Least Squares and Estimation Measures via Error Correcting Output Code

Reza Ghaderi and Terry Windeatt

Centre for Vision, Speech and Signal Processing (CVSSP)
University of Surrey, Guildford, Surrey, GU2 5XH, U.K.
R.Ghaderi,T.Windeatt@eim.surrey.ac.uk

Abstract. It is known that the Error Correcting Output Code (ECOC) technique can improve generalisation for problems involving more than two classes. ECOC uses a strategy based on calculating distance to a class label in order to classify a pattern. However in some applications other kinds of information such as individual class probabilities can be useful. Least Squares(LS) is an alternative combination strategy to the standard distance based measure used in ECOC, but the effect of code specifications like the size of code or distance between labels has not been investigated in LS-ECOC framework. In this paper we consider constraints on choice of code matrix and express the relationship between final variance and local variance. Experiments on artificial and real data demonstrate that classification performance with LS can be comparable to the original distance based approach.

1 Introduction

Use of Error Correcting Output Codes (ECOC) for decomposing a multi-class problem into a set of complementary two class problems is a well established method in many applications [1, 2, 4–13, 15–18]. When first suggested ECOC was based on the idea of using error-correcting codes as class labels, so that individual classification errors propagated from a set of binary classifiers can potentially be corrected [4]. For a two-class problem, classification errors can be one of two types, either predicted class w_1 for target class w_2 or predicted class w_2 for target class w_1.

In the ECOC method, a $k \times b$ binary code word matrix Z has one row (code word) for each of k classes, with each column defining one of b sub-problems that use a different labelling. Specifically, for the jth sub-problem, a training pattern with target class w_i $(i = 1...k)$ is re-labelled as class w_1 if $Z_{ij} = x$ and as class w_2 if $Z_{ij} = \bar{x}$ (where x is a binary variable, typically zero or one). One way of looking at the re-labelling is to consider the k classes as being arranged into two super-classes. The original ECOC combining strategy uses a simple distance measure (L1 norm) which is calculated with respect to real-valued classifier outputs to determine the closest code word and assigns a test pattern accordingly. If the code word matrix satisfies suitable constraints, this strategy is identical to the Bayesian decision rule [17, 16]. A problem with imposing constraints on code

words is that the generation process becomes very complex, but fortunately these constraints are approximated by random codes providing b is large enough [9].

Despite improvements in generalisation for many problems that have been reported for ECOC, there is some discussion as to why it works well. A long random code appears to perform as well or better than a code designed for its error-correcting properties [9]. Attempts have been made to develop a theory for ensemble classifiers in terms of bias/variance and margin [8], but so far these ideas have not provided a convincing explaination for ECOC. A practical approach to determining source of effectiveness of ECOC is to look at variants of the ECOC strategy to see how they perform. This is also useful if we want to extend ECOC to deal with applications for which it would be desirable to understand ECOC features as estimation measures.

In this paper we look at an alternative ECOC combination strategy based on Least Squares (LS-ECOC), which was investigated in [11] and extended by incorporating ridged regression when b is small [8]. Recovering individual class probabilities from super-class probabilities is easily accomplished by matrix inversion when the individual probability estimates are exact and columns of ECOC matrix are arranged in "one-per-class" structure. In practice, estimates are not perfect and a natural choice for attempting to recover probabilities is Least Squares. However the effect of the code on performance of LS-ECOC has not been investigated in the way that it has for ECOC.

In Sect. 2 we determine, for Least Squares combining, the required form of the ECOC matrix such that errors in super-class probabilities (local experts) and individual class probabilities are jointly minimised. In Sect. 3 we find the relationship between final variance and the variance of expert's error as a function of number of columns b and distance between rows of ECOC matrix for equidistance codes. Experimental results in Sect. 4 demonstrate the effect of code selection on classification performance in comparison with original distance-based approach.

