decision tree structure easily to adjust the performance on different classes. The default setting for misclassification cost matrix denoted by E is shown as Fig. 4(a). The element E_{ij}, at the row i and column j, means the misclassification cost of classifying an observation into i if its true class is j. The rows of matrix E correspond to the true classes and the columns correspond to the predicted classes.

$$\underset{(a)}{\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}} \quad \underset{(b)}{\begin{pmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 1 & 3 \\ 2 & 1 & 0 & 1 \\ 3 & 3 & 1 & 0 \end{pmatrix}} \quad \underset{(c)}{\begin{pmatrix} 0 & 1 & 2 & 3 \\ 6 & 0 & 1 & 3 \\ 8 & 1 & 0 & 3 \\ 10 & 3 & 1 & 0 \end{pmatrix}}$$

In practical decision making, the misclassification cost is difficult to estimate. If a “A” company is predicted as a “B”,“D” or “X” company, the investors will not make an investment on the company and the misclassification may cause opportunity cost for the investors. If a “B” or “D” company is predicted as a “A” company and the investors make investments on the company, the investors may suffer a great loss. To predict a “B” company as a “D” company or to predict a “D” company as a “B” company may not cause much opportunity cost or loss, because both classes can indicate the risk warning to the investors in spit of the risk level may be a little bit different.



(d)

Fig. 4. The misclassification cost matrices and the correct classification rate of OVO-ODDAG-DT with different cost matrices.

Due to the difficulty of estimating the misclassification cost matrix, a sensitivity analysis of misclassification cost matrix on the ODDAG models is conducted. As shown in Table 5, ODDAG model with C4.5 binary classifier achieve the highest MaA performance, thus the OVO-ODDAG is used for the cost matrix sensitivity analysis. Three different misclassification matrices shown in Fig. 4(a)–(c) are employed. The correct classification rate on each class is shown in Fig. 4(d).

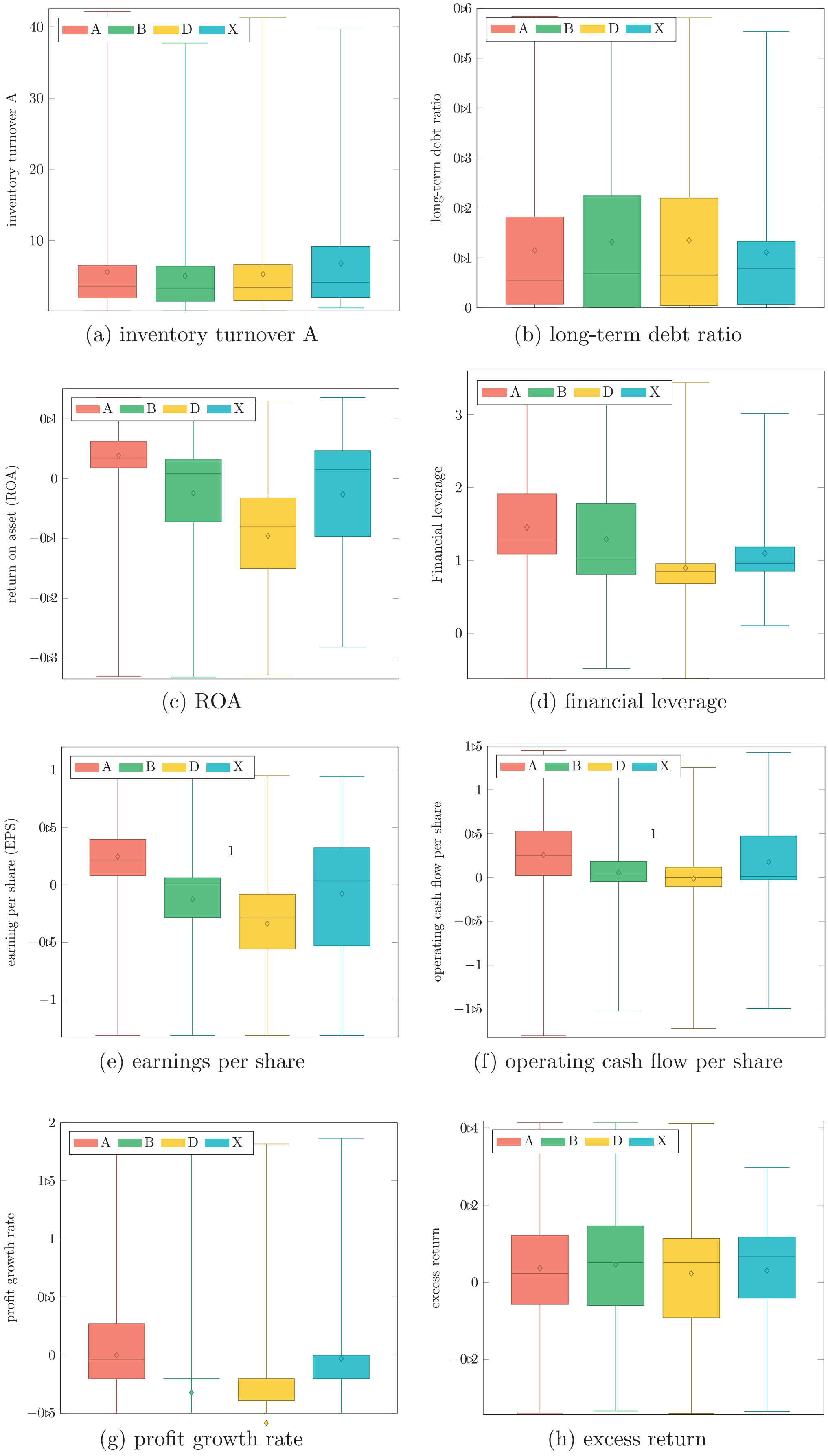
Fig. 4(d) shows that when the cost of misclassifying non-“A” company as “A” company increase, the classification accuracy on “A” class decrease while the classification accuracy on “B” increase. However, the classification accuracy on “D” decrease when changing cost matrix from (a) to (b) or (c). The cost matrix can only help to adjust the accuracy balance on the different classes and cannot actually increase the discriminative capability of the models. Therefore, it is hard to improve the classification accuracy on all four classes by adjustment on the cost matrix. As shown in Fig. 4(d), the OVO-ODDAG model cannot correctly predict “X” company even with different cost matrix. It indicates that the maximum cost 10 of misclassifying a true “X” as other classes is not enough to make the model to classify an observation to “X”. Theoretically, if the misclassification cost of classifying “X” into other classes is enough, an observation has larger opportunity to be classified as “X” but it must sacrifice the classification accuracy on other classes.

