Available online at www.sciencedirect.com
SCIENCE DIRECT
ELSEVIER
Neurocomputing 69 (2006) 679–685
NEUROCOMPUTING
www.elsevier.com/locate/neucom

Boosting by weighting critical and erroneous samples ☆

Vanessa Gómez-Verdejo*, Manuel Ortega-Moral, Jerónimo Arenas-García, Aníbal R. Figueiras-Vidal

Department of Signal Theory and Communications, Universidad Carlos III de Madrid, Avda. Universidad 30, 28911 Leganés (Madrid), Spain

Available online 20 January 2006

Abstract

Real Adaboost is a well-known and good performance boosting method used to build machine ensembles for classification. Considering that its emphasis function can be decomposed in two factors that pay separated attention to sample errors and to their proximity to the classification border, a generalized emphasis function that combines both components by means of a selectable parameter, λ, is presented. Experiments show that simple methods of selecting λ frequently offer better performance and smaller ensembles.

Keywords: Boosting; Real adaboost; Emphasis functions; Convex combination

1. Introduction

To combine artificial neural networks (ANN) is an avenue to get easier designs and better performance, as well as to a clearer understanding of how the resulting machines work. These reasons have fueled an increasing interest in ANN ensembles, following the denomination introduced by Hansen and Salomon [14]. Overviews of procedures to construct ensembles can be found in [15,25]. Among all combination procedures, boosting schemes [7,8], and in particular Real Adaboost (RA) [24], have demonstrated excellent performance. Although boosting designs have their roots in the consideration of weak learners [23] under the light of the Probably Approximately Correct learning theory [27], the alternative point of view first suggested in [3] and further explored in [2] opens possibilities to extend boosting mechanisms in forms different from a direct emphasis of erroneous samples. This alternative perspective says that, in classification problems, boosting progressively concentrates its attention in samples nearer and nearer to the classification border. Additionally, Breiman [5] shows that the particular mode of emphasizing erroneous samples is not essential. So, since there were previous evidences of the effectiveness of paying attention to samples lying near the classification borders [11,26], which we will call "critical" samples in the sequel, the idea of using both error and proximity to the border to emphasize sample populations emerges. From a conceptual point of view, this idea is not strictly new, since the implicit cost function and the regularization term corresponding to the Maximum Separation Margin (MSM) principle [6] exploited in Support Vector Machines (SVM) and Kernel-based Classifier designs represents a very similar approach. Then, it is not surprising that using MSM elements [20] plus regularization schemes [19] to construct boosting ensembles had lead to improved performance designs. All the above justifies further work with the same basic orientation.

In this paper, we will work starting from the RA algorithm, that, besides providing an excellent general performance in solving decision problems, allows to introduce a version of the proposed new emphasis in a

☆ Expanded version of a communication presented at the European Symposium on Artificial Neural Networks in Bruges (Belgium), April 2005. This work has been partly supported by grant CICYT TIC2002-03713 and the work of V. Gómez-Verdejo was also supported by the Chamber of Madrid Community and European Social Fund by a scholarship.

* Corresponding author. Tel.: +34 916248759.
E-mail addresses: vanessa@tsc.uc3m.es (V. Gómez-Verdejo), ortegam@tsc.uc3m.es (M. Ortega-Moral), jarenas@tsc.uc3m.es (J. Arenas-García), arfv@tsc.uc3m.es (A.R. Figueiras-Vidal).

0925-2312/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2005.12.011

principled form. After proving that the traditional RA emphasis function can be seen as the product of two factors, the first depending on the quadratic error of the sample, and the second being a function of the "proximity" of the sample to the classification border (measured as the absolute value of the output of the already constructed portion of the ensemble), the possibility of getting advantage of applying the same kind of factors, but as a combination governed by a selectable parameter, is explored. Experimental results support the interest of this possibility.