2 ECOC and LS-ECOC

Decomposition of a multi-class classification problem into binary sub-problems in ECOC can be interpreted as a transformation between spaces from the original output q to p, given in matrix form by

$$p = Z^T.q \tag{1}$$

Having the estimation of posterior probability $\hat{p_j}$ of super-classes (provided by jth expert), this matrix equation can be solved to find an estimation of class membership probabilities $\hat{q}$. However, Z^T is not a square matrix in general, and so does not have an inverse. Furthermore base classifiers will not produce correct probabilities, and the error can be represented by

$$\hat{p_j} = \sum_{i=1}^{k} Z_{ij}.q_i + \epsilon_{p_j} \qquad j = 1...b \tag{2}$$

A natural unbiased solution to equation (1) is based on using the method of least squares which means finding $\hat{q}$ which minimises a cost function such as

$$R_p = \sum_{j=1}^{b} \epsilon_{\hat{p}_j}^2 = \sum_{j=1}^{b} (\hat{p}_j - p_j)^2 = \sum_{j=1}^{b} (\hat{p}_j - \sum_{i=1}^{k} Z_{ij}.q_i)^2 \tag{3}$$

The optimum point is given by

$$q^* = (Z.Z^T)^{-1}.Z.\hat{p} \tag{4}$$

For the solution of equation (4) to exist, ZZ^T must be non-singular. If all elements of the ith row in Z are zero ($z_{il} = 0 \ \ for \ \ any \ \ l$), or if two rows (or columns) are equal, ZZ^T is singular. Also these conditions are not meaningful for the decomposition, so when the code is generated we make sure that they do not occur. In summary, having a precise estimation of p (Bayesian binary experts), we will find q precisely, but in the presence of noise the sensitivity of solution to the code matrix Z could be important.

3 Error and Code Selection

Any Z satisfying equation (3) will minimise R_p, but we may like to find a Z that will also minimise the sum square error of q (R_q). Now from (3)

$$R_p = \hat{p}^T\hat{p} - 2\hat{p}^T.p + p^Tp \tag{5}$$

and using equation (1)

$$R_p = \hat{q}^T.ZZ^T.\hat{q} - 2\hat{q}^T.ZZ^T.q + q^T.ZZ^T.q \tag{6}$$

If we let $ZZ^T = m.I$, where I is the identity matrix and m a positive integer

$$R_p = m.I.(\hat{q}^T.\hat{q} - 2\hat{q}^T.q + q^T.q) = m.I.R_q$$

However this corresponds to the one-per-class case since it implies that $Z = I$, which means Z has no error-correcting capability.

Consider the case that ZZ^T can be written in the form

$$ZZ^T = \begin{bmatrix} n & m & \cdots & m \\ m & n & \cdots & m \\ \cdots & \cdots & \cdots & \cdots \\ m & m & \cdots & n \end{bmatrix} \tag{7}$$

Using the fact that $ZZ^T.q$ is a vector whose elements can be written in the form $(n-m)q_i + m$ equation (6) can be written as

$$R_p = -2(n-m).\hat{q}^T\hat{q} + m - 2(n-m).\hat{q}^Tq - 2m(n-m).q^Tq + m - 2\hat{q}^T.ZZ^T.q$$

so that

$$R_p = (n-m).R_q \tag{8}$$

Therefore from equation (8), if Z is in the form given by equation (7), both R_q and R_p can be minimised simultaneously.

3.1 Equi-distance Code

Furthermore, consider the situation that Z is an equi-distance code, so that $\sum_{l=1}^{b} |Z_{il} - Z_{jl}| = 2d$ for any pair i, j. Since Hamming Distance between pair i, j is the sum of the number of ones in row i and row j minus number of common ones between i, j we may write

$$\sum_{l=1}^{b} Z_{il} + \sum_{l=1}^{b} Z_{jl} - 2\sum_{l=1}^{b} Z_{il}.Z_{jl} = 2d \tag{9}$$

similar equations can be written for pair i, k and pair j, k

$$\sum_{l=1}^{b} Z_{il} + \sum_{l=1}^{b} Z_{kl} - 2\sum_{l=1}^{b} Z_{il}.Z_{kl} = 2d \tag{10}$$

$$\sum_{l=1}^{b} Z_{jl} + \sum_{l=1}^{b} Z_{kl} - 2\sum_{l=1}^{b} Z_{jl}.Z_{kl} = 2d \tag{11}$$