4.4.5. Knowledge from DEM for listing status prediction

(1) Analysis of selected features

There are six pairs of classes and the tt-FVIF is used to select 10 features for each pair. The union set of each group of 10 features from each pair classes contains 43 different financial variables. The CSMARD categorizes the financial ratios into eight groups to measure different aspects of a company. In this study, four additional market variables used by some existing research on financial distress prediction are added to the data set. The categories distribution of the selected 43 variables is shown in Table 7.

Since the features in one category measure similar characterisitc of a company, only one feature is selected as a representative for demonstrating their distribution difference by different listing statuses. Fig. 5(a)–(h) show the boxplot of one representative feature selected from each of the eight categories listed in Table 7 by



(a) inventory turnover A

(b) long-term debt ratio

(c) ROA

(d) financial leverage

(e) earnings per share

(f) operating cash flow per share

(g) profit growth rate

(h) excess return

Fig. 5. Boxplot of representative feature in each category by listing statues.

Table 7
The frequency distribution of the selected features.

Categories	Frequency
Business operation capability	8
Long-term debt paying capability	3
Profit earning capability	9
Leverage level	2
Shareholders' profit earning capability	10
Cash flow capability	4
Development capability	5
stock market performance	2
Total	43

different listing statues. The feature characteristic of companies with different listing statuses can be roughly discussed as follows:

(a) The inventory turnover A is defined as the ratio of operating cost to inventory outstanding balance. There is no signifiant difference among the "A", "B" and "D" company, "X" company has a larger average.
(b) The companies with different listing status have similar percentile range of long-term debt ratio, but the median value of long-term debt ration of abnormal companies is greater than that of normal companies. It indicates that abnormal companies has a greater average long-term debt ratio than normal companies.
(c) The normal company has higher ROA than abnormal company. Most abnormal companies have negative ROA.
(d) "D" and "X" company has very narrow percentile range of financial leverage and their leverage level is lower than "A" and "B" company.
(e) The normal company has larger EPS than abnormal companies. It indicates the abnormal company has week earning capability.
(f) The median operating cash flow per share from normal companies is greater than that of abnormal company.
(g) The normal company has a larger profit growth rate.
(h) Abnormal companies have higher excess return of stock, because the stock of abnormal companies has higher risk. It matches the financial rules in stock market: "higher risk, higher return".

