Pattern Recognition Letters 31 (2010) 555–562

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

Re-coding ECOCs without re-training

Sergio Escalera *, Oriol Pujol, Petia Radeva

Centre de Visió per Computador, Campus UAB, Edifici O, Bellaterra, 08193 Barcelona, Spain
Dept. Matemàtica Aplicada i Anàlisi, Universitat de Barcelona, Gran Via 585, 08007 Barcelona, Spain

ARTICLE INFO

Article history:
Received 17 March 2009
Received in revised form 26 November 2009
Available online 13 January 2010

Communicated by R.P.W. Duin

Keywords:
Multi-class classification
Error-Correcting Output Codes
Coding
Decoding

ABSTRACT

A standard way to deal with multi-class categorization problems is by the combination of binary classifiers in a pairwise voting procedure. Recently, this classical approach has been formalized in the Error-Correcting Output Codes (ECOC) framework. In the ECOC framework, the one-versus-one coding demonstrates to achieve higher performance than the rest of coding designs. The binary problems that we train in the one-versus-one strategy are significantly smaller than in the rest of designs, and usually easier to be learnt, taking into account the smaller overlapping between classes. However, a high percentage of the positions coded by zero of the coding matrix, which implies a high sparseness degree, does not codify meta-class membership information. In this paper, we show that using the training data we can redefine without re-training, in a problem-dependent way, the one-versus-one coding matrix so that the new coded information helps the system to increase its generalization capability. Moreover, the new re-coding strategy is generalized to be applied over any binary code. The results over several UCI Machine Learning repository data sets and two real multi-class problems show that performance improvements can be obtained re-coding the classical one-versus-one and Sparse random designs compared to different state-of-the-art ECOC configurations.

1. Introduction

Recently, robust binary classifiers have been proposed in the bibliography with high performance, such as Support Vector Machines, Neural Networks, and Adaboost (Friedman et al., 1998). However, the extension of many binary classifiers to the multi-class case, where N possible classes appear, is a hard task. In this sense, a common strategy consists of defining a set of binary problems, which are combined by means of a voting procedure. In this sense, Error-Correcting Output Codes were designed as an alternative way of combining binary problems in order to deal with the multi-class case (Dietterich and Bakiri, 1995).

The ECOC framework is based on two main steps. At the first step, namely coding, a set of binary problems (dichotomizers) are defined based on the learning of different sub-partitions of classes by means of a base classifier. Then, each of the partitions is embedded as a column of a coding matrix Q. The rows of Q correspond to the codewords codifying each class. At the second step, namely decoding, a new data sample that arrives to the system is tested, and a codeword formed as a result of the output of the binary problems is obtained. This test codeword is compared with each class codeword based on a given decoding measure, and a classification prediction is obtained for the new object. Unlike the voting procedure, the information provided by the ECOC dichotomizers is shared among classes in order to obtain a precise classification decision, being able to reduce either the variance and the bias produced by the learners (Kong and Dietterich, 1995).

When Dietterich et. al. defined the binary ECOC framework in (Dietterich and Bakiri, 1995), all positions from the coding matrix Q belonged to the set $\{+1, -1\}$. This makes all classes to be considered by each dichotomizer as a member of one of the two possible partitions of classes that define each binary problem. In this case, the one-versus-all and dense random ECOC approaches were defined (Dietterich and Bakiri, 1995). Afterwards, Allwein et al. (2002) defined the ternary ECOC, where the positions of the coding matrix Q can be either +1, -1 or 0, and the sparse random and one-versus-one (pairwise voting) designs could be defined in the ECOC framework. In this case, the zero symbol means that a given class is not considered in the learning process of a particular dichotomizer. The huge set of possible bi-partitions of classes from this ternary ECOC framework has recently suggested the use of problem-dependent designs as well as new decoding strategies (Escalera et al., 2008a,b; Pujol et al., 2006).

Concerning the one-versus-one ECOC strategy, it codifies the splitting of each possible pair of classes as a dichotomizer, which results in $N(N-1)/2$ binary problems for an N-class problem. This number is usually larger in comparison with the linear tendency of the rest of ECOC designs. Although this suggests larger training

* Corresponding author. Tel.: +34 687291957; fax: +34 93 581 1670.
E-mail addresses: sergio@maia.ub.edu (S. Escalera), oriol@maia.ub.edu (O. Pujol), petia@maia.ub.edu (P. Radeva).

0167-8655/$ - see front matter © 2009 Published by Elsevier B.V.
doi:10.1016/j.patrec.2009.12.002

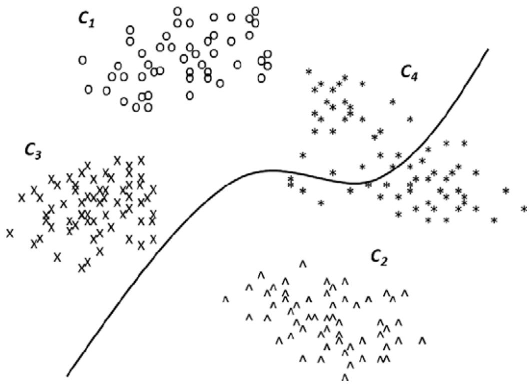

Fig. 1. Decision boundary for a binary problem C_1 against C_2.