From equations (9), (10),(11) after re-arranging

$$\sum_{l=1}^{b} Z_{il}.Z_{jl} = \sum_{l=1}^{b} Z_{kl}.Z_{jl} = \sum_{l=1}^{b} Z_{kl}.Z_{il} = m \tag{12}$$

where m is number of common bits in code word, and

$$\sum_{l=1}^{b} Z_{il} = \sum_{l=1}^{b} Z_{kl} = \sum_{l=1}^{b} Z_{jl} = n \tag{13}$$

where n is the number of ones in each row

Therefore if Z is an equi-distance matrix, the number of ones in different rows are the same, and the number of common ones between any pair of rows is equal. But a matrix Z of the form satisfying (7) will have the property of equation (13) and (12) and will minimise both R_p and R_q simultaneously, since

$$ZZ^T = \sum_{l=1}^{b} Z_{il} Z_{lj}^T = \sum_{l=1}^{b} Z_{il} Z_{jl} = \begin{cases} n & \text{if i=j} \\ m & \text{otherwise} \end{cases}$$

3.2 Variance and Bias

For ZZ^T of the form (7) the inverse is given by

$$C = (ZZ^T)^{-1} = \begin{bmatrix} c_1 & c_2 & \cdots & c_2 \\ c_2 & c_1 & \cdots & c_2 \\ \cdots & \cdots & \cdots & \cdots \\ c_2 & c_2 & \cdots & c_1 \end{bmatrix} \tag{14}$$

where c_1 and c_2 can be expressed in terms of m, n, k

$$c_1 = \frac{n + (k-1).m}{n^2 + (k-2).m.n - (k-1).m^2} \tag{15}$$

$$c_2 = \frac{-m}{n^2 + (k-2).m.n - (k-1).m^2} \tag{16}$$

From equation (4)

$$\hat{q} = \begin{bmatrix} c_1 & c_2 & \cdots & c_2 \\ c_2 & c_1 & \cdots & c_2 \\ \cdots & \cdots & \cdots & \cdots \\ c_2 & c_2 & \cdots & c_1 \end{bmatrix} Z \begin{bmatrix} \hat{p}_1 \\ \hat{p}_2 \\ \cdot\cdot \\ \hat{p}_b \end{bmatrix}$$

We assume that individual classifers have same variance of error σ_p, and that the covariance of expert's error between any pair is simply $\rho.\sigma_p$, Then from using equation (13), the final variance can be written

$$\sigma_q = (c_1 - (k-1)c_2)^2.n.\sigma_p(1 + (n-1)\rho) \tag{17}$$

From equations (15) and (16), and knowing that $d = n - m$ (equation (9) and that for any row of an equi-distance code $b = m + n$, equation (17) can be written as

$$\sigma_q = \frac{(b-d)^3(1 + (b-d-1)\rho)}{((1-2k)d^2 + kbd)^2}\sigma_p \tag{18}$$

Equation (18) tells us that final variance increases with correlation among experts. Although (18) is not a simple formula, with some simplification we can understand how d and b affect σ_q. If we consider the case of $\rho = 0$ for simplicity

$$\sigma_q = \frac{n^3}{(knd - (k-1)d^2)^2}.\sigma_p$$

so that σ_q increases with n, for fixed d. Also σ_q is reduced if d is increased for fixed n. In other words if we use longer words so that b is increased then m should also be increased to keep n fixed.

To determine effect of bias, suppose that local experts provide $\hat{p} + \delta$ where δ is the bias. From equation (3), R_p with bias is given by

$$\sum_{j=1}^{b}(\hat{p}_j + \delta - p_j)^2 = \sum_{j=1}^{b}(\hat{p}_j^2 + \delta^2 + p_j^2 + 2\hat{p}_j\delta - 2p_j\delta)$$

In most applications $\delta^2 \simeq 0$, and if $\hat{p}$ is an acceptable estimation, $\delta(\hat{p} - p)$ is small. Therefore R_p is not sensitive to bias.

