



A new re-encoding ECOC using reject option

Lei Lei¹ · Yafei Song² · Xi Luo¹

Published online: 1 May 2020

© The Author(s) 2020

Abstract

When training base classifier by ternary Error Correcting Output Codes (ECOC), it is well known that some classes are ignored. On this account, a non-competent classifier emerges when it classifies an instance whose real label does not belong to the meta-subclasses. Meanwhile, the classic ECOC dichotomizers can only produce binary outputs and have no capability of rejection for classification. To overcome the non-competence problem and better model the multi-class problem for reducing the classification cost, we embed reject option to ECOC and present a new variant of ECOC algorithm called as Reject-Option-based Re-encoding ECOC (ROECOC). The cost-sensitive classification model and cost-loss function based on Receiver Operating Characteristic (ROC) curve are built respectively. The optimal reject threshold values are obtained by combining the condition to be met for minimizing the loss function and the ROC convex hull. In so doing, reject option (t_1, t_2) provides a three-symbol output to make dichotomizers more competent and ROECOC more universal and practical for cost-sensitive classification issues. Experimental results on two kinds of datasets show that our scheme with low-degree freedom of initialized ECOC can effectively enhance accuracy and reduce cost.

Keywords Error-correcting output codes · Cost-sensitive · Reject option · Receiver operating characteristics

1 Introduction

Uncertainty caused by incomplete data has become a great challenge to the problem of pattern classification [1–5]. Multi-class classification using Error Correcting Output Codes (ECOC), first proposed by Dietterich and Bakiri [6] in 1995, attracts attention due to its excellent performance. As a decomposition framework, ECOC method effectively reduces a complex multi-class problem into a set of binary problems. ECOC Classification simplifies the complexity of pattern recognition and uses the state-of-the-art binary classifiers for multi-class classification. So far, ECOC has been widely applied to spectrum sensing [7], image recognition [8, 9] and disease and fault diagnosis [10, 11] with fairly good classification performance.

There are two steps when using ECOC methods to solve the multi-class issues: the encoding process and the decoding process. The goal of encoding is to construct a matrix $\mathbf{M} = (m_{ij})_{c \times l}$, $m_{ij} \in \{1, 0, -1\}$ where rows hold the code words of the class and columns represent bipartitions for the dichotomizers. The classes denoted by zero are ignored in training. The decoding strategy is chosen to merge the outputs of base classifiers. The framework of ECOC is described in Fig. 1:

Encoding as the first step is especially crucial. Three main encoding methods are mainly predefined code, data-dependent code and dichotomies-based code [12]. Independent of the specific application and the classes used to train dichotomizers, the predefined code ignores the potential information of original classes and confines the improvement of classification accuracy, including one-versus-one matrix, one-versus-all matrix, dense and sparse random matrices. The dichotomies-based code involves finding an optimal code matrix given a set of binary classifiers, proven to be NP-complete by Crammer and Singer [12].

However, the data-dependent code can make the most of class separability among samples to enhance classification performance as a whole, which has drawn special attention. The classic data-driven ECOC variants are

✉ Yafei Song
yafei_song@163.com

¹ College of Information and Navigation, Air Force Engineering University, Xi'an 710077, China

² College of Air and Missile Defense, Air Force Engineering University, Xi'an 710051, China

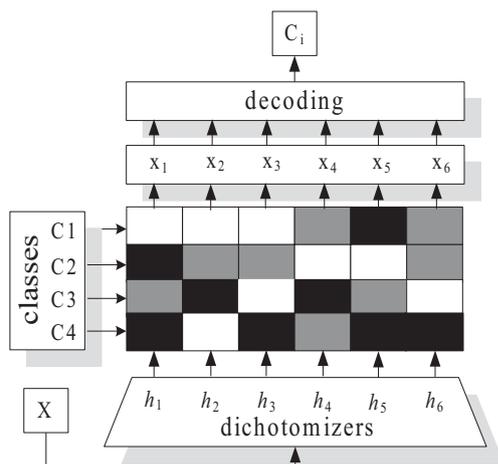


Fig. 1 four-class classification using ternary ECOC

addressed such as Discriminant ECOC [13], Subclass ECOC [14], Linear Discriminant ECOC [15], and Hierarchical ECOC [16]. As Wang et al. [17] notes: “ECOC brings a simple and common implementation of multi-class classification, but simultaneously, results in the under-exploitation of already-provided structure knowledge in individual original classes”. In Wang’s work, they show that the utilization of such prior structure knowledge improves the performance of ECOC. Escalera [18] proposed re-coding ECOC without re-training. Zhou et al. [19] used the Fisher formulation to construct an ECOC coding matrix by using confusion matrix. Experiments with the UCI dataset proved that the approach could offset the base classifiers’ error. To overcome the small samples size and overlaps among classes, Zhong [20] has proposed a new Self-Adjusting ECOC technique to generate diverse code matrices based on different feature subset in terms of the data complexity measure. Rocha et al. [21] took the correlation and joint probability of base binary learner into consideration when using ECOC-based approaches. Sun et al. [22] put a new ECOC algorithm to Enhance Class Separability named as ECOC-ECS by obtaining the optimal class split based on Data Complexity during encoding process. Zhao et al. [9] used deep neural networks as base classifiers to exhibits superior performance in spite of long computation time. Marie et al. [23] proposed an efficient modeling from the coding and decoding process to best gain the data information to enhance performance.

In order to form data-dependent codes to promote classification performance, two problems of ECOC classification are addressed in this paper. The first one is the non-competence [24] problem that the dichotomizers in ternary ECOCs exclude the prior information of the classes denoted by zero. On this account, a non-competent classifier emerges when it classify an instance whose real class does not belong to the meta-subclasses. Our aim is how to use the knowledge of the problem domain to design better codes and reduce

classification error. The other is the cost-sensitive classification problem. It is well known that the common criterion for evaluating the performance of a base classifier is classification error. The smaller the error, the better the performance. However, when considering the classification risk and dealing with class imbalance problem [25], the classification error become infeasible. At this point, it is better and costs less to reject an unlabeled sample than to misclassify. The classic ECOC dichotomizers can only produce binary outputs and have no capability of rejection for classification. So it is a need to modify dichotomizers to give a third output and be adapted to cost-sensitive classification in terms of cost loss.

Taking the analysis above into consideration, a new variant of ECOC algorithm called as Reject-Option-based Re-encoding ECOC (ROECOC) is presented in this paper. ROECOC embeds a reject option to ECOC and find the best threshold values for constructing data-driven codes by using the arbitrary ECOC as initialized matrix based on ROC curve. Then, the optimal base classifier with reject option which can produce three-symbol output and classify selectively is obtained and used to classify the corresponding classes denoted by zeros. Finally, the initial coding matrix is recoded according to the optimal classifier output and thresholds without retraining. The decoding strategy suitable for three-symbol-output classifiers is discussed at the same time. The ROECOC not only uses the prior class information to construct data-driven matrix and also is applicable to the cost-sensitive classification.