Although we will deal here with RA, it is worth to mention that the idea of using combined emphasis functions, even of different types, can be easily applied to other boosting methods (Least-Square-Boost [9], Logistic Regression [10], or algorithms that aim to maximize the margin [20,21]).

The rest of the paper is structured as follows. In the next section the classical RA algorithm will be described, so that it can be easily linked with the emphasis function proposed in Section 3. In Section 4, we show the importance of a good emphasis selection comparing classical RA with schemes adopting the combined emphasis in some benchmark problems. Finally, in Section 5, conclusions and future research lines will be presented.

2. A revision of Real Adaboost

Suppose we are given a training data set $\{(\mathbf{x}_i, d_i) \in \chi \times \{-1, 1\}, i = 1, \ldots, l\}$: to build up an RA classifier, iteratively, at each round $t = 1, \ldots, T$, a new base learner is trained with a resampled population from χ, which implements a function $o_t(\mathbf{x}_i) : \chi \longrightarrow [-1, 1]$ (this real interval is a characteristic of RA). An output weight, α_t, is then assigned to the learner o_t, and it is added to the ensemble. In this way, the overall output of the ensemble in round Tth, $f_T(\mathbf{x}_i)$, is calculated as the linear combination of all learners

$$f_T(\mathbf{x}_i) = \sum_{t=1}^{T} \alpha_t o_t(\mathbf{x}_i). \tag{1}$$

To resample the population, an emphasis function $D_t(i)$ is employed. This function indicates the importance that the tth learner o_t has to assign to the pattern $\mathbf{x}_i$. For the first learner, all patterns have the same importance, $D_1(i) = 1/l$, $\forall i = 1, \ldots, l$, and in each round these weights are updated according to

$$D_{t+1}(i) = \frac{D_t(i) \exp[-\alpha_t o_t(\mathbf{x}_i) d_i]}{Z_t} \tag{2}$$

where Z_t is a normalization factor assuring that $\sum_{i=1}^{l} D_t(i) = 1$. This updating rule causes the patterns that achieved a higher error in the previous rounds to have more relevance during the training of the next learner.

To obtain the output weights, training error bound

$$E_{\text{train}} = \sum_{i=1}^{l} |\text{sign}[f_t(\mathbf{x}_i)] \neq d_i| \leqslant \sum_{i=1}^{l} \exp[-f_t(\mathbf{x}_i) d_i] \tag{3}$$

is minimized. As Schapire and Singer showed in [24], the minimization of (3) results in the following analytic expression to calculate the new α_t value at each round

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1 + r_t}{1 - r_t}\right), \tag{4}$$

where r_t is calculated according to

$$r_t = \sum_{i=1}^{l} D_t(i) o_t(\mathbf{x}_i) d_i. \tag{5}$$

To obtain the output coming from each weak learner, $o_t(\mathbf{x})$, it will be desirable maximize the gradient of training error bound (3), so that a maximum reduction of this bound can be achieved. In fact, the maximization of this gradient directly corresponds to maximizing r_t (see [18,24] for a more detailed explanation). However, expression (5) is not convex with respect to o_t, and in many cases it is not possible to maximize it. Thus, it is usually easier to train each base learner to minimize the mean square error weighted by the emphasis function D_t,

$$C_t = \sum_{i=1}^{l} D_t(i)[d_i - o_t(\mathbf{x}_i)]^2. \tag{6}$$

This cost function is a regularized version of (5) and maximizes it indirectly since, expanding (6)

$$\begin{aligned} C_t &= \sum_{i=1}^{l} D_t(i)(d_i - o_t(\mathbf{x}_i))^2 \\ &= \sum_{i=1}^{l} D_t(i)(d_i^2 + o_t^2(\mathbf{x}_i) - 2 o_t(\mathbf{x}_i) d_i) \\ &= \sum_{i=1}^{l} D_t(i) d_i^2 + \sum_{i=1}^{l} D_t(i) o_t^2(\mathbf{x}_i) - 2 \sum_{i=1}^{l} D_t(i) o_t(\mathbf{x}_i) d_i \end{aligned} \tag{7}$$