4 Experimental Results

4.1 Artificial Data

We test our ideas on an artificial benchmark in which we can find the result of Bayesian classifier as reference and visualise the decision boundaries to show the behaviour of ECOC. It is helpful for understanding the behaviour of composite system in mimicking the Bayesian classifier.

Consider five groups of two dimensional random vectors having normal distribution as: $p(x|c_i) = \frac{1}{2\pi\sigma_i^2} exp[\frac{||x-\mu_i||^2}{-2\sigma_i^2}] \quad for \quad i = 1, 2, ..., 5$ with parameters given in table 1.

Table 1. Distribution parameters of data used in artificial benchmark

class	c_1	c_2	c_3	c_4	c_5
μ_i(mean)	[0,0]	[3,0]	[0,5]	[7,0]	[0,9]
σ_i^2(variance)	1	4	9	25	64

Having a set of patterns consisting of equal number of patterns from each group, our goal is to classify them. Our base classifiers are not made by training, but using the parameters from table 1 we will just find the posterior probability of class (or super-class) membership for each sample. Using equal number of patterns from each group for test set(equal prior probability for classes); Bayesian decision rule says: $assign \;\; x \rightarrow w_i \;\; if \;\; P(w_i|x) = ArgMax_i(P(c_i|x))$. $P(c_i|x)$ is the posterior probability of class membership for class c_i, and can be found by the Bayesian formula: $P(c_i|x) = \frac{p(x|c_i)P(c_i)}{p(x)}$, in which $P(c_i)$ is the prior probability of class i and $p(x)$ is the same for all classes. So the decision rule can be changed: $assign \;\; x \rightarrow c_i \;\; if \;\; p(x|c_i) = ArgMax_i(p(x|c))$

To simulate the behaviour of the system, Gaussian and uniform random data are added to the output of experts. For a fair comparison between different methods, the noise for each code matrix is produced once and used in all combining methods. To find a code with desired properties, we have used BCH method[14], followed by selecting rows using properties (12) and (13). Columns with all zeros or ones have been removed, as explained in Sect. 3.

The following code matrices are used in this experiment ($k = 5$):
C1: a $k \times k$ unitary code(one per class)
C2: a $k \times 7$ matrix with randomly chosen binary elements
C3: a $k \times 7$ BCH code (minimum distance of 3, non-equal)
C4: a $k \times 7$ BCH code with equal distance of 4
C5: a $k \times 15$ matrix with randomly chosen elements
C6: a $k \times 15$ BCH code with equal distance of 8
C7: a $k \times 31$ BCH code with equal distance of 16

Adding Gaussian noise with variance of 0.5 and zero bias, the classification rate of the Bayesian classifier is 71.82% and with zero variance and 0.5 bias it is 72.08%, The rates of matching (representing how close the Bayes rate is approximated) for original ECOC and LS-ECOC are presented in table 2.

Table 2. matching rate (% Bayesian) for ECOC and LS-ECOC with added noise

Code	Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6
C1	100.00	100.00	57.66	57.74	100.00	100.00
C2	97.56	100.00	65.74	59.98	53.88	86.88
C3	97.30	100.00	66.40	60.30	83.06	81.52
C4	100.00	100.00	69.04	69.04	98.64	98.64
C5	97.08	100.00	78.72	76.02	87.50	88.30
C6	100.00	100.00	82.78	82.78	100.00	100.00
C7	100.00	100.00	89.50	89.50	100.00	100.00

Exp 1: ECOC with no noise
Exp 2: LS-ECOC with no noise
Exp 3: ECOC with Gausian noise (Bias=0, Variance=.5)
Exp 4: LS-ECOC with Gausian noise (Bias=0, Variance=.5)
Exp 5: ECOC with Gausian noise (Bias=.5, Variance=0)
Exp 6: LS-ECOC with Gausian noise (Bias=.5, Variance=0)

From table 2:

1. Without noise, the performance of LS-ECOC for codes with unequal distance between rows is better than ECOC (Exp 1 and 2).
2. In noisy data
   (a) For equi-distance codes(C1,C4,C6,C7) original ECOC and LS -ECOC have similar performance (Exp. 3,4 and 5,6).
   (b) For codes with unequal distance, ECOC is better in variance reduction (Exp. 3 and 4) while LS-ECOC has better performance for added bias (Exp. 5 and 6). It seems reasonable that for added bias the distance measurement in ECOC is adversely affected since the number of ones in code word labels is different. On the other hand, LS-ECOC is less sensitive to bias as predicted in Sect. 3.
3. For longer random codes (C5), it can be expected that on average the number of ones in rows is similar and therefore there will be less difference in the ability of ECOC and LS-ECOC in handling bias and variance (Exp. 3,4 and 5,6).