(2) Decision rules

The structure of OVO-ODDAG is actually a decision tree which takes the output of six binary classifiers as input and fits the training set. One example of OVO-ODDAG model for listing status prediction is partially shown in Fig. 6. Since the binary classifier in OVO-ODDAG is also a decision tree, the decision process of OVO-ODDAG with C4.5 binary classifier is rule-based and is not a black-box. Because different feature groups are used by the different binary classifiers, the decision trees of all of the six binary classifiers construct a forest. One advantage of the OVO-ODDAG is that the decision makers can understand and check the rules disclosed in the decision tree and they can modify the decision tree and incorporate their own rules.

The complete structure of OVO-ODDAG shown in Fig. 6 contains 10 layers. It does not include any leaf node of "X", which can explain why the correct classification rate on "X" class from OVO-ODDAG methods shown in Fig. 4(d) is almost zero. It may be the disadvantage of OVO-ODDAG method that it has poor classification accuracy on the minor class for the highly imbalanced data set.

5. Conclusion

The prediction of listing statuses of Chinese listed company is considered as a multi-class classification problem in this study, while most of existing literature consider it as a binary classification problem by predicting whether a normal company will receive special treatment or not.

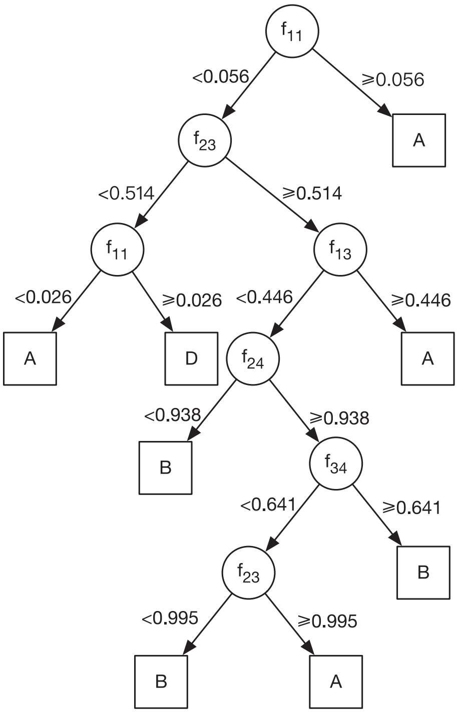


Fig. 6. Partial structure of one OVO-DDAG-DT model.

A new learning architecture based on optimizing decision directed acyclic graph was proposed and was compared to some widely used methods for multi-class classification problems, such as weighted voting, dynamic classifiers selection, generalization error guided DDAG, etc. The performance of ODDAG structure achieves the top average rank among the six methods for predicting listing status of companies. The ODDAG incorporating decision tree C4.5 achieves the highest classification accuracy in the 30 iterations of the tests. Its small standard deviation on macro-average in the 30 iterations of tests demonstrates that the ODDAG models are robust under different training sets. The disadvantage of all models in this study is that they cannot achieve reasonable performance on "X" classes.

The OVO-ODDAG model is a white box and the rules for classification can be viewed and checked by the decision makers. When the visibility of decision process is a great concern from the professional financial analysts, there may be a balance between the complexity and predictive performance of ODDAG models.

In this study, the ODDAG aggregative strategy is only applied for OVO scheme. Future research will extend its application for OVA, combination of OVA and OVO, or ECOC. Since dynamic classifier selection strategy can integrate any aggregative method, the integration of ODDAG with DCS which can reduce the complexity of decision structure of ODDAG and may improve performance will be another direction in future research.

Acknowledgement

We would like to thank two anonymous reviewers for their helpful comments and suggestions. This work is partly supported by the National Natural Science Foundation of China (NSFC No. 71671013).

Appendix A. MaA performance of different DEMs

Table A.8
Average of MaA measures of six different DEMs on UCI data sets.