and considering that $\sum_{i=1}^{l} D_t(i) d_i^2 = 1$, cost function C_t can be expressed as

$$C_t = \sum_{i=1}^{l} D_t(i) o_t^2(\mathbf{x}_i) - 2 \sum_{i=1}^{l} D_t(i) o_t(\mathbf{x}_i) d_i, \tag{8}$$

where the first term has a regularization role (it penalizes wide output ranges of o_t), and minimizing the second term is equivalent to maximizing (5).

Finally, we will show that emphasis function (2) not only pays attention to the error of each pattern but also to its proximity to the boundary, following our presentation in [13]. First, we rewrite the expression of $D_{t+1}(i)$ in the $t + 1$th round as a function of global output $f_t(\mathbf{x}_i)$

in the previous round

$$D_{t+1}(i) = \frac{D_t(i)\exp[-\alpha_t d_i o_t(\mathbf{x}_i)]}{Z_t} = \frac{\prod_{n=1}^{t}\exp[-\alpha_n d_i o_n(\mathbf{x}_i)]}{l\prod_{n=1}^{t} Z_n} = \frac{\exp[\sum_{n=1}^{t} -\alpha_n d_i o_n(\mathbf{x}_i)]}{Z'_t} = \frac{\exp(-f_t(\mathbf{x}_i)d_i)}{Z'_t}, \quad (9)$$

where Z'_t is a normalization factor. Second, if we take into account that

$$-2f(\mathbf{x}_i)d_i = [f(\mathbf{x}_i) - d_i]^2 - f(\mathbf{x}_i)^2 - d_i^2$$

and

$$d_i^2 = 1 \quad \forall i = 1, \ldots, l,$$

we get

$$D_{t+1}(i) = \frac{1}{Z'_t}\exp\left(-\frac{1}{2}\right)\exp\left\{\frac{[f_t(\mathbf{x}_i) - d_i]^2}{2}\right\}\exp\left[-\frac{f_t^2(\mathbf{x}_i)}{2}\right]. \quad (10)$$

Thus, the RA emphasis function is really composed of two different emphasis terms. The first term focuses on the patterns that have presented a highest quadratic error at the previous round. The second term pays attention to the "critical" patterns, those close to the "current" boundary, this boundary being the solution of $f_t(\mathbf{x}_i) = 0$ (the classification boundary given by the ensemble up to the previous round).

3. Boosting by weighting boundary and erroneous samples

In the light of (10), one may wonder if this fixed combination of emphasis terms is optimal in all situations. To answer this question, we study the effect of applying an emphasis function that uses a convex combination of the error and the "proximity" terms by means of a weighting parameter λ ($0 \leqslant \lambda \leqslant 1$):

$$D_{\lambda,t+1}(i) = \frac{1}{Z_t}\exp\{\lambda[(f_t(\mathbf{x}_i) - d_i]^2 - (1-\lambda)f_t^2(\mathbf{x}_i)\}. \quad (11)$$

This flexible formulation allows us to choose how much to consider the "proximity" to the boundary or the quadratic error of each sample, by selecting different values λ. Three particular cases are relevant

- $\lambda = 0$: we focus on "critical" patterns because only the "proximity" to the boundary is taken into account,

$$D_{\lambda=0,t+1}(i) = \frac{1}{Z_t}\exp\left[-f_t^2(\mathbf{x}_i)\right]. \quad (12)$$

- $\lambda = 0.5$: we get the classical RA emphasis function,

$$D_{\lambda=0.5,t+1}(i) = \frac{1}{Z_t}\exp\left\{\frac{[f_t(\mathbf{x}_i) - d_i]^2}{2} - \frac{f_t^2(\mathbf{x}_i)}{2}\right\}. \quad (13)$$