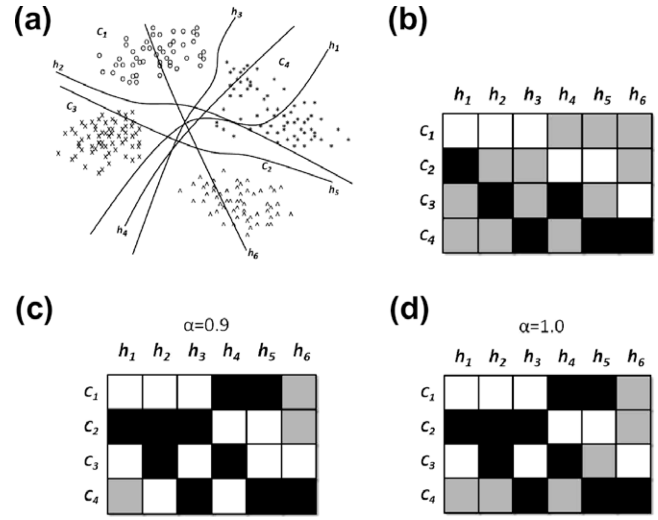

Fig. 2. ECOC codification for a 4-class problem: (a) Non-linear decision boundaries for the 4-class problem, (b) initial one-versus-one ECOC codification, (c) RE-ECOC codification with $\alpha = 0.9$, and (d) RE-ECOC codification with $\alpha = 1.0$.

times, the individual problems that we need to train on are significantly smaller. Thus, it is actually possible to save time. Moreover, the problems to be learnt are usually easier, since the classes have less overlapping. Hence, the one-versus-one ECOC design tends to obtain better results than the rest of ECOC designs in real multi-class problems (Escalera et al., 2008a,b).

In this paper, we focus on the one-versus-one coding matrix design, though the new methodology can be applied to any ternary-based coding strategy. The idea is to look for a better coding of the matrix without re-training the classifiers involved. Training data are used in a problem-dependent way for updating the zero positions to +1 or -1 symbols if a higher classification performance can be achieved. Observe the 4-classes problem as shown in Fig. 1. A decision boundary of a non-linear classifier has been obtained in the learning process of the dichotomizer that splits classes c_1 and c_2. The point of this article is that without the necessity of re-training the classifier, the same decision boundary can be used to give a prediction hypothesis about class c_3. On the other hand, note that the use of this decision boundary to classify class c_4 may result in a random decision function. Using this information, we re-code the classical problem-independent one-versus-one into a problem-dependent one-versus-one design extending the trained classifier on new classes for the binary classifier for which the dichotomizer is relevant. The design is possible thanks to a new weighting procedure that takes into account the performance of the dichotimizers at the decoding step (Escalera et al., 2008a). In order to show the adaptability of the new procedure to any ternary-based ECOC design, we also applied the re-coding methodology on sparse random designs. Moreover, the approach requires almost the same training and testing computational complexity than the classical one (since re-training classifiers is not required).

The paper is organized as follows: Section 2 describes the re-coded problem-dependent one-versus-one approach. Section 3 evaluates the methodology over a set of UCI data sets and two real multi-class problems: traffic sign and faces categorization. Finally, Section 4 concludes the paper.

2. Re-coded Error-Correcting Output Codes

In this section, we present a problem-dependent redefinition of the ternary ECOC designs. A ternary design is defined in the ternary ECOC framework $Q \in \{-1, 0, +1\}^{N \times M}$, where Q is a coding matrix of N rows (as the number of classes), M the number of columns (dichotomizers to be learnt, where $M = N(N-1)/2$ in the case of the one-versus-one design), $\{-1, +1\}$ the symbols that codify the class membership, and the zero symbol ignores a particular class for a given dichotomizer. Each column of the matrix Q corresponds to the ith binary problem h_i, which splits a sub-set of classes using a given base classifier. Fig. 2(a) and (b) show the binary problems and the one-versus-one coding matrix Q for a 4-class problem. The white positions correspond to the symbol +1, the black positions to the symbol -1, and the grey positions to the zero symbol. Note that this design is independent from the problem-domain, remaining the same for two problems with the same number of classes independently of the distribution of the data. Once the set of binary problems $h = \{h_1, \ldots, h_M\}$ is learnt, a new test sample ρ that arrives to the system is tested applying the set h, and a test codeword $x \in \{-1, +1\}^{1 \times M}$ is obtained. Note that the response of each binary problem when applied to ρ does not yield zero symbols, since it always has to strictly vote for either +1 or -1. Afterwards, a decoding function $d(x, y_j)$ is used to compare the test codeword x with each codeword y_j (jth row from Q) codifying class c_j. Finally, the classification prediction corresponds to the class c_j which corresponding codeword y_j minimizes d.

In the particular case of the one-versus-one ECOC design, only $2 \cdot M$ from the $N \cdot M$ possible positions are coded with $\{-1, +1\}$ symbols, which corresponds to a $(1 - 2/N) \cdot 100$ percentage of positions coded to zero. Note that the zero symbol does not give class membership information for its corresponding dichotomizer. Then, it could happen that if some of these positions coded to zero are re-coded to +1 or -1 without the need of re-training the dichotomizers, the final performance could be improved. For this task, the training data should be analyzed by the dichotomizers once in order to re-code the positions of the coding matrix Q, yielding a RE-coded problem-dependent ECOC design (RE-ECOC).