4.2 Real Data

We tested Codes C1-C6 on real data for problems from [3] The base classifier is an MLP trained by BackPropagation with fixed learning rate, momentum and

number of training epochs. The number of hidden nodes of MLP, number of training and test patterns and number of classes for the problems are shown in table 3. We also compared ECOC and LS-ECOC with Centroid-ECOC [8], which is identical to ECOC except distance is calculated to centroid of classes rather than to code word label. The mean and standard deviation of classification rates for ten independent runs are given in tables 4 and 5.

From tables 4 and 5:

1. The combining srategy (ECOC, Cent-ECOC, LS-ECOC) appears to have little impact, except for codes C2, C3 with LS.
2. In all datasets for 7-bit code, equi-distant is best (C2,C3,C4).
3. Longer codes perform better. However for the 15-bit code, random is better for two datasets, while equidistant is better for the other two.

Table 3. Specification of problems, showing nunber of problems, number of train and test patterns, and number of MLP hidden nodes

Database	Class (Num)	Train (Num)	Test (Num)	Nodes (Num)
zoo	7	50	51	1
car	4	50	1678	1
vehicle	4	350	496	5
satellite	6	1000	5435	2

Table 4. Mean and Std classification rate for ECOC, LS-ECOC and Centroid-ECOC on zoo and car data base.

code	ECOC(zoo)	Cent(zoo)	Lsqu(zoo)	ECOC(car)	Cent(car)	Lsqu(car)
C1	89.54	89.54	89.54	72.15	72.15	72.15
	5.99	5.99	5.99	4.83	4.83	4.83
C2	77.78	77.78	43.14	73.60	73.60	72.63
	13.91	13.91	6.79	0.93	0.93	1.21
C3	88.89	88.89	84.97	72.96	72.96	71.99
	6.30	6.30	2.26	3.62	3.62	2.79
C4	86.27	86.27	86.27	74.16	74.16	74.16
	8.98	8.98	8.98	3.70	3.70	3.70
C5	94.77	94.77	94.12	74.33	74.33	74.55
	2.99	2.99	3.39	2.35	2.35	2.39
C6	93.46	93.46	93.46	72.79	72.79	72.79
	2.99	2.99	2.99	2.65	2.65	2.65

Table 5. Mean and Std classification rate for ECOC, LS-ECOC and Centroid-ECOC on vehicle and satellite data base.

code	ECOC(veh)	Cent(veh)	Lsqu(veh)	ECOC(sat)	Cent(sat)	Lsqu(sat)
C1	62.77	62.77	62.77	65.05	65.05	65.05
	8.65	8.65	8.65	17.29	17.29	17.29
C2	66.94	66.94	61.22	80.29	80.29	23.91
	4.42	4.42	5.54	6.915	6.915	2.30
C3	53.02	53.02	57.12	70.06	70.06	62.67
	14.90	14.90	15.33	10.42	10.42	6.96
C4	69.15	69.15	69.15	69.48	69.48	69.48
	5.62	5.62	5.62	3.88	3.88	3.88
C5	73.32	73.32	73.72	77.74	77.74	77.74
	2.78	2.78	3.82	4.31	4.31	4.98
C6	75.34	75.34	75.34	80.43	80.43	80.43
	1.81	1.81	1.81	1.73	1.73	1.73

5 Discussion and Conclusion

We have demonstrated theoretically and practically that LS-ECOC used with equi-distant code words may give better performance, at least for shorter codes. However as length of code word was increased no performance advantage was apparent when comparing ECOC with LS-ECOC. Results on real data confirmed that any theoretical advantage of LS-ECOC is not necessarily realised in practice if longer codes are used. Comparison of three combining strategies ECOC, LS-ECOC and Centroid-ECOC suggest that the combination strategy does not play a major role in improving performance. This result lends support to the finding of others [9] that the error-correcting capability of a designed code may not be a significant aspect of the ECOC method, at least with respect to the combining strategies considered here.