The paper is organized as follows: Section 2 discusses the idea of presented ROECOC. Section 3 focuses on the proposed re-encoding ECOC and addresses the potential problems. Section 4 presents the experiments and results, and section 5 concludes the paper.

2 Re-encoding idea

As a more universal algorithm, ternary ECOC [26] method nearly unify all decomposing frameworks for multi-class classification. The introduction of a third output value of

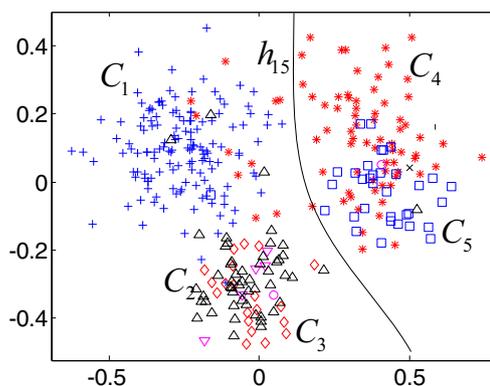


Fig. 2 The distribution of five-class dataset

zero makes the construction of the encoding matrix more flexible and diverse, which greatly promotes the development of ECOC classification. However, there still exist some problems.

On the one hand, the dichotomizers may provide some useful decision information for the classes ignored. Suppose a five-class dataset with a mixed Gaussian distribution as shown in Fig. 2. The decision boundary of classifier h_{15} trained with C_1 and C_5 can classify $C_2, C_3,$ and C_4 correctly for the most of samples. On the other hand, the dichotomizers are non-competent for class ignored. We consider a matrix $M_{4 \times 7}$ shown in Fig. 3. The testing instance x might be classified as C_1 according to Hamming and Euclidean distance decoding. However, from another perspective, x belongs to class C_2 because the predicted class label of x is in accordance with C_2 as long as C_2 was not ignored during base classifier training. If C_2 is ignored, the classifier's decision boundary cannot make the correct classification. This is explained by the disturbance introduced by zero, which introduces the classification error into the decoding process. When there are zeros in the coding matrix, the dichotomizers do not have the distribution information of the corresponding classes and fail to make the right decision.

The non-competence problem has been discussed by Escalera [18] who proposed a re-encoding approach without retraining and used the classification accuracy as the threshold to recode the zeros. However, how to determine the thresholds is still an open issue.

On account of the above situation, it is easy to handle when dichotomizers classify the classes ignored correctly. If not, it is better that the outputs of dichotomizers remain zeros. Therefore, the output of dichotomizers must expand from $\{1, -1\}$ to $\{1, 0, -1\}$ and zero means rejection. It is well known that the binary output have no capability of rejection for classification. The most commonly used method is to construct a reject option $(t_1, t_2), t_2 > t_1$ to produce a third output to classify selectively. The question is how to form the reject option.

As an efficient means of classifier performance assessment, the concept of ROC is clear, intuitive, and independent of prior distribution information, base classifiers, and cost matrices, which provides a powerful tool for the construction of reject option. Tortorella [27] presented a 2D ROC-based rejection

	h_1	h_2	h_3	h_4	h_5	h_6	h_7	HD	ED
C_1	1	1	-1	-1	1	-1	-1	1	4
C_2	1	-1	0	0	0	0	0	2.5	5
C_3	-1	1	1	1	-1	-1	1	6	12
C_4	0	1	1	0	-1	1	-1	4	8
x	1	-1	-1	-1	1	-1	-1		

Fig. 3 $M_{4 \times 7}$ coding matrix

mechanism, but the experiments show that the decision thresholds are the same and irrelevant when the datasets are small. Bernard et al. [28] find that it is more efficient to exploit the ROC space for learning a pool of classifiers than a single classifier. Zhao [29] models the loss functions of cost-sensitive issue with rejection option and obtains the thresholds via the tangent of ROC curve. In binary classification, the ROC curve has shown to be very powerful for designing cost-sensitive classifiers, but has been poorly exploited for multi-class classification. It is well known that the ECOC base classifier cannot produce the precise posteriori probability, so $t_1 + t_2 \neq 1$ and the reject option for each dichotomizer is different. How to formulate the decision of the three-symbol output and model the minimal cost-loss objective function and constraints is the premise of obtaining reject option based on ROC for multi-class classification. To solve the questions, we construct the reject cost matrix at first and find a formula which can formulate the three-symbol output and use the cost loss as objective function at the same time. Finally, the thresholds values are calculated with the help of ROC convex hull.

In conclusion, ROECOC applies ROC to design the reject option, extending two-symbol output into three-symbol output. Then, the ignored classes are classified by the corresponding optimal classifiers as 1,-1 or 0 respectively. Figure 4 shows the framework of the re-encoding ROECOC based on ROC, where S_{train} and S_V represent the training and validation subsets respectively. By doing so, a new data-driven matrix including more class information is obtained. It is worth noting that the re-encoding process remains in the training phase, avoiding a second training step and reducing complexity. The three-symbol output classifiers can apply to the cost-sensitive classification.

3 Reject option-based ECOC by using ROC

This section focuses on ROECOC and explains the determination of the reject option (t_1, t_2) based on the cost-sensitive model and ROC curve [30].

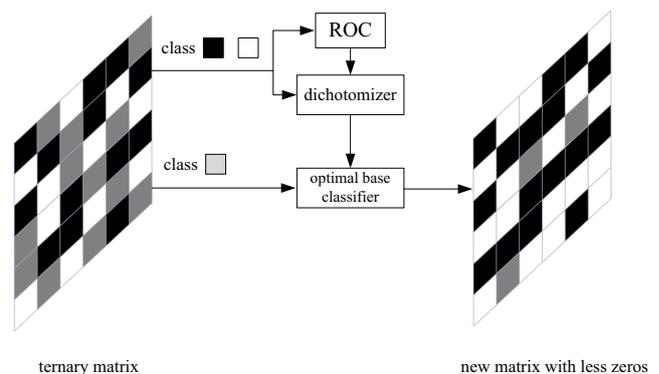


Fig. 4 The framework of the re-encoding ROECOC

A confusion matrix in Fig. 5 can be acquired after the bipartition training. The true positive rate and false positive rate can be calculated by $tp = \frac{TP}{TP+FN} = \frac{TP}{P}$, $fp = \frac{FP}{FP+TN} = \frac{FP}{N}$.

In order to realize rejection classification:

$$w_i(x, t_1^i, t_2^i) = \begin{cases} 1 & \text{if } f_i(x) > t_2^i \\ -1 & \text{if } f_i(x) < t_1^i \\ \delta & \text{otherwise} \end{cases} \quad (1)$$

we define a new cost matrix $C_r = \begin{matrix} & 1 & -1 & \delta \\ 1 & 0 & c_{12} & c_{13} \\ -1 & c_{21} & 0 & c_{23} \end{matrix}$, in

which $c_{11} = c_{22} = 0$, $c_{12} > c_{13} > 0$ and $c_{21} > c_{23} > 0$.