2.1. RE-ECOC coding

Given the training data $C = \{C_1, \ldots, C_N\}$, where $C_i = \{\rho_1, \ldots, \rho_m\}$ are the data samples belonging to class c_i and Q a coding matrix, the set of dichotomizers $h = \{h_1, \ldots, h_M\}$ is learnt applying a base classifier over the corresponding subsets of C, obtaining a ternary ECOC design. In order to update the coding matrix in a problem-dependent way, for each position $Q(i,j) = 0$, the corresponding data $C_i, i \in \{1, \ldots, N\}, i \notin \delta_j$, where δ_j is the set of classes considered by the jth dichotomizer, are tested using h_j under the hypothesis that their membership should be +1, obtaining a classification accuracy a. Then, the matrix location is re-coded subjected to the following condition:

$$Q(i,j) = \begin{cases} +1, & \text{if } a \geqslant \alpha \\ -1, & \text{if } 1-a \geqslant \alpha \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

being $\alpha \in (0.5, 1]$ a performance threshold.

Then, a classification accuracy a is obtained. If the value of a or $(1-a)$ is greater than a performance threshold, then that position of the coding matrix Q is set (re-coded) to +1 (or -1), respectively. Otherwise, the value of $Q(i,j)$ is kept to zero.

Since we use the training data to modify the positions of Q, the design mutates in a problem-dependent way. Moreover, since the modification of the positions of Q does not require to re-train the set h, the computational cost of the coding process is not significantly increased. Table 1 shows the algorithm for re-coding a generic ternary ECOC design. The algorithm modifies the positions of Q based on the input value of α. Note that in the algorithm, a matrix of weights W saving the accuracy values a is defined. This matrix will be used at the decoding process in order to weight the final classification.

In the case of the one-versus-one ECOC design, the ternary coding matrix displays the highest sparseness degree of the state-of-the-art ECOC configurations. It contains a high number of candidate positions to be re-coded. Thus, for the rest of the paper we consider the one-versus-one coding matrix as the base design to apply the proposed re-coding methodology showed in Table 1.

2.1.1. Estimating the α parameter

In order to obtain more precise classification results, we need to know which values of α are useful to increase the generalization capability of the system, since some values of α may result in wrong classification predictions. In order to look for the values of α, cross-validation is applied. For this task, the training data C is split into a training C^T and a validation C^V subsets, so that $C = C^T \cup C^V$. The use of a validation subset helps the system to increase generalization. Thus, for a set of values $\alpha = \{\alpha_1, \ldots, \alpha_k\}$, algorithm Table 1 is called. However, the set h is only learnt once over C at the beginning. At each round, the set C^T is used to mutate the positions of Q, and the validation set C^V will be used to test the performance of each Q for a particular α. For this last task, a decoding procedure using the weighting matrix W is proposed next. This step is required to obtain a successful classification. Finally, the matrix Q for which value of α maximizes the classification performance over C^V is selected.

Table 1
Generic RE-ECOC learning algorithm.

Input: $\alpha, C = \{C_1, \ldots, C_N\}, Q$, and $h = \{h_1, \ldots, h_M\}$

```
// Accuracy value, multi-class data, coding matrix, and set of classifiers
Output: Q, W// Coding matrix and weight matrix
W^{N×M} := 0
for i ∈ {1,...,N}
    for j ∈ {i,...,M}
        if Q(i,j) = 0
            // Accuracy for class c_i considered as label +1
            a := h_j(C_j, +1)
            // Consider the coding matrix position as +1
            if a ⩾ α then
                // Update membership and accuracy
                Q(i,j) := +1, W(i,j) := a
            // Consider coding matrix position as −1
            elseif 1 − a ⩾ α then
                // Update membership and accuracy
                Q(i,j) := −1, W(i,j) := 1 − a
            endif
        elseif Q(i,j) = +1
            W(i,j) := h_j(C_i, +1) // Update accuracy
        else
            W(i,j) := h_j(C_i, −1) // Update accuracy
        endif
    endfor
endfor
```

Fig. 2 shows an example of a training process for a 4-class problem. Fig. 2(a) shows the non-linear decision boundaries that splits all possible pairs of classes. Fig. 2(b) shows the classical one-versus-one design. Fig. 2(c) shows the problem-dependent coding matrix Q for $\alpha = 0.9$. Note that several positions previously coded to zero are now set to +1 or -1 values since they achieve an accuracy upon 90% over the training data. Finally, Fig. 2(d) shows the same process for $\alpha = 1.0$. Now, less positions satisfy the performance restrictions. Note that in the current example if the testing of the validation data C^V does not take benefits from the values of α, then, the classical one-versus-one design is selected. Thus, in the worst case, the re-coded problem-dependent one-versus-one approach attains the same performance than the classical one.