In order to apply ECOC to situations where super-class probabilities are not suitable measures by themselves, we conclude that it may be useful to look at variants of ECOC. Least Squares represents an alternative combining strategy for ECOC that can give comparable classification results to the original distance-based strategy. If individual class probabilities are required, LS-ECOC provides a method of recovering them.

References

1. E. Alpaydin and E. Mayoraz. Learning error-correcting output codes from data. In *Proceeding of ICANN'99*, Edinburgh, U.K., September 1999. //www.cmpe.boun.edu.tr/ ethem/.
2. A. Berger. Error-correcting output coding for text classification. In *Proccedings of IJCAI'99*, Stockholm, Sweden, 1999. http://proxy3.nj.nec.com/did/8956.

3. C.L. Blake and C.J. Merz. UCI repository of machine learning databases. University of California, Irvine, Dept. of Information and Computer Sciences, 1998. http://www.ics.uci.edu/~mlearn/MLRepository.html.
4. T.G Dietterich and G. Bakiri. Error-correcting output codes: A general method for improving multiclass inductive learning programs. In *Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI-91)*, pages 572–577. AAAI Press, 1991.
5. T.G. Dietterich and G Bakiri. Solving multi-class learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, 2:263–286, 1995.
6. R. Ghaderi and T. Windeatt. Circular ecoc, a theoretical and experimental analysis. In *International Conference of Pattern Recognition(ICPR2000)*, pages 203–206, Barcelona,Spain, September 2000.
7. R. Ghaderi and T Windeatt. Viewpoints of error correcting output coding in classification task. In *The 7th Electrical and electronic Engineering seminar of Iranian students in Europ.*, Manchester U.K, May 2000.
8. G. James. *Majority Vote Classifiers: Theory and Applications.* PhD thesis, Dept. of Statistics, Univ. of Stanford, May 1998. http://www-stat.stanford.edu/ gareth/.
9. G. James and T. Hastie. The error coding method and PICT's. *Computational and Graphical Statistics*, 7:377–387, 1998.
10. E.B. Kong and T.G. Diettrich. Error-correcting output coding correct bias and variance. In *12th Int. Conf. of Machine Learning*, pages 313–321, San Fransisco, 1995. Morgan Kaufmann.
11. E.B. Kong and T.G. Diettrich. Probability estimation via error-correcting output coding. In *Int. Conf. of Artificial Inteligence and soft computing*, Banff,Canada, 1997. http://www.cs.orst.edu/ tgd/cv/pubs.html.
12. F. Leisch and K. Hornik. Combining neural networks voting classifiers and error correcting output codes. In I. Frolla and A. Plakove, editors, *MEASURMENT 97*, pages 266–269, Smolenice, Slovakia, May 1997.
13. F Masulli and G Valentini. Effectiveness of error correcting output codes in multiclass learning problems. In J.Kittler and F.Roli, editors, *Multiple Classifier Systems, MCS2000*, pages 107–116, Cagliari, Italy, 2000. Springer Lecture Notes in Computer Science.
14. W.W. Peterson and JR. Weldon. *Error-Correcting Codes.* MIT press, Cambridge,MA, 1972.
15. R.E. Schapire. Using output codes to boost multiclass learning problems. In *14th International Conf. on Machine Learning*, pages 313–321. Morgan Kaufman, 1997.
16. T. Windeatt and R. Ghaderi. Binary codes for multi-class decision combining. In *14th Annual International Conference of Society of Photo-Optical Instrumentation Engineers (SPIE)*, volume 4051, pages 23–34, Florida,USA, April 2000.
17. T. Windeatt and R. Ghaderi. Multi-class learning and ecoc sensitivity. *Electronics Letters*, 36(19), September 2000.
18. T. Windeatt and R. Ghaderi. Binary labelling and decision level fusion. *Information fusion, to be published*, 2001.