Subclass Error Correcting Output Codes using Fisher's Linear Discriminant Ratio

Nikolaos Arvanitopoulos Dimitrios Bouzas
Anastasios Tefas
Aristotle University of Thessaloniki Department of Informatics
Artificial Intelligence & Information Analysis Laboratory
{*niarvani,dmpouzas*}@*csd.auth.gr*, tefas@aiia.csd.auth.gr

Abstract

Error-Correcting Output Codes (ECOC) with subclasses reveal a common way to solve multi-class classification problems. According to this approach, a multi-class problem is decomposed into several binary ones based on the maximization of the mutual information (MI) between the classes and their respective labels. The MI is modelled through the fast quadratic mutual information (FQMI) procedure. However, FQMI is not applicable on large datasets due to its high algorithmic complexity. In this paper we propose Fisher's Linear Discriminant Ratio (FLDR) as an alternative decomposition criterion which is of much less computational complexity and achieves in most experiments conducted better classification performance. Furthermore, we compare FLDR against FQMI over the Cohn-Kanade facial expression recognition dataset.

1. Introduction

In the literature one can find various binary classification techniques. However, in the real world the problems to be addressed are usually multi-class. In dealing with multi-class problems we must use the binary techniques as a leverage. This can be achieved by defining a method that decomposes the multi-class problem into several binary ones, and combines their solutions to solve the initial multi-class problem [1]. In this context, the ECOC procedure emerged [4].

As proposed by Escalera et al., on the ECOC framework we can apply the *sub-class technique* [5]. According to this technique, we group and split the classes into subsets with respect to the improvement we obtain in the training performance, resulting in more but easier to solve binary problems.

In the resulting *sub-ECOC* technique, the decomposition of the multi-class problem is achieved by maximizing the MI between the classes and their respective labels via the *Sequential Forward Floating Search (SFFS)* algorithm [8]. The MI is computed through the FQMI method [10]. However, FQMI is computationally costly. In this paper we propose the FLDR as an alternative optimization criterion in the SFFS algorithm, which is of much less computational complexity and achieves in most cases better classification performance over a number of artificial, UCI machine learning repository [2] and the Cohn-Canade facial expression recognition [6] multi-class datasets.

2. Sub-ECOC Framework

The ECOC consists of two separate steps: a) the *encoding* and b) the *decoding* step [9]. In the encoding step, given a set of N classes, we assign a unique binary string called *codeword* to each class. The length n of each codeword represents the number of *bi-partitions* (i.e., groups of classes) that are formed and, consequently, the number of binary problems to be trained. Each bit of the codeword represents the response of the corresponding binary classifier and it is coded by +1, 0 or -1, where 0 means that a certain class is not considered by a specific binary classifier [1]. The next step is to arrange all these codewords as rows of a matrix obtaining the so-called *coding matrix* $\mathbf{M} \in \{-1, 0, +1\}^{N \times n}$. Each column of this matrix defines a partition of classes, while each row defines the membership of the corresponding class in the specific binary problem.

The decoding step consists of applying the n different binary classifiers to each data sample in the test set, in order to obtain a code for this sample. This code is then compared to all the codewords of the classes de-

fined in the coding matrix $\mathbf{M}$ and the sample is assigned to the class with the closest codeword. The most frequently used decoding methods are the *Hamming* and the *Euclidean* decoding distances.

Additionally to the ECOC framework Pujol proposed that we can use a ternary problem dependent design of ECOC, called *discriminant ECOC (DECOC)* where, given a number of N classes, we can achieve a high classification performance by training only $N-1$ binary classifiers [9]. This can be achieved by finding via the SFFS searching procedure an encoding matrix $\mathbf{M}$ with high discriminative power. The SFFS algorithm is described in Algorithm 1.