2.2. RE-ECOC decoding

In (Escalera et al., 2008a), the authors show that to properly decode a ternary ECOC matrix two biases must be avoided at the decoding step. First, classical decoding strategies introduce a bias when comparing positions that contain the zero symbol, which does not give information about meta-class membership. On the other hand, the addition of the bias produced by the comparison with the zero symbol makes the codewords to take values from different ranges, which leads to non-comparable measures among codewords. In this sense, the authors present how to robustly decode sparse coding matrices where codewords may contain different number of positions coded to $\{-1, +1\}$ symbols. This is done by weighting the final decision so that it avoids the influence of the zero symbol at the same time that all classes codewords have the same probability of being predicted.

In order to take into account the previous properties, we use a Loss-Weighted decoding (Escalera et al., 2008a), which uses the weighting matrix W computed at the coding step to decode the RE-ECOC matrix Q. The approach uses a Loss-based model (Allwein et al., 2002) with a Loss-function $L(\theta)$ to penalize the miss-classifications produced by the set of dichotomizers h.

First, we normalize each row of the weighting matrix W obtained at the coding step so that M_W can be considered as a discrete probability density function $M_W(i,j) = \frac{W(i,j)}{\sum_{j=1}^{M} W(i,j)}$, $\forall i \in [1, \ldots, N]$, $\forall j \in [1, \ldots, M]$. Once we obtain the normalized weighting matrix M_W, we introduce it in the decoding function as follows:

$$LW(\rho, i) = \sum_{j=1}^{n} M_W(i,j) L(y_i^j \cdot h_j(\rho)) \quad (2)$$

where $L(\theta) = -\theta$. Then, the final classification decision is done by the class c_i which corresponding codeword y_i minimizes the LW function.

3. Results

In order to present the results, first, we discuss the data, methods, measurements, and experimental settings of the experiments.

- *Data*: The data used for the experiments consist of 11 multi-class data sets from the UCI Machine Learning Repository database (Asuncion and Newman, 2007). The number of training samples, features, and classes per data set are shown in Table 2. We also categorize two real Computer Vision classification problems. First, we use the video sequences obtained from a Mobile Mapping System (Casacuberta et al., 2004) to test the methods in a real traffic sign categorization problem consisting

Table 2
UCI repository data sets characteristics.

Problem	#Training samples	#Features	#Classes
Dermathology	366	34	6
Iris	150	4	3
Ecoli	336	8	8
Vehicle	846	18	4
Wine	178	13	3
Segmentation	2310	19	7
Glass	214	9	7
Thyroid	215	5	3
Vowel	990	10	11
Balance	625	4	3
Yeast	1484	8	10

of 36 traffic sign classes. Second, 30 classes from the ARFaces (Martinez and Benavente, 1998) data set are classified using the present methodology.

- *Methods*: We compare the classical one-versus-one ECOC design, one-versus-all (Nilsson, 1965), dense random (Allwein et al., 2002), and sparse random ECOC strategies (Allwein et al., 2002) with the RE-ECOC strategy for three base classifiers: Gentle Adaboost (Friedman et al., 1998), Linear Support Vector Machines (OSU-SVM-TOOLBOX, 2003), and Support Vector Machines with Radial Basis Function kernel (*RBF SVM*) (OSU-SVM-TOOLBOX, 2003). In order to compare the methods at the same conditions, we use a linear Loss-Weighted decoding for all ECOC strategies. Moreover, in order to analyze the adaptability of the re-coding design, we apply the re-coding methodology to sparse random designs. Once a sparse coding matrix is computed with the standard procedure, each position coded to zero is analyzed applying the re-coding methodology for different values of α. We call this method the RE-sparse design.
- *Measurements*: To measure the performance of the different strategies, we apply stratified 10-fold cross-validation and test for confidence interval with a two-tailed t-test. We also use the Friedman and Nemenyi tests (Demsar, 2006) to analyze the statistical significance among the obtained performances.
- *Experimental settings*: Fifty decision stumps are considered for the Gentle Adaboost algorithm. The *RBF SVM* classifier is tuned via cross-validation at each fold training set for each ensemble, where σ and the regularization parameters are tested from 0.05 increasing per 0.05 up to one and from one increasing per five up to 150, respectively. For the RE-ECOC strategy cross-validation on the training set is also applied, where α is tested from 0.7 increasing per 0.05 up to one, and 10% of the training data are used as a validation subset. The dense and sparse matrices are selected from a set of 20,000 random generated matrices with codewords of length N, where N corresponds to the number of classes.

3.1. UCI classification

Tables 3–5 show the performance results of the different ECOC configurations. For each UCI data set, the performance obtained by each method is shown. The symbol '*' is used to show the best performance between the one-versus-one and RE-ECOC configurations as well as between the sparse random and RE-sparse designs. The positions in bold correspond the best global performances for each data set. Note that though the number of improved data sets varies depending on the considered base classifiers, there are not experiments where the RE-ECOC designs obtains inferior results to the one-versus-one design. The same behavior occurs with the RE-sparse and the classical sparse random design.

In order to compare the performances provided for each strategy, Table 6 shows the mean rank of each ECOC design considering the 33 different experiments. The rankings are obtained estimating each particular ranking r_i^j for each problem i and each ECOC design j, and computing the mean ranking R for each decoding as $R_j = \frac{1}{N}\sum_i r_i^j$, where N is the total number of problems (3 base

Table 3
UCI accuracy using Gentle Adaboost base classifier and different ECOC configurations.