It is hard to acquire the reject option thresholds directly in practice. The classic binary output of classifiers can be described as

$$w_i(x, r) = \begin{cases} 1 & \text{if } f_i(x) > r \\ -1 & \text{if } f_i(x) < r \end{cases} \quad (2)$$

, and r is the cutoff value. The eq. (1) can be seen as the voting rules under some constraints of eq. (2):

$$w_i(x, r_1, r_2) = \begin{cases} 1 & \text{if } f_{t_1}(x) > r_1 \wedge f_{t_2}(x) > r_2 \\ -1 & \text{if } f_{t_1}(x) < r_1 \wedge f_{t_2}(x) < r_2 \\ \delta & \text{if } f_{t_1}(x) < r_1 \wedge f_{t_2}(x) > r_2 \vee f_{t_1}(x) > r_1 \wedge f_{t_2}(x) < r_2 \end{cases}$$

s.t. $\forall x w_i(x, r_i) = 1 \Rightarrow w_j(x, r_j) = 1 \wedge w_j(x, r_j) = -1 \Rightarrow w_i(x, r_i) = -1$ (3)

Therefore the reject option can be constructed by finding a formula which can meet the constraints of eq. (3) and use the cost loss as objective function at the same

		True label	
		P	N
Prediction	P	True Positives (TP)	False Positives (FP)
	N	False Negatives (FN)	True Negatives (TN)

Fig. 5 Confusion matrix

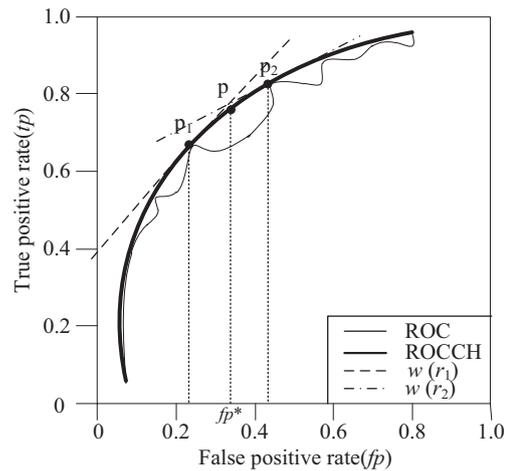


Fig. 6 ROC, ROCCH and the optimal classifier $f_{ROC}(fp^*)$

time. On this account, we use the eq. (2) as the decision rules to obtain ROC curve. The ROC convex hull can be acquired by fitting the ROC curve described in Fig. 6 in heavy line. According to the characteristics of ROCCH, any two points are satisfied with the constraints of eq. (3), so the question is transferred to how to find the two points p_1, p_2 to meet our need. Suppose the corresponding confusion matrices of the two points (base classifiers) are (TP_1, FN_1, FP_1, TN_1) and (TP_2, FN_2, FP_2, TN_2) respectively.

The classification cost loss function can be defined as:

$$EC = \frac{(FP_2 - FP_1)c_{23} + (FN_1 - FN_2)c_{13} + FP_1 \cdot c_{21} + FN_2 \cdot c_{12}}{TP + FN + FP + TN} = \frac{(FN_1 \cdot c_{13} + FP_1 \cdot (c_{21} - c_{23}) + FN_2 \cdot (c_{12} - c_{13}) + FP_2 \cdot c_{23}}{TP + FN + FP + TN} \quad (4)$$

Note that

$$P = FN + TP \Rightarrow FN = P - TP = P \left(1 - \frac{TP}{P} \right) = P(1 - tp) = P(1 - f_{ROC}(fp))$$

Then the eq. (4) can be rewritten as:

$$EC = P \left(1 - f_{ROC} \left(\frac{FP_1}{N} \right) \right) c_{13} + FP_1 (c_{21} - c_{23}) + P \left(1 - f_{ROC} \left(\frac{FP_2}{N} \right) \right) (c_{12} - c_{13}) + FP_2 \cdot c_{23} \quad (5)$$

Take the partial derivatives of eq. (5) for FP_1 and FP_2 , then we can get the final results:

$$f'_{ROC}(fp_1^*) = \frac{c_{21}-c_{23}}{c_{13}} \frac{N}{P}$$

$$f'_{ROC}(fp_2^*) = \frac{c_{23}}{c_{12}-c_{13}} \frac{N}{P}$$
(6)

fp_1^* and fp_2^* are the two points we find and used as the thresholds values of reject option (t_1, t_2) .

After embedding reject options for each classifier, the outputs become three-symbol. In order to decode the output of reject option, the state-of-the-art Hamming decoding strategy is extended as follows according to the decoding process of the ternary ECOC [31]:

- Step 1: Replace all the rejected code words denoted as zero with “-1”, and use the classic Hamming distance to find the nearest class code c_{-1} ;
- Step 2: Replace all the rejected code words denoted as zero with “1”, and use the classic Hamming distance to find the nearest class code c_1 ;
- Step 3: Compare the distance of the non-rejected code words between c_{-1} and c_1 , and the nearer class code is the final decision.

After solving the construction of reject option and decoding strategy, the ROECOC process is presented particularly in Table 1.

The reject option is trained by the original binary splits, which can ensure the independence and diversity of dichotomizers. Meanwhile, the outputs of ROECOC

Table 2 Characteristics of the UCI datasets (Features: C-continuous, B-binary, and N-nominal)

	Dataset	Cases	Classes	Features		
				C	B	N
(a)	Balance	625	3	4	–	–
(b)	Ecoli	336	8	7	–	1
(c)	Glass	214	6	9	–	–
(d)	Iris	150	3	4	–	–
(e)	Satimag	6435	6	36	–	–
(f)	Segment	2310	7	19	–	–
(g)	vehicle	846	4	18	–	–
(h)	Vowel	990	11	10	–	–
(i)	Wine	178	3	13	–	–
(j)	Yeast	1484	10	8	–	–
(k)	Zoo	101	7	1	15	–
(l)	letter	1214	26	16	–	–
(m)	Page-blocks	5473	5	10	–	–
(n)	shuttle	14,500	7	9	–	–
(o)	soybean	306	18	35	–	–
(p)	thyroid	215	3	5	–	–

base classifiers provide the prior knowledge of classes ignored. It is worth noting that zeros denoted as rejection may still appear in the new matrix, which make the output flexible. However, when a classifier is non-

Table 1 The algorithm of the re-encoding ROECOC using the reject option

Input: initial matrix $M_{init} \in \{-1, 0, +1\}^{K \times L}$, training set \mathfrak{S} and cost matrix C_r
Output: classification accuracy
Step 1. Extract samples from \mathfrak{S} to form the validation set $\mathfrak{S}_{validate}$, update the training set $\mathfrak{S}' = \mathfrak{S} - \mathfrak{S}_{validate}$, set $M = M_{init}$.
Step 2. Train L based classifiers l_i $i = 1, 2, \dots, L$ on the basis of M and \mathfrak{S}' , use l_i to classify $\mathfrak{S}_{validate}$, and save the classification results.
Step 3. Draw the ROCs for each base classifier, construct the thresholds (t_1, t_2) based on equation (6) and obtain the corresponding optimal classifier f_i $i = 1, 2, \dots, L$ and reject option.
Step 4. Use f_i to classify the classes denoted by zero in M , and recode them to get the new matrix M_{new} .
Step 5. Use f_i to classify the testing sample, and decode the output by the modified Hamming distance decoding strategy.

competent for an instance to be classified, the output should be zero, which exactly matches the corresponding code in the original matrix.