- $\lambda = 1$: the emphasis function only pays attention to the quadratic error of each pattern,

$$D_{\lambda=1,t+1}(i) = \frac{1}{Z_t}\exp[f_t(\mathbf{x}_i) - d_i]^2. \quad (14)$$

To build a boosted ensemble that employs emphasis function (11) we need an efficient way to combine all base learner outputs, i.e., to select the output weights α_t. In order to be able to compare the effectiveness of using different tradeoffs of the emphasis terms with the fixed tradeoff that the classical RA algorithm uses, we continue to minimize training error bound (3). In this way, independently of the weighting parameter value, the output weights will be obtained according to (4), and r_t will be calculated with a general expression independent of the emphasis function employed,

$$r_t = \sum_{i=1}^{l} o_t(\mathbf{x}_i)d_i\frac{\exp[-f_{t-1}(\mathbf{x}_i)d_i]}{Z_{t-1}}. \quad (15)$$

Note that this new formulation of r_t is equivalent to (5) when $\lambda = 0.5$, since $D_{\lambda=0.5,t}(i) = \exp[-f_{t-1}(\mathbf{x}_i)d_i]/Z_{t-1}$.

In the following, and for reasons of clarity, we will indicate classical RA as RA-se (RA with standard emphasis), and our proposed scheme as RA-we (RA with weighted emphasis).

4. Experiments

In this section, we will evaluate the potential advantages of applying RA-we with respect to RA-se, as well as the effectiveness of using a cross-validation process to select mixing parameter λ for emphasis function (11). The corresponding experiments will consider (selectively small) Multi-Layer Perceptrons (MLPs) as ensemble building "weak learners" elements because other kind of architectures such as Radial Basis Functions networks used as "weak learners" offer similar basic results [12].

We select a series of well-known binary problems for carrying out the evaluation: *Phoneme* (Ph in tables) from [1]; from [4], *Abalone* (Ab) (a multiclass case converted to a binary problem according to [22]), *Contraceptive* (Co), *Image* (Im), *Spam* (Sp) and *Tictactoe* (Ti); and synthetic problem *Kwok* (Kw) from [16]. Table 1 summarizes the main features of these problems, and it also shows the recognized record error rate and the machine that offers it (or, for the synthetic problem, the Bayesian solution). We use the predefined test set for each problem if given, and, if not, we have tested the performance of the classifiers using ten random partitions with 40% of the data.

Number of M hidden units of the MLP elements has been selected in two different forms:

- First, as the value corresponding to pre-existing RA-se designs that offer record performances. In this way, RA-we is competing against the best RA-se ensemble in non-

Table 1
Characteristics of the benchmark problems. dim: number of dimensions; n_1/n_{-1}: number of samples of each class

Problem	dim	Train samples (n_1/n_{-1})	Test samples (n_1/n_{-1})	SVM error rate	Record error rate
Ab	8	1238/1269	843/827	20.9	19.20 (RA-se)
Co	9	506/377	338/252	28.61	28.61 (SVM)
Im	18	821/1027	169/293	3.47	2.25 (RA-se)
Kw	2	300/200	6120/4080	11.74	11.3 (Bayes)
Ph	5	952/2291	634/1527	15.35	13.70 (RA-se)
Sp	57	1673/1088	1115/725	7.2	5.69 (RA-se)
Ti	9	199/376	133/250	1.7	1.47 (RA-se)

Table 2
(Percent) test errors (E) and ensemble sizes (T) achieved by RA-se, RA-we using cross-validation (CV), and an "omniscient" approach to select mixing parameter λ_0