Problem	One-versus-one	RE-ECOC	One-versus-all	Dense	Sparse	RE-sparse
Balance	**87.46***	**87.46***	81.43	80.03	78.81	81.14*
Wine	94.38*	94.38*	91.15	**94.90**	94.38*	94.38*
Thyroid	**95.37***	**95.37***	87.48	92.15	92.15*	92.15*
Iris	**95.33***	**95.33***	**95.33**	**95.33**	**95.33***	**95.33***
Glass	63.10	**68.65***	58.14	64.12	59.23	65.14*
Ecoli	81.29	**83.36***	62.17	55.15	55.15	59.18*
Dermatology	91.76	**92.52***	82.97	91.56	74.95	91.25*
Vowel	57.88	**62.73***	50.92	44.21	43.85	47.52*
Vehicle	57.81	63.57*	72.89	**69.80**	**69.80***	**69.80***
Yeast	55.46	**56.67***	32.97	48.10	37.15	42.93*
Segmentation	**97.45***	**97.45***	82.11	93.18	92.45	93.47*

Table 4
UCI accuracy using Linear *SVM* base classifier and different ECOC configurations.

Problem	0m	RE-ECOC	One-versus-all	Dense	Sparse	RE-sparse
Balance	**91.64***	**91.64***	87.14	85.55	85.55*	85.55*
Wine	**95.55***	**95.55***	**95.55**	**95.55**	**95.55***	**95.55***
Thyroid	**96.71***	**96.71***	**96.71**	87.30	85.43	88.83*
Iris	**98.67***	**98.67***	94.18	97.30	97.30*	97.30*
Glass	28.74	37.58*	32.15	40.80	42.30	**47.83***
Ecoli	**74.63***	**74.63***	62.23	69.44	69.08	72.44*
Dermatology	94.79	**95.07***	89.14	93.15	88.93	93.15*
Vowel	63.33	**64.44***	43.80	38.52	34.58	38.93*
Vehicle	80.24*	80.24*	78.93	71.52	87.61	**91.71***
Yeast	26.11	**37.81***	21.88	33.47	21.15	32.47*
Segmentation	96.02	**96.32***	82.95	79.85	83.37	85.41*

Table 5
UCI accuracy using *RBF SVM* base classifier and different ECOC configurations.

Problem	One-versus-one	RE-ECOC	One-versus-all	Dense	Sparse	RE-sparse
Balance	97.25	**97.41***	87.88	84.97	85.04	88.43*
Wine	**95.55***	**95.55***	**95.55***	94.87	**95.55***	**95.55***
Thyroid	**95.35***	**95.35***	94.32	91.49	93.85	94.32*
Iris	**96.67***	**96.67***	**96.67**	**96.67**	**96.67***	**96.67***
Glass	46.41*	46.41*	44.14	43.80	44.38	**53.73***
Ecoli	**86.74***	**86.74***	83.15	82.94	85.15	**86.74***
Dermatology	88.80	89.05*	91.15	83.99	89.01	**93.87***
Vowel	54.95	**55.76***	47.15	51.85	50.08	52.80*
Vehicle	72.00	**72.12***	68.20	67.85	70.91	71.83*
Yeast	**56.68***	**56.68***	52.18	48.95	54.81*	54.81*
Segmentation	95.14	**95.25***	94.44	79.95	92.87	95.03*

Table 6
Mean rank for each ECOC design over all the experiments.

ECOC design	One-vs-one	RE-ECOC	One-vs-all	Dense	Sparse	RE-sparse
Mean rank	2.36	1.29	4.80	4.50	4.01	2.91

Table 7
UCI accuracy using Gentle Adaboost base classifier.

Problem	One-versus-one	RE-ECOC	α	Wins	Losses	Draws
Balance	87.46	87.46	–	0	0	10
Wine	94.38	94.38	–	0	0	10
Thyroid	95.37	95.37	–	0	0	10
Iris	95.33	95.33	–	0	0	10
Glass	63.10	68.65	0.95	10	0	0
Ecoli	81.29	83.36	0.75	8	2	0
Dermatology	91.76	92.52	0.85	5	0	5
Vowel	57.88	62.73	0.95	9	1	0
Vehicle	57.81	63.57	0.95	9	1	0
Yeast	55.46	56.67	0.95	5	2	3
Segmentation	97.45	97.45	–	0	0	10

classifiers $\times$ 11 databases). One can see that the RE-ECOC strategy ECOC attains the best position in most of the experiments. To analyze if the difference between methods ranks is statistically significant, we apply the Friedman and Nemenyi tests. In order to reject the null hypothesis that the measured ranks differ from the mean rank, and that the ranks are affected by randomness in the results, we use the Friedman test. The Friedman statistic value is computed as follows:

$$X_F^2 = \frac{12N}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right] \tag{3}$$