Algorithm 1 SFFS for Classes

Input:
$Y = \{y_j | j = 1, \ldots, N_c\}$ // available classes
Output: // disjoint subsets with maximum MI between the features and their class labels
$X_k = \{x_j | j = 1, \ldots, k,\ x_j \in Y\},\ k = 0, 1, \ldots, N_c$
$X'_{k'} = \{x_j | j = 1, \ldots, k',\ x_j \in Y\},\ k' = 0, 1, \ldots, N_c$
Initialization:
$X_0 := \emptyset,\ X'_{N_c} := Y;\quad k := 0, k' := N_c$ // k and k' denote the number of classes in each subset
Termination:
Stop when $k = N_c$ and $k' = 0$
Step 1 (*Inclusion*)
// x^+ is the most significant class with respect to the group $\{X_k, X'_{k'}\}$
$x^+ := \arg \max_{x \in Y - X_k} J(X_k + x, X'_{k'} - x)$
$X_{k+1} := X_k + x^+;\ X'_{k'-1} := X'_{k'} - x^+;\ k := k+1, k' := k'-1$
Step 2 (*Conditional exclusion*)
// x^- is the least significant class with respect to the group $\{X_k, X'_{k'}\}$
$x^- := \arg \max_{x \in X_k} J(X_k - x, X'_{k'} + x)$
if $J(X_k - x^-, X'_{k'} + x^-) > J(X_{k-1}, X'_{k'+1})$ **then**
    $X_{k-1} := X_k - x^-;\ X'_{k'+1} := X'_{k'} + x^-;\ k := k-1, k' := k'+1$
    go to **Step 2**
else
    go to **Step 1**
end if

On the DECOC procedure Escalera et al. proposed that from an initial set of classes $\mathcal{C}$ of a given multi-class problem, using a clustering method (e.g., K-means) we can define a new set of classes $\mathcal{C}'$, where $|\mathcal{C}'| > |\mathcal{C}|$, obtaining a new configuration of binary problems that are easier to solve [5]. In the resulting sub-ECOC procedure, we compare in each binary decomposition the training performances of the created classifiers against a user defined threshold θ_p, which denotes the maximum training error the classifier addressing each binary decomposition should attain. If θ_p is not satisfied, we split the classes of the binary decomposition into subclasses. The size (i.e., number of samples) of the created subclasses is compared against another user defined threshold θ_s, which defines the minimum size of a sub-class. If θ_s is satisfied, we create two new classifiers to address the new created sub-problems. The performance of the two new created classifiers is compared against a third user defined threshold θ_i, which defines the improvement we want to obtain in the training error with respect to the initial classifier's performance. If these thresholds are satisfied, the new created pair of sub-problems is accepted along with their new created binary classifiers, otherwise they are rejected and we keep the initial configuration with its respective binary classifier [5].

3. FQMI

Consider two random vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ and let $p(\mathbf{x}_1)$ and $p(\mathbf{x}_2)$ be their probability density functions respectively. Then the MI of $\mathbf{x}_1$ and $\mathbf{x}_2$ can be regarded as a measure of the dependence between them and is defined as follows:

$$\mathcal{I}(\mathbf{x}_1, \mathbf{x}_2) = \int\int p(\mathbf{x}_1, \mathbf{x}_2) \log \frac{p(\mathbf{x}_1, \mathbf{x}_2)}{p(\mathbf{x}_1)p(\mathbf{x}_2)} d\mathbf{x}_1 d\mathbf{x}_2 \quad (1)$$

It is of great importance to mention that (1) can be interpreted as a Kullback-Leibler divergence, defined as follows:

$$\mathcal{K}(f_1, f_2) = \int f_1(\mathbf{x}) \log \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} dx \quad (2)$$

where $f_1(\mathbf{x}) = p(\mathbf{x}_1, \mathbf{x}_2)$ and $f_2(\mathbf{x}) = p(\mathbf{x}_1)p(\mathbf{x}_2)$.

According to Kapur and Kesavan [7], if we seek to find the distribution that maximizes or alternatively minimizes the divergence, several axioms could be relaxed and it can be proven that $\mathcal{K}(f_1, f_2)$ is analogically related to $D(f_1, f_2) = \int (f_1(\mathbf{x}) - f_2(\mathbf{x}))^2 dx$. Consequently, maximization of $\mathcal{K}(f_1, f_2)$ leads to maximization of $D(f_1, f_2)$ and vice versa. Considering the above we can define the *quadratic mutual information* as

$$\mathcal{I}_Q(\mathbf{x}_1, \mathbf{x}_2) = \int\int (p(\mathbf{x}_1, \mathbf{x}_2) - p(\mathbf{x}_1)p(\mathbf{x}_2))^2 d\mathbf{x}_1 d\mathbf{x}_2 \quad (3)$$