4 Experiments

4.1 Experimental data and design

To validate the performance of our proposal, we use two kinds of dataset: 16 different multi-class datasets from the University of California at Irvine (UCI) repository [32, 33] and high resolution range profile (HRRP) dataset of three airplanes: B-52, Farmer and Fishbed. Table 2 provides the characteristics of the UCI datasets [19]. Meanwhile, the principal component analysis (PCA) is used to reduce the dimensionality to promote classification speed. The HRRP dataset was acquired with zoom models in a microwave anechoic chamber and it was composed of data in the range of 0°-155°. There are 322 location data for B-52, 311 for Farmer and 451 for Fishbed. Each data sample is described by 64 attributes, namely range cell.

4.2 Experimental design

The experiments chose six different coding matrices with different freedom degree (the distribution density of zero): one-versus-all coding, dense random, Discriminant

ECOC, sparse random, SA-ECOC [20] and one-versus-one coding. As for ternary ECOC, we add ReECOC [18] to compare the results with ROECOC. The random matrices were selected from a set of 2000 randomly generated matrices with $P(-1) = P(+1) = 0.5$ for the dense random matrix and $P(-1) = P(+1) = P(0) = 1/3$ for the sparse random matrix [19]. In all algorithms, the parameters are predefined or default values given by the authors. The LOGLC and SVM with polynomial kernel $K(x, x_i) = [x, x_i + 1]^q$ are considered as base classifiers. The regularization parameter C and the kernel parameter q are selected by K -fold cross-validation ($K = 5$). The range of values allowed for q is 1-10, the initial value of C is 2.

The cost matrix is set as $C_r = \begin{bmatrix} 0 & 6 & 1 \\ 3 & 0 & 1 \end{bmatrix}$ by manual, which is the same for each base classifier and will not affect the feasibility of the experiment.

Furthermore, the ROECOC is used for target recognition with HRRP dataset of three different planes. We pick four different angle ranges to evaluate the performance in practice (0°-20°, 20°-40°, 130°-150°, 0°-150°). To simplify the experiments, the SVM was used for the based classifiers with the same parameters as the mentioned before. The decoding strategy is the modified Hamming distance strategy. At last, we discuss the influence of freedom degree on cost-loss classification.

To evaluate the performance of the different results, the experiments perform stratified ten-fold cross validation and test for a confidence interval of 95% with a two-tailed t -test

Table 3 Accuracy rates and confidence interval at 95% for ECOC matrices using SVM(%)

		(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
OVA	<i>classic</i>	85.87 ± 1.46	68.75 ± 6.15	59.82 ± 2.27	74.67 ± 2.77	62.00 ± 8.19	63.72 ± 5.63	67.61 ± 1.73	87.68 ± 8.24
	<i>ROECOC</i>	95.96 ± 6.23	72.80 ± 3.10	63.87 ± 1.58	80.71 ± 2.99	68.24 ± 0.82	73.94 ± 14.30	73.52 ± 3.80	89.49 ± 3.70
dense	<i>classic</i>	90.24 ± 2.88	42.56 ± 0.56	76.02 ± 4.07	74.00 ± 4.11	67.80 ± 6.54	64.29 ± 8.17	53.52 ± 0.80	82.12 ± 7.97
	<i>ROECOC</i>	92.47 ± 3.61	59.64 ± 1.38	80.80 ± 4.62	80.80 ± 2.46	76.46 ± 1.27	77.36 ± 2.44	68.35 ± 2.38	85.47 ± 1.25
DECOC	<i>classic</i>	92.16 ± 3.12	69.49 ± 2.80	71.45 ± 3.08	94.07 ± 3.85	80.71 ± 4.15	77.57 ± 1.27	74.26 ± 2.45	96.03 ± 1.03
	<i>ReECOC</i>	93.16 ± 2.26	70.89 ± 3.04	80.80 ± 4.60	97.33 ± 2.14	82.03 ± 2.13	80.42 ± 3.55	76.96 ± 2.37	96.70 ± 3.66
	<i>ROECOC</i>	94.18 ± 2.05	71.14 ± 1.78	79.04 ± 3.87	97.42 ± 1.12	83.36 ± 1.09	80.95 ± 1.56	76.85 ± 3.32	97.87 ± 1.93
sparse	<i>classic</i>	96.28 ± 6.04	69.64 ± 9.67	80.83 ± 10.49	65.67 ± 5.41	72.80 ± 3.10	82.34 ± 2.86	62.88 ± 5.28	79.80 ± 2.48
	<i>ReECOC</i>	95.36 ± 7.14	70.79 ± 8.54	87.95 ± 8.39	74.67 ± 8.47	71.09 ± 13.59	83.85 ± 14.85	69.57 ± 10.23	80.03 ± 1.03
	<i>ROECOC</i>	96.32 ± 6.02	70.67 ± 11.81	88.30 ± 8.72	79.33 ± 3.85	71.63 ± 1.98	87.14 ± 11.55	70.73 ± 3.54	79.09 ± 0.00
SAECOC	<i>classic</i>	95.81 ± 3.10	66.20 ± 3.11	86.42 ± 1.32	95.06 ± 3.81	79.96 ± 2.58	83.64 ± 3.46	74.94 ± 9.76	87.57 ± 4.16
	<i>ReECOC</i>	94.36 ± 4.14	68.72 ± 2.34	85.07 ± 3.85	96.00 ± 3.70	81.89 ± 1.87	86.96 ± 1.40	75.53 ± 12.78	88.74 ± 3.60
	<i>ROECOC</i>	94.19 ± 7.38	70.21 ± 4.02	88.74 ± 6.30	97.23 ± 1.24	81.03 ± 2.08	92.16 ± 1.78	78.09 ± 6.06	92.17 ± 3.50
OVO	<i>classic</i>	95.52 ± 8.22	74.93 ± 14.60	86.70 ± 18.60	96.68 ± 5.38	81.54 ± 3.03	93.76 ± 3.85	76.92 ± 1.12	96.26 ± 6.42
	<i>ReECOC</i>	95.68 ± 6.01	75.34 ± 2.68	88.23 ± 24.65	96.67 ± 8.47	80.40 ± 4.59	93.50 ± 9.90	75.65 ± 17.29	98.58 ± 1.36
	<i>ROECOC</i>	96.00 ± 2.11	75.88 ± 5.86	89.69 ± 17.26	97.33 ± 1.94	82.59 ± 12.39	93.68 ± 9.35	76.85 ± 2.38	98.71 ± 2.89