	M	RA-se		RA-we with CV				RA-we "omniscient"			
		T	E	T	λ_{CV}	$E_{\lambda_{CV}}$	p	T	λ_0	E_{λ_0}	p
Ab	9	70	19.43	40	0.9	**19.40**	0.23	30	0.2	19.18	0
	6	60	19.20	50	0.9	**19.16***	0.54	40	0.2	19.00	0
Co	4	50	28.90	30	0.2	**28.61**	0.27	50	0	28.56	0.46
	3	60	29.20	20	0	**28.50***	0	20	0	28.50	0
Im	9	90	**2.25**	90	0.5	2.25	1	90	0.5	2.25	1
	2	130	**2.86**	130	0.4	2.89	0.58	130	0.5	2.86	1
Kw	9	80	11.68	70	0.4	**11.63**	0	70	0.4	11.63	0
	4	80	11.82	40	0.3	**11.72**	0	60	0.4	11.70	0
Ph	36	50	13.89	20	0	**13.59***	0.02	20	0.1	13.56	0
	28	20	13.70	20	0.1	**13.60**	0.5	20	0	13.52	0.19
Sp	5	80	**5.69**	60	0.6	5.73	0.99	80	0.5	5.69	1
	2	60	6.04	70	0.3	**6.03**	0.93	70	0.3	6.03	0.93
Ti	8	470	**1.47**	470	0.5	1.47	1	470	0.5	1.47	1
	2	830	8.12	800	0.3	**6.94**	0.03	800	0.3	6.94	0.03

favourable conditions. (These cases correspond to dark rows in Table 2).

- The second way of selecting M is as the (approximate) value that means there are 25 training samples for each MLP parameter.[1] Thus, MLP elements are truly "weak learners".

We must remark that, in all cases but one (*Tictactoe*: probably because its low amount of training samples), obtained performances are not very sensitive to M. To establish a better procedure to select MLP sizes will be reasonable when a really good method to select mixing parameter λ is available.

All MLP have random initial weights in $[-1, 1]$, and they are updated by means of a back-propagation algorithm with learning step 0.01 for both hidden and output layers so that cost function (6) is minimized. The back propagation-algorithm is run up to 100 epochs over a random partition of 80% of the training data set; the remaining 20% data is used to validate the solution, since we select the MLP weights of the epoch that has achieved the minimum error over this data partition.

Among the eleven values of λ obtained dividing $[0, 1]$ in ten equal intervals, "best" value λ_{CV} is selected by means of the following 5-fold cross validation: we randomly divide the training data set in 5 (approximately) equal size subsets, and the machine ensembles are trained 20 times with the five (effective) training sets obtained by separating each one of the above subsets and using it for validation, computing the ensemble performance. λ_{CV} is the value corresponding to the highest average performance.

With respect to ensemble size T, its optimum value varies according the problem, the number of hidden neurons of the learners, and λ, so, to assure the convergence of the ensemble in each case, we have stopped the ensemble building when the mean value of the output weights[2] is

[1] Actually, we are using 20 "effective" training samples for each MLP parameter, due to, as we will explain later, the MLP are trained with 80% of the total of training data.

[2] The experiments have been carried out with 50 independent runs for each algorithm, so the mean value of the output weights is calculated over these 50 runs.

practically equal to zero ($\alpha_t \approx 0.01$), because when α_t values are lower, new learners practically do not modify the ensemble performance.

Test results corresponding to all the eleven values of λ are also obtained. It is true that select λ_0 corresponding to the best performance is a perverse trick for designing purposes, but here this sort of "omniscient" approach is followed just as a form to evaluate if the previously described cross-validation process is successful in finding appropriate values for hyperparameter λ.

Table 2 presents the overall results averaged over 50 independent runs. It also includes the statistical significance of the difference of error rates $E_{\lambda_{CV}}$ and E_{λ_0} and those corresponding to RA-se, E, measured by parameter p of the Wilconxon Rank-sum or Mann–Whitney test [17]. This test considers two sample populations, ranks all their elements, and decides if these populations are or not statistically shifted by computing probability p of that the sum of ranks of the "first" population samples falls into the tails of the distribution of the whole population. When p is lower than 0.1 it means that there is a statistical difference, that disappears when p goes to 1. In our case, the populations are the 50 values we have for error rates $E_{\lambda_{CV}}$ (or E_{λ_0}) and E.