The practical implementation of the FQMI computation is defined as follows: Let N be the number of pattern samples in the entire data set, J_i the number of samples of class i, N_c the number of classes in the entire data set, $\mathbf{x}_i$ the ith feature vector of the data set, and $\mathbf{x}_{ij}$ the jth feature vector of the set in class i. Consequently, $p(\mathbf{x})$, $p(y = y_i)$ and $p(\mathbf{x}|y = y_i)$, where $1 \leq i \leq N_c$ can be written as:

$$\begin{aligned} p(\mathbf{x}) &= \frac{1}{N} \sum_{j=1}^{J_i} \mathcal{N}(\mathbf{x} - \mathbf{x}_j, \sigma^2 I), \\ p(y = y_i) &= \frac{J_i}{N}, \\ p(\mathbf{x}|y = y_i) &= \frac{1}{J_i} \sum_{j=1}^{J_i} \mathcal{N}(\mathbf{x} - \mathbf{x}_{ij}, \sigma^2 I). \end{aligned}$$

By the expansion of (3) while using a Parzen estimator with a *Gaussian kernel* $\mathcal{N}$ we get the following equation:

$$\mathcal{I}_Q(\mathbf{x}, y) = V_{IN} + V_{ALL} - 2V_{BTW} \quad (4)$$

where

$$V_{IN} = \sum_y \int_{\mathbf{x}} p(\mathbf{x}, y)^2 d\mathbf{x}$$
$$= \frac{1}{N^2} \sum_{i=1}^{N_c} \sum_{l=1}^{J_i} \sum_{k=1}^{J_i} \mathcal{N}(\mathbf{x}_{il} - \mathbf{x}_{ik}, 2\sigma^2 I) \quad (5)$$

$$V_{ALL} = \sum_y \int_{\mathbf{x}} p(\mathbf{x})^2 p(y)^2 d\mathbf{x}$$
$$= \frac{1}{N^2} \sum_{i=1}^{N_c} \left(\frac{J_i}{N}\right)^2 \sum_{l=1}^{N} \sum_{k=1}^{N} \mathcal{N}(\mathbf{x}_l - \mathbf{x}_k, 2\sigma^2 I) \quad (6)$$

$$V_{BTW} = \sum_y \int_{\mathbf{x}} p(\mathbf{x}, y) p(\mathbf{x}) p(y) d\mathbf{x}$$
$$= \frac{1}{N^2} \sum_{i=1}^{N_c} \frac{J_i}{N} \sum_{l=1}^{N} \sum_{k=1}^{J_i} \mathcal{N}(\mathbf{x}_l - \mathbf{x}_{ik}, 2\sigma^2 I) \quad (7)$$

It is known that the FQMI requires many samples to be accurately computed by Parzen window estimation. Thus, we can assume that when the number of samples N is much greater than their respective dimensionality d (i.e., $N >> d$), the complexity of V_{ALL}, which is $O(N_c N^2 d^2)$, is dominant for the equation (4).

4 FLDR

Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be two classes of a binary classification problem. The Fisher's Linear Discriminant ratio is defined as:

$$J(\mathbf{w}) = \frac{|\mathbf{m}_1 - \mathbf{m}_2|^2}{\mathbf{s}_1^2 + \mathbf{s}_2^2} \quad (8)$$

where $\mathbf{m}_1$, $\mathbf{m}_2$ are the sample means and $\mathbf{s}_1$, $\mathbf{s}_2$ the variances of classes $\mathcal{C}_1$ and $\mathcal{C}_2$ respectively. We define the scatter matrices $\mathbf{S}_w$ and $\mathbf{S}_b$ as

$$\mathbf{S}_w = \sum_{\mathbf{x} \in \mathcal{C}_1} (\mathbf{x} - \mathbf{m}_1)(\mathbf{x} - \mathbf{m}_1)^T + \sum_{\mathbf{x} \in \mathcal{C}_2} (\mathbf{x} - \mathbf{m}_2)(\mathbf{x} - \mathbf{m}_2)^T \quad (9)$$

and

$$\mathbf{S}_b = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T \quad (10)$$

The matrix $\mathbf{S}_w$ is the within-class scatter matrix and the matrix $\mathbf{S}_b$ is the between class scatter matrix. As a result of the above, $J(\cdot)$ can be written as

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}} \quad (11)$$

The within-class scatter matrix $\mathbf{S}_w$ can be considered as a class density indicator and, as such, corresponds to the V_{IN} term in equation (5), which is a class density measure as well. Furthermore, the between-class scatter matrix $\mathbf{S}_b$ can be considered as a class location indicator inversely analogue to the V_{BTW} term in equation (7), which is also a class similarity measure. Thus, we can define the ratio

$$J' = \frac{tr\{\mathbf{S}_b\}}{tr\{\mathbf{S}_w\}} \quad (12)$$

as an alternative maximization criterion function that can be used in the SFFS procedure in similar manner to FQMI.