Table 4 Accuracy rates and confidence interval at 95% for ECOC matrices using LOGLC(%)

		(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
OVA	classic	86.80 ± 4.20	65.77 ± 6.77	63.33 ± 2.76	64.00 ± 2.63	77.73 ± 0.60	69.12 ± 8.66	75.65 ± 12.75	69.08 ± 2.95
	ROECOC	96.96 ± 10.23	69.36 ± 4.01	76.91 ± 3.11	74.00 ± 2.63	79.39 ± 3.43	73.12 ± 8.66	78.40 ± 3.59	78.08 ± 2.95
dense	classic	86.89 ± 1.98	73.21 ± 5.12	59.72 ± 7.77	74.00 ± 2.53	63.82 ± 0.06	68.96 ± 3.72	48.41 ± 2.45	79.89 ± 1.50
	ROECOC	90.56 ± 4.89	74.50 ± 1.56	66.79 ± 4.07	81.29 ± 4.57	65.78 ± 2.78	74.29 ± 14.29	70.56 ± 3.23	7.89 ± 1.60
DECOC	classic	85.87 ± 2.47	64.48 ± 6.34	75.38 ± 2.37	92.50 ± 2.18	77.05 ± 1.32	76.34 ± 1.14	68.90 ± 5.83	74.94 ± 4.53
	ReECOC	89.43 ± 2.15	66.53 ± 2.34	78.49 ± 4.32	95.12 ± 8.12	82.01 ± 2.69	78.00 ± 1.49	77.01 ± 2.43	77.84 ± 1.49
	ROECOC	90.43 ± 6.18	67.45 ± 4.23	82.19 ± 2.19	95.14 ± 7.49	85.45 ± 4.78	79.13 ± 2.78	79.45 ± 4.31	82.68 ± 3.23
sparse	classic	86.08 ± 0.94	73.21 ± 5.12	68.27 ± 2.89	74.66 ± 5.08	79.78 ± 0.71	71.77 ± 3.30	71.76 ± 4.27	75.56 ± 5.05
	ReECOC	87.04 ± 13.15	84.23 ± 2.13	69.62 ± 1.28	75.67 ± 4.23	80.20 ± 2.07	74.16 ± 2.80	68.56 ± 17.08	78.28 ± 2.84
	ROECOC	89.76 ± 17.29	87.69 ± 4.10	70.90 ± 3.70	76.33 ± 16.94	80.26 ± 2.25	74.50 ± 3.52	69.86 ± 3.60	78.84 ± 3.85
SAECOC	classic	88.28 ± 3.11	68.33 ± 3.25	79.78 ± 2.52	92.50 ± 2.18	59.61 ± 3.28	76.83 ± 2.78	79.61 ± 3.34	80.10 ± 2.65
	ReECOC	90.13 ± 4.12	69.04 ± 2.85	78.06 ± 1.31	94.04 ± 5.95	59.52 ± 1.62	78.83 ± 4.31	80.91 ± 2.15	84.04 ± 1.50
	ROECOC	91.02 ± 1.23	71.64 ± 5.12	82.96 ± 5.01	96.67 ± 2.81	68.05 ± 1.74	79.97 ± 3.25	82.68 ± 3.09	86.68 ± 4.32
OVO	classic	88.48 ± 4.30	88.41 ± 2.27	88.90 ± 8.04	94.00 ± 8.47	86.53 ± 4.07	89.07 ± 3.05	76.31 ± 3.21	83.17 ± 5.13
	ReECOC	90.04 ± 3.88	87.03 ± 2.13	87.79 ± 3.85	96.00 ± 16.94	86.15 ± 0.51	88.03 ± 11.00	77.42 ± 3.83	83.93 ± 17.97
	ROECOC	91.72 ± 12.01	89.23 ± 2.31	89.70 ± 14.32	97.33 ± 16.94	86.76 ± 6.79	92.03 ± 13.35	78.01 ± 3.66	89.33 ± 7.70