From Table 2, the following general comments emerge:

(a) In two cases (*Contraceptive*, $M = 3$, and *Phoneme*, $M = 36$), RA-we with cross-validation offers absolute performance records showing significant statistical difference with respect to RA-se. Nevertheless, the performance difference is small. In one more case (*Abalone*, $M = 6$) an absolute record without statistical relevance is obtained. Note that the RA-we cross-validation design beats the recognized SVM record for *Contraceptive*.

(b) In one more case (*Kwok*), RA-we with cross-validation gives a slight statistically significant improvement with respect to RA-se, although without approaching the theoretical limit.

(c) The rest of performance advantages or disadvantages of RA-we with cross-validation with respect to RA-se are not important nor statistically significant.

(d) An aspect in which RA-we with cross-validation offers a clear advantage is the number of elements composing the designed ensembles. There is an important size reduction in most the cases, with the exceptions of irrelevant case *Spam* $M = 2$, and ties in *Image* and in *Tictactoe*, $M = 8$.
This aspect can be a definitive reason for applying RA-we when massive on-line processing is required, such as in fraud detection or other security or safety tasks, since equivalent or slight better performances are obtained. Obviously, the design will need a higher-computational effort.
Besides the above facts:

(e) Cross-validation seems to be moderately successful in finding appropriate values for λ. Note that it would be possible to get a new absolute statistically significant (with respect to RA-se) record for *Abalone*, $M = 6$, with $\lambda = 0.2$ ($\lambda_{CV} = 0.9$), using a smaller ensemble, and that there are slight improvements for other cases (*Kwok*, $M = 4$, and *Phoneme*, $M = 36$).

This moderate success seems to indicate that it is convenient to explore how to select λ in a manner more related to the fundamental principles of RA (although this would be not a real requirement for applications in which a high number of training samples are available; that is not, obviously, the situation in the examples we have selected). We are currently working along this interesting complementary line. To support that there is a reasonable opportunity of getting improvements, let us present Fig. 1, that shows that there is an attractive flat region for the test error around $\lambda = 0.2$ just for *Abalone*, $M = 6$, and Fig. 2, that shows how

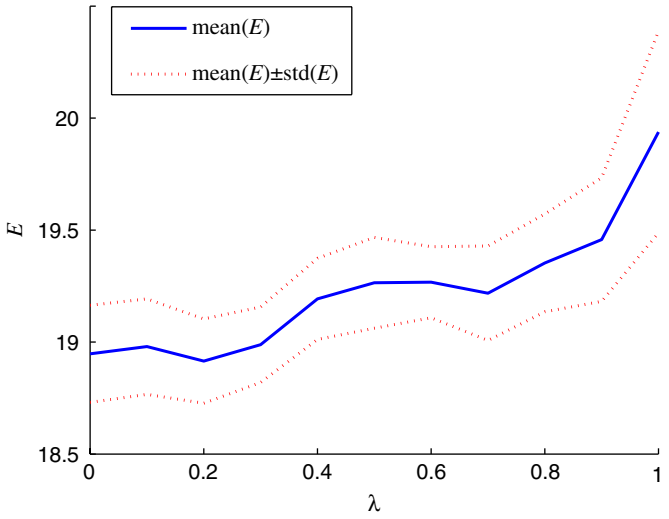

Fig. 1. Test error as a function of weighting parameter λ in *Abalone*, $M = 6$.