Dataset	OVA	OVO				
	MC	WV	DCS	DDAG	GDDAG	ODDAG
Car	95.21	96.34	96.98	96.15	96.52	**97.09**
Lymphography	49.05	**58.58**	46.08	46.08	46.08	46.08
Vehicle	70.22	70.84	71.53	70.99	**72.17**	70.76
Cleveland	29.28	28.25	27.82	29.87	**31.89**	31.09
Nursery	**73.71**	73.48	72.53	73.48	73.48	73.48
Pageblocks	60.76	57.04	52.04	56.96	56.96	**60.92**
Shuttle	69.82	68.15	74.87	68.19	78.14	**78.15**
Autos	75.45	74.62	62.09	74.67	**75.81**	73.64
Dermatology	94.90	**96.05**	94.17	95.49	95.01	92.41
Glass	**68.84**	61.39	63.48	63.27	64.24	55.38
Satimage	75.54	77.64	**79.24**	77.65	76.33	76.48
Segment	94.69	96.79	**96.83**	96.47	96.37	96.65
Zoo	76.90	80.08	**85.79**	80.08	80.08	68.33
Ecoli	51.37	60.32	**61.21**	58.35	60.93	56.20
Leddigit7	70.68	71.14	71.03	70.55	70.87	**73.00**
Penbased	85.55	88.15	**89.60**	85.10	85.30	85.99
Yeast	44.38	**50.20**	47.88	47.98	46.89	44.70
Vowel	75.94	78.91	**81.41**	78.10	75.54	76.90

References

[1] M. Aly, Survey on multiclass classification methods, Technical Report, Caltech, 2005.

[2] H. Matsui, T. Araki, S. Konishi, Multiclass functional discriminant analysis and its application to gesture recognition, J. Classification 28 (2) (2011) 227–243.

[3] S. Ghorai, A. Mukherjee, P.K. Dutta, Discriminant analysis for fast multiclass data classification through regularized kernel function approximation, IEEE Trans. Neural Netw. 21 (6) (2010) 1020–1029.

[4] Y.Q. Guo, T. Hastie, R. Tibshirani, Regularized linear discriminant analysis and its application in microarrays, Biostatistics 8 (1) (2007) 86–100.

[5] M. Suresha, A. Danti, S.K. Narasimhamurthy, Decision trees to multiclass prediction for analysis of arecanut data, Comput. Syst. Sci. Eng. 29 (1) (2014) 105–114.

[6] L. Breiman, J. Friedman, R. Olshen, C. Stone, Classification and Regression Trees, CRC Press, Boca Raton, Florida, 1984.

[7] O. Maimon, L. Rokach, Data Mining and Knowledge Discovery Handbook, Springer, New York, 2005.

[8] W.H. Ahn, S.P. Nah, B.S. Seo, Automatic classification of digitally modulated signals based on k-nearest neighbor, Lect. Notes Electr. Eng. 329 (2015) 63–69.

[9] V. Metsis, I. Androutsopoulos, G. Paliouras, Spam filtering with naive bayes - which naive bayes?, in: The Third Conference on Email and Anti-Spam - Proceedings, CEAS 2006, Mountain View, California.

[10] T.A. Almeida, J. Almeida, A. Yamakami, Spam filtering: how the dimensionality reduction affects the accuracy of naive bayes classifiers, J. Internet Serv. Appl. 1 (3) (2011) 183–200.

[11] M.L. Lin, K. Tang, X. Yao, Dynamic sampling approach to training neural networks for multiclass imbalance classification, IEEE Trans. Neural Netw. Learn. Syst. 24 (4) (2013) 647–660.

[12] W. Kim, H.K. Lee, J. Park, K. Yoon, Multiclass adult image classification using neural networks, Lect. Notes Comput. Sci. 3501 (2005) 222–226.

[13] G.B. Ou, Y.L. Murphey, Multi-class pattern classification using neural networks, Pattern Recognit. 40 (1) (2007) 4–18.

[14] C.W. Hsu, C.J. Lin, A comparison of methods for multiclass support vector machines, IEEE Trans. Neural Netw. 13 (2) (2002) 415–425.

[15] A. Anand, P.N. Suganthan, Multiclass cancer classification by support vector machines with class-wise optimized genes and probability estimates, J. Theor. Biol. 259 (3) (2009) 533–540.

[16] I. Guler, E.D. Ubeyli, Multiclass support vector machines for EEG-signals classification, IEEE Trans. Inf. Technol. Biomed. 11 (2) (2007) 117–126.

[17] S.C. Du, D.L. Huang, J. Lv, Recognition of concurrent control chart patterns using wavelet transform decomposition and multiclass support vector machines, Comput. Ind. Eng. 66 (4) (2013) 683–695.

[18] L.P.F. Garcia, J.A. Sáez, J. Luengo, A.C. Lorena, A.C.P.L.F. de Carvalho, F. Herrera, Using the one-vs-one decomposition to improve the performance of class noise filters via an aggregation strategy in multi-class classification problems, Knowl. Based Syst. 90 (2015) 153–164.