In our case, with $k = 6$ ECOC designs to compare, $X_F^2 = 16.20$. Since this value is undesirable conservative, Iman and Davenport proposed a corrected statistic:

$$F_F = \frac{(N-1)X_F^2}{N(k-1) - X_F^2} \tag{4}$$

Applying this correction we obtain $F_F = 3.48$. With six methods and 33 experiments, F_F is distributed according to the F distribution with 5 and 160 degrees of freedom. The critical value of $F(5,160)$ for 0.05 is 2.21. As the value of F_F is higher than 2.21 we can reject the null hypothesis. One we have checked for the non-randomness of the results, we can perform a post hoc test to check if one of the techniques can be singled out. For this purpose we use the Nemenyi test – two techniques are significantly different if the corresponding average ranks differ by at least the critical difference value (CD):

$$\mathrm{CD} = q_\alpha\sqrt{\frac{k(k+1)}{6N}} \tag{5}$$

where q_α is based on the Studentized range statistic divided by $\sqrt{2}$. In our case, when comparing six methods with a confidence value $\alpha = 0.10, q_{0.10} = 1.44$. Substituting in Eq. (5), we obtain a critical difference value of 0.66. Since the difference of any technique rank with the RE-ECOC rank is higher than the CD, we can infer that the RE-ECOC approach is significantly better than the rest with a confidence of 90% in the present experiments. Moreover, the difference among the one-versus-one and RE-sparse procedure is less than the CD value, being then statistically comparable in the second position of the ranking. Note that the RE-sparse design considerably improves the results obtained by the classical sparse design, obtaining significant performance improvements in several data sets.

Now, we analyze the performances obtained by the RE-ECOC strategy, which showed the best results in the previous experiment, in comparison with the classical one-versus-one. Tables 7–9 show the performance results of the one-versus-one ECOC and RE-ECOC algorithms. For each UCI data set, the performance obtained by each method is shown. In the cases where RE-ECOC improves the one-versus-one ECOC results, the selected values of α are shown. The number of wins, losses, and draws considering the 10 experiments of the 10-fold cross-validation for each data set are also shown in the table. Note that in several data sets, RE-ECOC obtains performance improvements for the three base classifiers. The tables show that the more classes there are, the more

Table 8
UCI accuracy using Linear *SVM*.

Problem	One-versus-one	RE-ECOC	α	Wins	Losses	Draws
Balance	91.64	91.64	–	0	0	10
Wine	95.55	95.55	–	0	0	10
Thyroid	96.71	96.71	–	0	0	10
Iris	98.67	98.67	–	0	0	10
Glass	28.74	37.58	1.00	5	1	4
Ecoli	74.63	74.63	–	0	0	10
Dermatology	94.79	95.07	0.95	1	0	9
Vowel	63.33	64.44	0.95	8	2	0
Vehicle	80.24	80.24	–	0	0	10
Yeast	26.11	37.81	0.95	9	1	0
Segmentation	96.02	96.32	1.00	6	2	2

Table 9
UCI accuracy using *RBF SVM*.

Problem	One-versus-one	RE-ECOC	α	Wins	Losses	Draws
Balance	97.25	97.41	0.95	1	0	9
Wine	61.31	61.84	1.00	1	0	9
Thyroid	95.35	95.35	–	0	0	10
Iris	96.67	96.67	–	0	0	10
Glass	46.41	46.41	–	0	0	10
Ecoli	86.74	86.74	–	0	0	10
Dermatology	88.80	89.05	0.85	3	0	7
Vowel	54.95	55.76	0.90	4	1	5
Vehicle	72.00	72.12	0.90	1	0	9
Yeast	56.68	56.68	–	0	0	10
Segmentation	95.14	95.25	0.90	2	0	8

significant the results are. The highest performances are achieved for high values of α (about 0.90–0.95 in most cases). The general intuition is that if the original binary classifier achieves high performance by itself, a high performance of the re-coded positions is also required to improve the results. Note that in the worst case, RE-ECOC achieves the same performance than the one-versus-one ECOC design. Moreover, looking at the wins and losses of each experiment, one can see that though in some cases the performance improvements of RE-ECOC are no significant, the number of wins of the 10-fold experiments are statistically significant.

Now, we compare the results obtained by the RE-ECOC approach on the UCI data sets with the results obtained with the same strategy re-training classifiers. In Fig. 3 one can see the performance obtained by both classification strategies for the three different base classifiers. Note that there are not significant differences among the obtained performances. Moreover, the RE-ECOC strategy obtains better performance in more cases than using the same coding matrix re-training classifiers, with far less computational complexity.

3.2. Traffic sign categorization

For this experiment, we use the video sequences obtained from a Mobile Mapping System (Casacuberta et al., 2004) to test the classification methodology on a real traffic sign categorization problem. In this system, the position and orientation of the different traffic signs are measured with video cameras fixed on a moving vehicle. From this system, a set of 36 circular and triangular traffic sign classes are obtained. Some classes from this data set are shown in Fig. 4. The data set contains a total of 3481 samples of size 32×32, filtered using Weickert anisotropic filter, masked to exclude the background pixels, and equalized to deal with illumination changes. These feature vectors are then projected into a 100 feature vector by means of PCA.

The classification results of the one-versus-one ECOC and RE-ECOC strategies for the three base classifiers are shown in Table 10. In this experiment, for all base classifiers, the RE-ECOC design obtains performance improvements for high values of α.