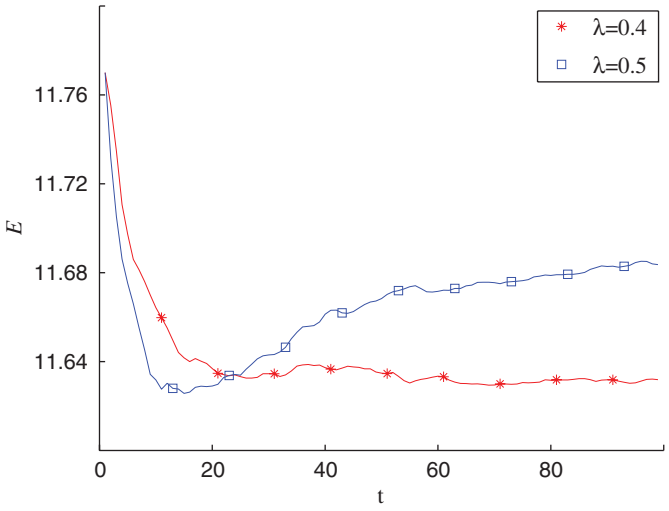

Fig. 2. RA-se overfitting problem for *Kwok*, $M = 9$.

the overfitting risk that appears when increasing T in *Kwok*, $M = 9$, when RA-se is applied ($\lambda = 0.5$), and that in fact reduces its performance when stopping when α_t is small enough, disappears when using RA-we with $\lambda = 0.4 (= \lambda_{CV})$.

To conclude this discussion, let us address the question of establishing what are the characteristics of the problems for which RA-we can offer advantage with respect to RA-se. An answer in general terms is well-known: mainly in those cases in which the desirable classification border depends heavily on samples near to that border. We know this is somewhat a "circular" answer, but its practical interpretation is easy: when there are many highly erroneous samples that are not important to define the border, such as in *Kwok*, that has a "rigid" sample structure; and, with a lower importance because their higher dimensionality, in *Abalone*, *Contraceptive*, and *Phoneme*. Just the problems in which "better results", in some sense, are obtained with RA-we. Needless to say, the presence of outliers will increase the possibility of getting practical advantage. In any case, let us remark that the above observations do not mean that it is impossible to get advantage in "clearer separated" classification problems, that may appear according to the specific shape of the classification border needed to get the best performance. Going further (considering overfitting and similar questions) will be reasonable when a better mechanism to select the mixing parameter be available.

5. Conclusions and future work

A new emphasis scheme, combining error size and proximity to the classification border of the training samples, has been presented for Real Adaboost ensemble designs, including the classical scheme as a particular case. It has been shown experimentally the potential advantage, both in classification performance and in ensemble size, of using such a mixed emphasis, even determining the mixing parameter by means a direct cross-validation process.

The limited capacity of cross-validation to find an appropriate value of the mixing parameter has also been verified. Consequently, looking for automatic methods to select a good value for this parameter, and mainly those principled in the fundamentals of Real Adaboost, appears as an interesting complementary research topic. We are currently following this direction.

References

[1] P. Alinat, Periodic Progress Report 4, ROARS Project ESPRIT II—Number 5516. Technical Thomson Report TS. ASM 93/S/EGS/NC/079, 1993.

[2] J. Arenas-García, A. Figueiras-Vidal, A.J.C. Sharkey, The beneficial effects of using multi net systems that focus on hard patterns, in: T. Windeatt, F. Rolli (Eds.), Multi Classifier Systems, Proceedings of the fourth International Workshop, Surrey, UK, Lecture Notes in Computer Science, Springer, Berlin, 2003, pp. 45–54.

[3] P.L. Bartlett, R.E. Schapire, Y. Freund, W.S. Lee, Boosting the margin: a new explanation for effectiviness of voting methods, Ann. Statist. 26 (5) (1998) 1651–1686.

[4] C.L. Blake, C.J. Merz, UCI Repository of machine learning databases, ⟨http://www.ics.uci.edu⟩. University of California, Irvine, Department of Information and Computer Sciences, 1998.

[5] L. Breiman, Multi-Net Systems, in: A.J.C. Sharkey (Ed.), Combining Artificial Neural Nets: Ensemble and Modular Multi-Nets Systems, Springer, London, UK, 1999, pp. 31–50.