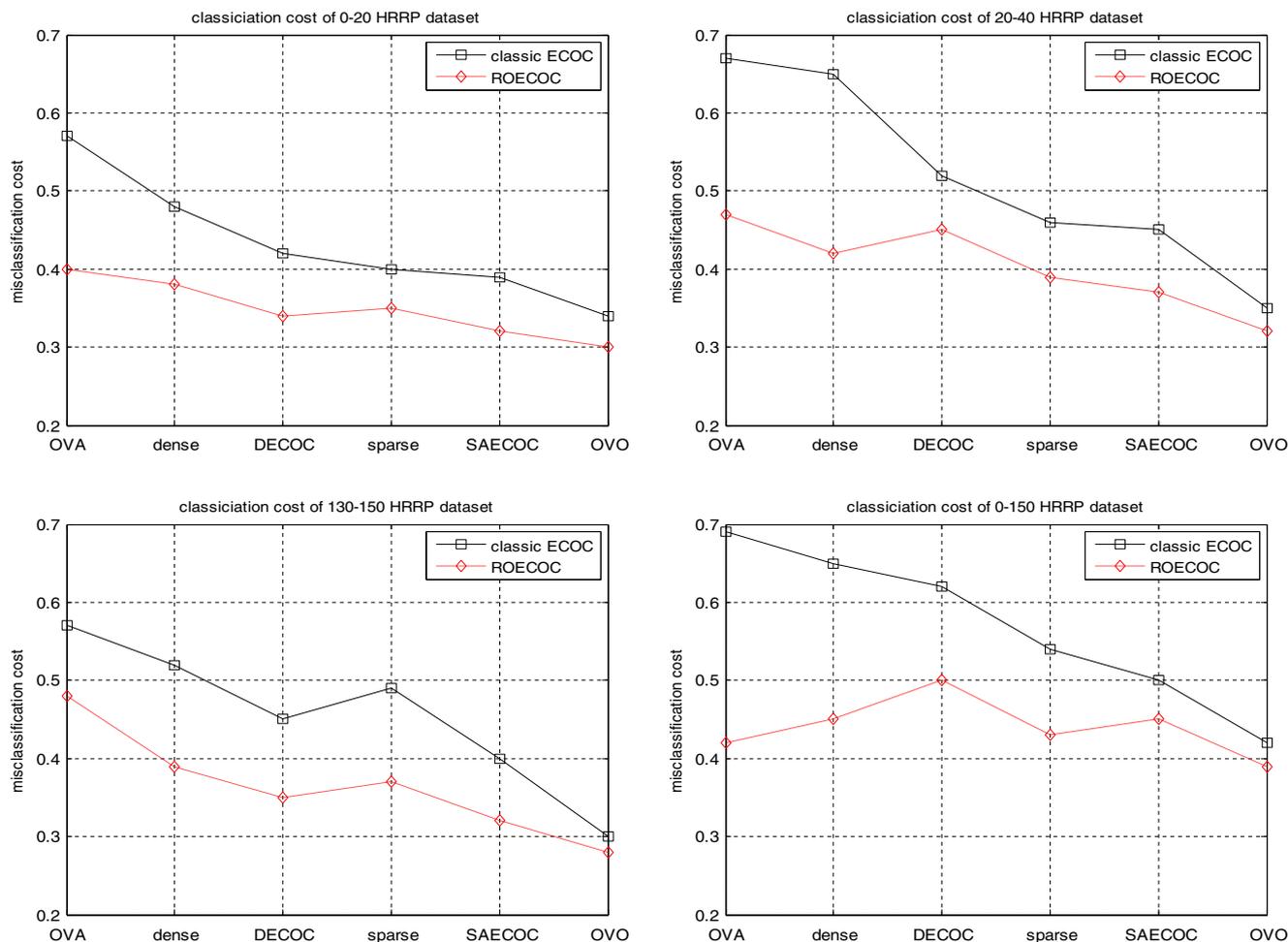


Fig. 7 Normalized classification cost of four different angle ranges of HRRP datasets

if the number of samples was larger than 500 [24, 34]. The calculating formula is given as follows:

$$\frac{|\bar{x}-\mu|}{\sigma/\sqrt{n}} \geq t_{0.025}(n-1) \quad (7)$$

where μ and σ indicate mean and variance respectively and $t_{0.025}(9) = 2.2622$.

4.3 Experimental results and analysis

4.3.1 UCI dataset

Tables 3 and 4 show the classification accuracy for different encoding matrices based on different classifiers and the modified Hamming decoding. The best performance per dataset is highlighted in boldface. The results of the rest datasets are listed in Table 5 and Table 6 as shown in the Appendix.. From the results in tables, we can see that the classification accuracy got by ROECOC outperform the corresponding state-of-the-art matrices and Re-coding methods most of the time. This illustrates that the classification performance of ROECOC based on reject option is much better than that of classic ECOCs without reject option. It is worth noting that the classification results have no distinct difference when using one-versus-one code as the initial matrix. The most likely reason is that there is only one class in each binary split in one-versus-one matrix and the class imbalance problem has little impact on classification. One of the advantages of ROECOC is to overcome the class imbalance problem according to the introduction section, which can be detected easily with the one-versus-all as the initial matrix. In general, the ROECOC based on reject option has better performance and can avoid making the direct decision for samples to be easily misclassified or with high misclassification risk, which can be seen from other ECOC matrices.

4.3.2 HRRP dataset

Figure 7 show the classification cost of four different angle ranges of HRRP dataset based on classic ECOCs and ROECOC. The freedom degree of matrices along the x-axes increases from left to right. According to the results, the classification cost of ROECOC is much less than that of classic ECOCs in general. We can also find that with the increasing of the freedom degree, the

classification performance of classic ECOCs and ROECOC are approaching gradually. When the freedom degree increases to a certain extent, the advantage diminishes. Especially when using one-versus-one code as the initial matrix, the classification performance has no big difference, which is in accordance with the results of UCI datasets. The most likely reason is that with the increasing of the freedom degree, the degree of class imbalance decreases. The more balanced the data distribution, the less samples rejected. However, it is worthy of considering the freedom degree when using ROECOC to classify (Tables 5 and 6).

It is also worth noting that the classification cost is less when the angle range is smaller such as 0° - 20° , 20° - 40° and 130° - 150° . Because the recognition based on HRRP has orientation sensitivity. Taking the results and analysis above into consideration, the proposed ROECOC can promote the classification performance aiming at reducing the classification risk in practice.

5 Conclusions

The multi-class classification aiming at reducing classification cost has been widely used in practice. However, the classic ECOC classification still takes the error as the evaluation criteria, which is not suitable for the cost-loss classification. Meanwhile, the dichotomizers can only produce binary outputs and have no capability for rejection. To reduce classification cost and construct data-driven matrix, a new reject option-based ECOC is proposed. ROECOC does not change the framework of ECOC classification and introduces reject option for each base classifier. The reject option based on ROC curve is constructed by minimizing the cost-loss function model with the help of cost matrix and ROCCH. The dichotomizers with reject option can produce three-symbol output and classify selectively, which can provide more information of class distribution. Given any initial matrix, ROECOC can produce a data-driven and competent matrix for the cost-sensitive classification. The experimental results illustrate that ROECOC can reduce the classification cost and enhance the performance especially when the freedom degree of the initial matrix is low.

Acknowledgments This work is supported by the National Science Foundation of China under grant 61806219, 61703426 and 61876189, and by Young Talent fund of University and Association for Science and Technology in Shaanxi, China (20190108).

Appendix

Table 5 Accuracy rates and confidence interval at 95% for ECOC matrices using SVM(%)