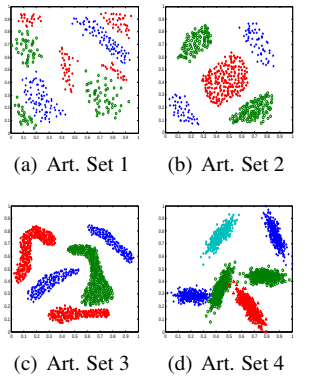
As can be seen in equation (12), the FLDR runs in linear time with respect to the number of the dataset samples N (i.e., $O(d^2 N)$), where d denotes samples' dimensionality. From the above it is obvious that FLDR clearly outranks FQMI in terms of computational complexity, making more appealing the use of the promising sub-ECOC approach to large datasets.

5 Experimental Results

Datasets. We compared FQMI and FLDR using 7 datasets of the UCI Machine Learning Repository, 4 artificially created 2D datasets and the Cohn-Kanade Facial Expression Recognition Database. The characteristics of each UCI dataset can be seen in Table 1. The 2D artificial datasets are illustrated in Fig. 1.

Table 1. UCI Machine Learning Repository Data Sets Characteristics

Database	Samples	Attributes	Classes
Iris	150	4	3
Ecoli	336	8	8
Wine	178	13	3
Glass	214	9	7
Thyroid	215	5	3
Vowel	990	10	11
Balance	625	4	3



(a) Art. Set 1 (b) Art. Set 2

(c) Art. Set 3 (d) Art. Set 4

Figure 1. Artificial Datasets

The Cohn-Kanade Facial Expression Database consists of approximately 704 image sequences (40×30 pixels) from 100 subjects (mixed race, gender, appearance) with 7 facial expressions (6 basic facial expressions sad, happy, surprise, anger, fear, disgust and 1 neutral).

The features of the UCI datasets were scaled to the interval $[-1, +1]$, the artificial datasets to $[0, 1]$, whereas those of the Cohn-Kanade dataset where standardized to zero mean and unit variance. To evaluate the test error on the various experiments we used 10-fold cross validation.

Classifier. As a standard classifier for our experiments we used the LIBSVM's implementation of the Support Vector Machine with linear and RBF kernels [3]. SVMs are very powerful classifiers that give improved results in the test domain. For both linear and RBF SVM we fixed the SVM parameters to C = 100 and $\sigma = 1$ for the UCI datasets, to C = 1 and $\sigma = 0.5$ for the artificial 2D datasets and C = 100 and $\sigma = 1/d$ (where $d = 1200$ the dimensionality of the dataset) for the Cohn-Kanade dataset.

Sub-ECOC configuration. The set of the threshold parameters where fixed to the following values: $\theta = \{\theta_p = 0\%, \theta_s = \frac{N}{50}, \theta_i = 5\% \}$ where N is the number of samples in each dataset.

As a clustering method we used the K-means algorithm with the number of clusters $K = 2$ which obtains similar results with other more sophisticated clustering algorithms, such as hierarchical and graph cut clustering, but with much less computational cost [5].

In Tables 2 and 3 we present the classification performances of our experiments in the UCI, artificial 2D and Cohn-Canade datasets using the DECOC and the sub-ECOC approaches.

Table 2. Artificial 2D Datasets

	Linear SVM				RBF SVM			
	FQMI		FLDR		FQMI		FLDR	
Database	ECOC	sub-ECOC	ECOC	sub-ECOC	ECOC	sub-ECOC	ECOC	sub-ECOC
A. D. 1	**37.54%**	68.34% (27×28.7)	**37.54%**	**76.35%** (10.8×9.8)	**57.27%**	68.23% (20.4×22.3)	**57.27%**	**77.52 %** (11×10)
A. D. 2	40.45%	**100%** (5×4)	**44.91%**	**100%** (5×4)	97.99%	**100%** (5×4)	**98.32%**	**100%** (4.5×3.5)
A. D. 3	**65.65%**	98.38% (15.6×15.5)	**65.65%**	**99.71%** (7×6)	**86.31%**	93.94% (11.4×11.4)	**86.31%**	**99.71%** (7×6)
A. D. 4	**88.54%**	90.83% (6.7×5.7)	**88.54%**	**97.12%** (5×4)	**99.83%**	**99.83%** (4×3)	**99.83%**	**99.83%** (4×3)