3.3. ARFaces classification

Martinez and Benavente (1998) is composed of 26 face images from 126 different subjects (70 men and 56 women). The images have uniform white background. The database has two sets of images from each person, acquired in two different sessions, with the following structure: one sample of neutral frontal images, three samples with strong changes in the illumination, two samples with occlusions (scarf and glasses), four images combining occlusions and illumination changes, and three samples with gesture effects. One example of each type is plotted in Fig. 5. For this experiment, we selected all the samples from 30 different classes (persons).

The classification results of the one-versus-one ECOC and RE-ECOC strategies for the three base classifiers are shown in Table 11. As in the previous experiments, all base classifiers obtain performance improvements using the RE-ECOC strategy for high values of α ($\alpha = 0.95$).

3.4. Discussion

As a final conclusion of the results, we can state that performance improvements are obtained using the RE-ECOC approach instead of the original one-versus-one ECOC. For the three base classifiers used in this paper, one can see that RE-ECOC performs better than the rest of ECOC designs in more data sets using Adaboost and Linear *SVM* instead of *RBF SVM*. Note that *RBF SVM* has been tuned via cross-validation so that the binary problems learnt are optimal in the sense of the kernel parameters for those data sets. Thus, the new re-coded matrix positions should include highly accurate coding information to increase the generalization of the design. It is the main reason why *RBF SVM* may obtain several draws with the one-versus-one approach in different classification problems. However, note that for the *RBF SVM* there are

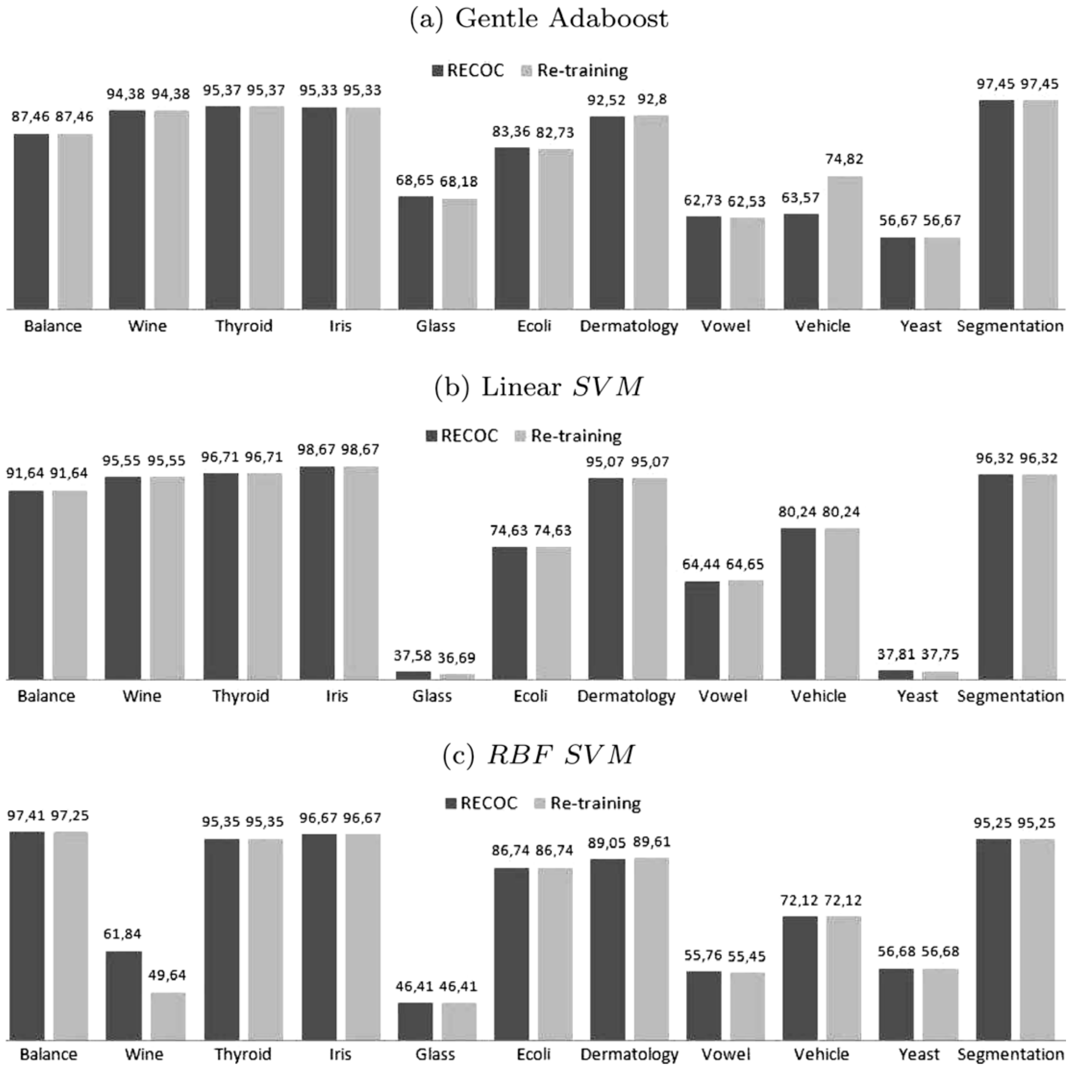

Fig. 3. UCI data sets performance using the re-coded matrix with and without re-training. (a) Gentle Adaboost, (b) Linear *SVM*, and (c) *RBF SVM*.