		(i)	(j)	(k)	(l)	(m)	(n)	(o)	(p)
OVA	<i>classic</i>	68.54 ± 0.90	23.86 ± 3.25	67.06 ± 2.56	67.17 ± 0.71	94.58 ± 9.02	81.70 ± 0.69	72.52 ± 3.71	91.62 ± 12.31
	<i>ROECOC</i>	74.71 ± 10.75	44.68 ± 3.46	75.33 ± 2.71	74.93 ± 0.87	96.80 ± 9.40	89.19 ± 2.22	77.98 ± 3.54	92.59 ± 8.87
dense	<i>classic</i>	71.13 ± 2.34	29.84 ± 5.10	46.44 ± 2.84	69.30 ± 0.48	95.07 ± 0.52	79.96 ± 0.65	73.59 ± 3.44	83.72 ± 6.87
	<i>ROECOC</i>	84.75 ± 1.90	56.40 ± 2.05	50.66 ± 3.49	79.12 ± 0.53	99.36 ± 9.92	88.04 ± 0.89	78.32 ± 2.85	92.57 ± 7.41
DECOC	<i>classic</i>	83.70 ± 3.68	32.02 ± 4.43	58.38 ± 7.54	59.61 ± 1.39	92.16 ± 8.95	83.61 ± 3.16	72.56 ± 1.62	92.07 ± 1.22
	<i>ReECOC</i>	84.38 ± 3.51	45.96 ± 2.46	69.36 ± 10.4	65.58 ± 1.36	95.89 ± 8.28	87.42 ± 1.64	78.32 ± 2.87	93.50 ± 3.06
	<i>ROECOC</i>	83.72 ± 3.68	57.40 ± 3.85	76.96 ± 7.34	65.54 ± 1.51	96.81 ± 8.87	89.12 ± 2.15	79.78 ± 1.59	93.75 ± 2.61
sparse	<i>classic</i>	71.95 ± 4.01	47.04 ± 2.93	49.94 ± 6.17	64.18 ± 2.28	90.24 ± 7.25	87.16 ± 0.87	72.90 ± 4.54	87.44 ± 6.65
	<i>ReECOC</i>	78.89 ± 14.12	55.05 ± 3.20	52.97 ± 8.41	65.03 ± 2.30	94.83 ± 0.84	90.17 ± 1.29	77.35 ± 4.88	88.75 ± 4.78
	<i>ROECOC</i>	79.18 ± 3.86	56.55 ± 5.91	53.16 ± 22.10	66.78 ± 2.37	94.84 ± 7.86	91.14 ± 0.57	79.51 ± 3.82	89.10 ± 17.26
SAECOC	<i>classic</i>	79.46 ± 5.45	52.09 ± 3.12	68.27 ± 3.10	69.61 ± 0.99	94.32 ± 3.42	91.18 ± 0.78	55.59 ± 4.03	86.03 ± 3.05
	<i>ReECOC</i>	83.89 ± 3.06	58.53 ± 5.73	76.25 ± 7.25	72.72 ± 0.47	95.75 ± 4.88	92.55 ± 1.29	72.56 ± 3.72	92.07 ± 2.22
	<i>ROECOC</i>	85.58 ± 2.84	59.43 ± 3.61	78.09 ± 6.06	76.74 ± 0.36	96.66 ± 4.80	93.52 ± 1.77	76.31 ± 2.83	93.50 ± 1.22
OVO	<i>classic</i>	85.83 ± 6.19	58.43 ± 3.61	56.33 ± 5.67	76.69 ± 0.81	97.10 ± 2.98	91.86 ± 0.95	73.87 ± 4.94	87.93 ± 5.89
	<i>ReECOC</i>	86.81 ± 3.95	60.68 ± 3.46	65.73 ± 2.01	78.94 ± 0.97	97.30 ± 2.06	93.46 ± 1.06	77.67 ± 4.21	92.09 ± 6.38
	<i>ROECOC</i>	87.35 ± 2.19	61.33 ± 2.00	66.61 ± 14.90	80.12 ± 0.52	98.09 ± 2.80	94.66 ± 1.69	79.32 ± 4.68	92.50 ± 5.32

Table 6 Accuracy rates and confidence interval at 95% for ECOC matrices using LOGLC(%)

		(i)	(j)	(k)	(l)	(m)	(n)	(o)	(p)
OVA	<i>classic</i>	68.56 ± 7.71	31.33 ± 2.00	57.82 ± 3.23	52.14 ± 0.27	87.70 ± 10.35	80.60 ± 1.52	67.27 ± 2.76	94.40 ± 4.35
	<i>ROECOC</i>	75.12 ± 5.31	43.10 ± 5.19	79.19 ± 1.78	67.24 ± 1.39	95.24 ± 13.64	89.64 ± 1.25	80.20 ± 2.63	95.83 ± 4.12
dense	<i>classic</i>	62.22 ± 5.69	45.35 ± 1.17	48.44 ± 0.84	65.55 ± 0.38	89.38 ± 1.13	83.93 ± 2.38	65.94 ± 2.96	93.97 ± 4.01
	<i>ROECOC</i>	69.54 ± 4.11	53.82 ± 5.68	66.10 ± 2.31	69.63 ± 0.87	93.80 ± 1.34	90.82 ± 1.24	79.92 ± 4.03	94.93 ± 3.86
DECOC	<i>classic</i>	68.23 ± 2.82	48.92 ± 2.42	68.90 ± 5.83	61.29 ± 1.24	91.37 ± 7.18	83.58 ± 1.16	65.62 ± 3.34	85.79 ± 1.44
	<i>ReECOC</i>	73.16 ± 4.26	51.15 ± 4.53	73.45 ± 2.60	67.33 ± 1.03	93.92 ± 5.72	87.31 ± 3.04	78.76 ± 3.06	93.00 ± 1.57
	<i>ROECOC</i>	74.90 ± 3.14	53.40 ± 8.26	75.12 ± 1.20	67.04 ± 1.05	95.25 ± 6.10	89.54 ± 3.18	80.24 ± 3.12	94.50 ± 1.22
sparse	<i>classic</i>	69.91 ± 1.07	51.30 ± 3.08	52.49 ± 6.28	65.82 ± 0.95	94.39 ± 13.48	86.69 ± 1.38	68.23 ± 3.34	94.41 ± 12.15
	<i>ReECOC</i>	72.95 ± 1.31	55.53 ± 2.34	64.43 ± 2.11	69.23 ± 0.40	92.16 ± 14.45	89.22 ± 1.13	78.95 ± 2.19	93.96 ± 5.55
	<i>ROECOC</i>	73.12 ± 1.23	57.40 ± 3.85	64.55 ± 13.86	70.33 ± 1.06	95.48 ± 29.20	91.91 ± 1.40	82.17 ± 3.18	92.10 ± 17.26
SAECOC	<i>classic</i>	70.22 ± 1.78	55.54 ± 2.71	63.03 ± 7.64	68.49 ± 3.02	94.31 ± 3.64	88.47 ± 0.95	70.81 ± 3.55	93.50 ± 3.05
	<i>ReECOC</i>	73.34 ± 1.57	57.19 ± 2.15	66.10 ± 2.31	73.63 ± 0.40	96.22 ± 2.78	92.53 ± 0.84	80.89 ± 3.34	93.46 ± 4.18
	<i>ROECOC</i>	75.20 ± 1.23	58.23 ± 5.68	67.36 ± 2.09	73.84 ± 1.25	97.27 ± 3.81	92.32 ± 0.76	81.21 ± 2.60	94.52 ± 4.03
OVO	<i>classic</i>	72.07 ± 3.85	56.71 ± 3.85	72.48 ± 2.08	75.40 ± 1.87	94.12 ± 13.16	93.86 ± 0.91	75.31 ± 4.64	95.82 ± 5.65
	<i>ReECOC</i>	73.19 ± 1.94	57.02 ± 3.69	74.17 ± 3.58	75.39 ± 2.39	95.05 ± 12.98	94.27 ± 0.79	80.22 ± 4.63	95.81 ± 17.98
	<i>ROECOC</i>	74.00 ± 1.86	59.01 ± 1.49	75.89 ± 3.33	78.96 ± 0.33	96.06 ± 7.70	95.90 ± 0.78	81.01 ± 3.45	96.27 ± 12.04