Table 3. UCI & Cohn-Kanade Datasets

	Linear SVM				RBF SVM			
	FQMI		FLDR		FQMI		FLDR	
Database	ECOC	sub-ECOC	ECOC	sub-ECOC	ECOC	sub-ECOC	ECOC	sub-ECOC
Iris	**96%**	**96%** (3×2)	**96%**	**96%** (3×2)	**96%**	**96%** (3×2)	**96%**	**96 %** (3×2)
Ecoli	**82.98%**	80.22% (11.1×12.7)	82.72%	**81.51%** (8.1×7.1)	82.83%	83.42% (11.8×13.4)	**85.04%**	**85.04%** (8.5×7.5)
Wine	95.52%	95.52% (3×2)	**96.6%**	**96.6%** (3×2)	**97.74%**	**97.74%** (3×2)	**97.74%**	**97.74%** (3×2)
Glass	**63.16%**	**67.04%** (14.6×15.9)	62.43%	65.34% (10.3×9.7)	**69.39%**	**70.76%** (7.6×7.3)	68.48%	68.48% (6.3×5.3)
Thyroid	**96.77%**	**96.77%** (3.3×2.6)	**96.67%**	**96.67%** (3×2)	**95.33%**	**95.33%** (3.2×2.4)	**95.35%**	**95.35%** (3×2)
Vowel	73.43%	77.47% (22.4×23.3)	**75.56%**	**85.35%** (20.9×19.9)	**99.09%**	**99.09%** (11×10)	98.99%	98.99% (11×10)
Balance	**91.7%**	89.31% (44.3×51.7)	**91.7%**	**89.6%** (9.6×8.5)	95.04%	95.04% (3×2)	**95.2%**	**95.2%** (3×2)
Kanade	79.81%	79.81% (7×6)	**80.93%**	**80.93%** (7×6)	73.35%	73.35% (7×6)	**74.58%**	**74.58%** (7×6)

In each column we illustrate the corresponding 10 fold cross-validation performance and in the case of the sub-ECOC method the (mean number of rows $\times$ mean number of columns) of the encoding matrices which are formed in each fold.

From the results it is obvious that FLDR attains in most cases better performance than the FQMI criterion, whereas in terms of computational speed in the largest of our datasets (i.e., Cohn-Kanade) FLDR trains in less than one minute, while the FQMI trains in approximately 5 days. Moreover, FLDR over-fits less than FQMI in the case where the use of sub-classes causes overtraining.

6 Conclusion

Due to its high computational complexity FQMI makes the use of the sub-ECOC technique impractical for large classification problems. As it has been illustrated in our paper, we can substitute FQMI with the FLDR which is of much less computational complexity and attains in most cases better classification results. This makes the promising sub-ECOC method applicable in large datasets arising in applications such as Facial Expression Recognition, datamining (e.t.c).

References

[1] E. L. Allwein, R. E. Schapire, and Y. Singer. Reducing multi-class to binary: A unifying approach for margin classifiers. *Journal of Machine Learning Research*, 1:113–141, 2002.

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

[3] C.-C. Chang and C.-J. Lin. LIBSVM: a library for support vector machines., 2001.

[4] T. G. Dietterich and G. Bakiri. Solving multi-class learning problems via error-correcting output codes. *Journal of Machine Learning Research*, 2:263–282, 1995.

[5] S. Escalera, D. M. Tax, O. Pujol, P. Radeva, and R. P. Duin. Subclass problem-dependent design for error-correcting output codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):1041–1054, June 2008.

[6] T. Kanade, J. F. Cohn, and Y. li Tian. Comprehensive database for facial expression analysis. In *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, pages 46–53. Grenoble, France, 2000.

[7] J. Kapur and H. Kesavan. *Entropy Optimization principles with Applications.* 1992.

[8] P. Pudil, F. Ferri, J. Novovicova, and J. Kittler. Floating search methods for feature selection with non-monotonic criterion functions. *Proc. Int'l Conf. Pattern Recognition*, 3:279–283, March 1994.

[9] O. Pujol, P. Radeva, and J. Vitria. Discriminant ecoc a heuristic method for application dependent design of error correcting output codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:1001–1007, June 2006.

[10] K. Torkkola. Feature extraction by non-parametric mutual information maximization. *Journal of Machine Learning Research*, 3:1415–1438, March 2003.