Fig. 4. Traffic sign classes.

statistically significant performance improvement in terms of ranking for six of the UCI data sets, the same results than one-versus-one for the rest of data sets, and improvements in the two real applications. Concerning the Adaboost and Linear *SVM*, in about half of the UCI data sets and in both real applications performance improvements are obtained. Moreover, the experiments shows that the more classes are considered, the more significant the improvements are. Finally, note that none of the RE-ECOC experi-

Fig. 5. ARFaces data set classes. Examples from a class with neutral, smile, anger, scream expressions, wearing sun glasses, wearing sunglasses and left light on, wearing sun glasses and right light on, wearing scarf, wearing scarf and left light on, and wearing scarf and right light on.

Table 10
Traffic data set accuracy.

Problem	One-versus-one	RE-ECOC	α	Wins	Losses	Draws
Gentle Adaboost	88.70	88.95	0.95	3	1	6
Linear *SVM*	88.02	91.23	1.00	4	0	6
RBF SVM	97.44	97.85	0.95	1	0	9

Table 11
ARFaces data set accuracy.

Problem	One-versus-one	RE-ECOC	α	Wins	Losses	Draws
Gentle Adaboost	65.50	70.06	0.95	6	1	3
Linear *SVM*	39.41	43.92	0.95	9	1	0
RBF SVM	88.33	88.75	0.95	2	0	8

ments for any base classifier obtains inferior results to the one-versus-one performance.

Concerning the computational complexity of the strategy, the classifiers learnt at the coding step are not re-trained during the RE-ECOC re-codification. Thus, though cross-validation of α should be applied to assure the best performance, the training cost is not significantly increased. On the other hand, testing time remains the same than in the standard codings since all classifiers should be applied on the test sample, and the comparison among matrix codewords must also be performed at the decoding step. Moreover, we shown that we obtain similar (even superior) results with the re-coded RE-ECOC matrix Q than using the same procedure but re-training classifiers (that is, using the re-coded positions to re-train again the dichotomizers).

Finally, it is important to bring up that though the re-coding strategy has been performed on the one-versus-one and sparse random coding matrices, this strategy is directly applicable to any kind of ternary ECOC design where the symbol zero may appear.

4. Conclusion

In this paper, we presented a problem-dependent design of Error-Correcting Output Codes to deal with multi-class categorization problems. The method is based on redefining the classical one-versus-one ECOC design so that the generalization of the system is increased. For this task, the training data are analyzed using the previously learnt binary problems, and the coding matrix is re-coded without the need of re-training classifiers. A weighting matrix is also included in order to weight the final classification and obtain more precise results. The experimental evaluation over several UCI Machine Learning repository data sets and two real multi-class problems: traffic sign and faces categorization, show that significant performance improvements can be obtained compared to different state-of-the-art ECOC configurations, assuring at least the same performance than the one-versus-one ECOC design.

Acknowledgments

This work has been supported in part by projects TIN2009 and CONSOLIDER-INGENIO CSD 2007-00018.

References

Allwein, E., Schapire, R., Singer, Y., 2002. Reducing multiclass to binary: A unifying approach for margin classifiers. J. Mach. Learn. Res. 1, 113–141.

Asuncion, A., Newman, D.J., 2007. UCI Machine Learning Repository, University of California, Irvine, School of Information and Computer Sciences.

Casacuberta, J., Miranda, J., Pla, M., Sanchez, S., Serra, A., Talaya, J., 2004. On the Accuracy and Performance of the GeoMobil system, International Society for Photogrammetry and Remote Sensing.

Demsar, J., 2006. Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. 7, 1–30.

Dietterich, T., Bakiri, G., 1995. Solving multiclass learning problems via Error-Correcting Output Codes. J. Artificial Intell. Res. 2, 263–282.

Escalera, S., Pujol, O., Radeva, P., 2008a. On the decoding process in ternary Error-Correcting Output Codes, IEEE transactions in pattern analysis and machine intelligence. IEEE computer Society Digital Library. IEEE Computer Society. <http://doi.ieeecomputersociety.org/10.1109/TPAMI.2008.266>.

Escalera, S., Tax, D., Pujol, O., Radeva, P., Duin, R., 2008b. Subclass problem-dependent design of Error-Correcting Output Codes. IEEE Trans. Pattern Anal. Mach. Intell. 30, 1041–1054.

Friedman, J., Hastie, T., Tibshirani, R., 1998. Additive logistic regression: A statistical view of boosting. Ann. Stat. 38, 337–374.

Kong, E.B., Dietterich, T.G., 1995. Error-Correcting Output Coding corrects bias and variance. In: ICML, pp. 313–321.

Martinez, A., Benavente, R., 1998. The AR face database. Computer Vision Center Technical Report #24.

Nilsson, N.J., 1965. Learning Machines. McGraw-Hill.

OSU-SVM-TOOLBOX. 2003. <http://svm.sourceforge.net>.

Pujol, O., Radeva, P., Vitrià, J., 2006. Discriminant ECOC: A heuristic method for application dependent design of error correcting output codes. IEEE Trans. Pattern Anal. Mach. Intell. 28, 1001–1007.