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Song Y, Wang X, Lei L, Xue A (2015) A novel similarity measure on intuitionistic fuzzy sets with its applications. *Appl Intell* 42:252–261. <https://doi.org/10.1007/s10489-014-0596-z>
- Wang X, Song Y (2018) Uncertainty measure in evidence theory with its applications. *Appl Intell* 48:1672–1688. <https://doi.org/10.1007/s10489-017-1024-y>
- Song Y, Wang X, Quan W, Huang W (2019) A new approach to construct similarity measures for intuitionistic fuzzy sets. *Soft Comput* 23:1985–1998. <https://doi.org/10.1007/s00500-017-2912-0>
- Song Y, Wang X, Zhu J, Lei L (2018) Sensor dynamic reliability evaluation based on evidence and intuitionistic fuzzy sets. *Appl Intell* 48:3950–3962. <https://doi.org/10.1007/s10489-018-1188-0s>
- Song Y, Fu Q, Wang Y-F, Wang X (2019) Divergence-based cross entropy and uncertainty measures of Atanassov's intuitionistic fuzzy sets with their application in decision making. *Appl Soft Comput* 84(105):703. <https://doi.org/10.1016/j.asoc.2019.105703>
- T. G. Dietterich and G. Bakiri. Solving. Multi-class learning problems via error-correcting output codes. *J Artif Intell Res*, 1995, 34(2):263–286.
- Olusegun P, Anastasios D et al (2018) Spatio-temporal spectrum sensing in cognitive radio networks using beamformer-aided SVM algorithms. *IEEE Access* 99(6):25377–25,388
- Ma C, Tsang IW, Shen F et al (2019) Error correcting input and output hashing. *IEEE Transactions on Cybernetics* 49(3):781–791
- Zhao KK, Matsukawa T, Suzuki E (2019) Experimental validation for N-ary error correcting output codes for ensemble learning of deep neural networks. *J Intell Inf Syst* 52(2):367–392
- Liu KH, Zeng ZH, Vincent TY (2016) A hierarchical ensemble of ECOC for cancer classification based on multi-class microarray data. *Inf Sci* 349(1):102–118
- Gu XD, Deng F, Gao X, Zhou R (2018) An improved sensor fault diagnosis scheme based on TA-LSSVM and ECOC-SVM. *J Syst Sci Complex* 31:372–384
- Crammer K, Singer Y (2000) On the learnability and design of output codes for multi-class problems. In *Proc. 13th Annual Conference on Computational Learning Theory*, Kluwer Academic Publishers, Boston, pp. 896–909
- Pujol O, Radeva P, Vitria J (2006) Discriminate ECOC: A heuristic method for application dependent design of error correcting output codes. *IEEE Trans on Pattern Analysis and Machine Intelligence* 28(6):1001–1007
- Escalera S, Tax DMJ, Pujol O, Radeva P, Duin R (2008) Subclass Problem-Dependent Design for Error-Correcting Output Codes. *IEEE Trans on Pattern Analysis and Machine Intelligence* 30(6):1041–1054
- Bouzas D, Arvanitopoulos N, Tefas A (2011) Optimizing Linear Discriminant Error Correcting Output Codes Using Particle Swarm Optimization. *Lect Notes Comput Sci* 6792(4):79–86
- Lei L, Wang XD, Luo X, Song YF (2016) Hierarchical Error-correcting Output Codes based on SVDD. *Pattern Anal Applic* 19(1):163–171
- Wang Y, Chen S, Xue H (2012) Can under-exploited structure of original-classes help ECOC-based multi-class classification. *Eurocomputing* 89(complete):158–167
- Escalera S, Pujol O (2013) Recoding ECOCs without Retraining. *Pattern Recogn Lett* 31(5):555–562
- Zhou JD, Yang Y, Zhang MJ, Xing HB, Xing H (2016) Constructing ECOC based on Confusion Matrix for Multi-class Learning Problems. *Science China Inf Sci* 51(1):012107:1
- Zhong TY, Liu KH, Wang BZ (2017) Multiclass microarray data classification based on SA-ECOC. 10th International Symposium on Computational Intelligence and Design, IEEE
- Rocha A, Goldenstein SK (2014) Multiclass from Binary: Expanding one-versus-all, one-versus-one and ECOC-based approaches. *IEEE Transactions on Neural Networks and Learning Systems* 25(2):289–302
- Sum MX, Liu KH, Hong QQ, Wang BZ (2018) A new ECOC algorithm for multiclass microarray data classification. 24th International Conference on Pattern Recognition, Beijing
- Marie L, Hegarat-Masclé SL, Aldea E (2018) Evidential framework for Error Correcting Output Codes classification. *Eng Appl Artif Intell* 73:10–21
- Galar M, Fernandez A, Barrenechea E (2015) DRCW-OVO: Distance-based relative competence weighting combination for one-vs-one strategy in multi-class problems. *Pattern Recogn* 48(1):28–42
- Ohsaki M, Wang P, Matsuda K (2017) Confusion-matrix-based kernel logistic regression for imbalanced data classification. *IEEE Trans Knowl Data Eng* 99(3):1–10
- Allwein E, Schapire R, Singer Y (2000) Reducing multiclass to binary: A unifying approach for margin classifiers. In *Machine Learning: Proceedings of the Seventeenth International Conference*, pp. 1545–1550
- Tortorella F (2005) A roc-based reject rule for dichotomizers. *Pattern Recogn Lett* 26(2):167–180
- Benard S, Chatelain C, Adam S, Saborin R (2016) The multi-class roc front method for cost-sensitive classification. *Pattern Recogn* 46:46–60
- Zhao ZC, Wang XD (2018) A minimum risk recognition method of ballistic targets with rejection options. *J Xi'an Jiaotong Univ* 52(4):132–138
- Pietraszek T (2007) On the use of ROC analysis for the optimization of abstaining classifiers. *Mach Learn* 68(5):137–169
- Escalera S, Pujol O, Radeva P (2010) On the Decoding Process in Ternary Error-Correcting Output Codes. *IEEE Trans Pattern Analysis and Machine Intelligence* 32(1):120–134
- Passerini A, Pontil M, Frasconi P (2004) New Results on Error Correcting Output Codes of Kernel Machines. *IEEE Trans Neural Netw* 15(1):45–54
- Bi JJ, Zhang CS (2018) An empirical comparison on state-of-the-art multi-class imbalance learning algorithms and a new diversified ensemble learning scheme. *Knowl-Based Syst* 158(5):81–93
- Demsar J (2006) Statistical Comparisons of Classifiers over Multiple Data Sets. *Machine Learning Research* 35(7):1–30

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Lei Lei received the Ph.D. degree in computer science from Air Force Engineering University (AFEU) in 2017.

She is currently a Lecturer at the College of Information and Navigation, AFEU. She has contributed more than 20 papers to professional journals. Her research interests include pattern recognition, machine learning and deep learning.



Xi Luo was born in Xi'an, Shaanxi, China in 1988. He received his Master degree in 2012 from Air Force Engineering University. He is now a lecturer at the College of Information and Navigation College, Air Force Engineering University. His research interest covers intelligent information processing, data mining and network security.



Yafei Song received the Ph.D. degree in computer science from Air Force Engineering University (AFEU), China, in 2015.

He is currently a Lecturer at the College of Air and Missile Defense, AFEU. He also worked as a post-doctoral researcher in AFEU since 2017. He is the author or coauthor of approximately 30 original articles. His research interests include pattern recognition, intelligent information processing, fuzzy set theory, and evidential reasoning.