[19] M. Paz Sesmero, J.M. Alonso-Weber, G. Gutierrez, A. Ledezma, A. Sanchis, An ensemble approach of dual base learners for multi-class classification problems, Inf. Fusion 24 (2015) 122–136.

[20] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, F. Herrera, An overview of ensemble methods for binary classifiers in multi-class problems: experimental study on one-vs-one and one-vs-all schemes, Pattern Recognit. 44 (8) (2011) 1761–1776.

[21] M. Galar, J. Derrac, D. Peralta, I. Triguero, D. Paternain, C. Lopez-Molina, S. Garca, J.M. Bentez, M. Pagola, E. Barrenechea, H. Bustince, F. Herrera, A survey of fingerprint classification part I: taxonomies on feature extraction methods and learning models, Knowl. Based Syst. 81 (2015) 76–97.

[22] M. Galar, J. Derrac, D. Peralta, I. Triguero, D. Paternain, C. Lopez-Molina, S. Garca, J.M. Bentez, M. Pagola, E. Barrenechea, H. Bustince, F. Herrera, A survey of fingerprint classification part II: experimental analysis and ensemble proposal, Knowl. Based Syst. 81 (2015) 98–116.

[23] J.A. Sáez, M. Galar, J. Luengo, F. Herrera, Analyzing the presence of noise in multi-class problems: alleviating its influence with the one-vs-one decomposition, Knowl. Inf Syst. 38 (1) (2014) 179–206.

[24] P. Sun, M.D. Reid, J. Zhou, An improved multiclass logitboost using adaptive-one-vs-one, Mach. Learn. 97 (3) (2014) 295–326.

[25] Y. Shiraishi, Game theoretical analysis of the simple one-vs.-all classifier, Neurocomputing 71 (13–15) (2008) 2747–2753.

[26] E.L. Allwein, R.E. Schapire, Y. Singer, Reducing multiclass to binary: a unifying approach for margin classifiers, J. Mach. Learn. Res. 1 (2000) 113–141.

[27] A.C. Lorena, A.C.P.L.F. De Carvalho, J.M. Gama, A review on the combination of binary classifiers in multiclass problems, Artif. Intell. Rev. 30 (2008) 19–37.

[28] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, F. Herrera, Dynamic classifier selection for one-vs-one strategy: avoiding non-competent classifiers, Pattern Recognit. 46 (12) (2013) 3412–3424.

[29] J.C. Platt, N. Cristianini, J. Shawe-Taylor, Large margin DAGs for multiclass classification, in: Advances in Neural Information Processing Systems, 2000, pp. 547–553.

[30] F. Takahashi, S. Abe, Optimizing directed acyclic graph support vector machines, in: Artificial Neural Networks in Pattern Recognition (ANNPR), 2003, pp. 166–173.

[31] L. Zhang, E.I. Altman, J. Yen, Corporate financial distress diagnosis model and application in credit rating for listing firms in china, Front. Comput. Sci. China 4 (2) (2010) 220–236.

[32] J. Chen, B.R. Marshall, J. Zhang, S. Ganesh, Financial distress prediction in china, Rev. Pacific Basin Financ. Markets Policies 9 (2) (2006) 317–336.

[33] L. Zhou, K.K. Lai, J. Yen, Empirical models based on features ranking techniques for corporate financial distress prediction, Comput. Math. Appl. 64 (8) (2012) 2484–2496.

[34] L. Zhou, K.P. Tam, H. Fujita, Predicting the listing status of chinese listed companies with multi-class classification models, Inf. Sci. 328 (2016) 222–236.

[35] E. Hüllermeier, S. Vanderlooy, Combining predictions in pairwise classification: an optimal adaptive voting strategy and its relation to weighted voting, Pattern Recognit. 43 (1) (2010) 128–142.

[36] Statistics and Machine Learning Toolbox Release 2014a, The MathWorks, Inc., Natick, Massachusetts, United States, 2014.

[37] A. Asuncion, D. Newman, UCI machine learning repository, 2007.

[38] T. Shumway, Forecasting bankruptcy more accurately: a simple hazard model, J. Bus. 74 (1) (2001) 101–124.

[39] H. Li, Y.C. Lee, Y.C. Zhou, J. Sun, The random subspace binary logit (RSBL) model for bankruptcy prediction, Knowl. Based Syst. 24 (8) (2011) 1380–1388.