[6] C.J.C. Burges, A tutorial on support vector machines for pattern recognition, Data Mining Knowledge Discovery 2 (2) (1998) 121–167.

[7] Y. Freund, R.E. Schapire, Experiments with a new boosting algorithm, in: Proceedings of the 13th International Conference on Machine Learning, Bari, Italy, 1996, pp. 148–156.

[8] Y. Freund, R.E. Schapire, Game theory on-line prediction and boosting, in: Proceedings of the Ninth Annual Conference on Computer Learning Theory, Desenzano del Garda, Italy, 1996, pp. 325–332.

[9] J. Friedman, Greedy Function Approximation, Department of Statistics, Stanford University Technical Report, 1999.

[10] J. Friedman, T. Hastie, R. Tibshirani, Additive logistic regression: a statistical view of boosting, Ann. Statist. 28(2) (2000) 337–374.

[11] R. Garcia-Marciel, I. Mora-Jimenez, A. Figueiras-Vidal, Improving kernel-based classifiers by guided dynamic sample selection, in: Proceedings of the 13th International Conference on Artificial Neural Networks in Engineering, St. Louise, MI, ASME Press, 2003, pp. 27–32.

[12] V. Gómez-Verdejo, J. Arenas-García, M. Ortega-Moral, A. Figueiras-Vidal, Designing RBF classifiers for weighted boosting, in: Proceedings of IJCNN'05, Montreal, Canada, 2005, pp. 1057–1062.

[13] V. Gómez-Verdejo, M. Ortega-Moral, J.P. Cabrera, J. Arenas-García, A. Figueiras-Vidal, Boosting by emphasizing boundary samples, in: Proceedings of the Learning'04 International Conference, Elche, Spain, 2004, pp. 67–72.

[14] L. Hansen, O. Salomon, Neural network ensembles, IEEE Trans. Pattern Anal. Mach. Intell. 12 (10) (1990) 993–1001.

[15] S. Haykin, Neural Networks: A Comprehensive Foundation, second ed., Prentice-Hall, Upper Saddle River, NJ, 1999.

[16] J.T. Kwok, Moderating the output of support vector classifiers, IEEE Trans. Neural Networks 10 (5) (1999) 1018–1031.

[17] H.B. Mann, D.R. Whitney, On a test of whether one of two random variables is stochastically larger than the other, Ann. Math. Statist. 18 (1947) 50–60.

[18] R. Meir, G. Ratsch, An introduction to boosting and leveraging, in: S. Mendelson, A. Smola (Eds.), Advanced Lectures on Machine Learning, Lecture Notes in Computer Science, Springer, Berlin, 2003, pp. 119–184.

[19] G. Ratsch, T. Onoda, K.R. Muller, Soft margins for Adaboost, Mach. Learn. 42 (3) (2001) 287–320.

[20] G. Ratsch, M.K. Warmuth, Marginal boosting, in: Proceedings of the 15th of the Annual Conference on Computational Learning Theory, Sydney, Australia, Springer, Berlin, 2002.

[21] D. Rudin, R.E. Schapire, I. Daubechies, Boosting based on a smooth margin, in: Proceedings of COLT'04, 2004, pp. 502–517.

[22] A. Ruiz, P.E. López de Teruel, Nonlinear kernels-based statistical pattern analysis, IEEE Trans. Neural Networks 12 (1) (2001) 16–32.

[23] R.E. Schapire, The strength of weak learnability, Mach. Learn. 5 (2) (1990) 197–227.

[24] R.E. Schapire, Y. Singer, Improved boosting algorithms using confidence-rated predictions, Mach. Learn. 37 (3) (1999) 297–336.

[25] A.J.C. Sharkey, Multi-net Systems, in: A.J.C. Sharkey (Ed.), Combining Artificial Neural Nets: Ensemble and Modular Multi-Nets Systems, Springer, London, UK, 1999, pp. 1–30.

[26] J. Sklansky, L. Michelotti, Locally trained piecewise linear classifiers, IEEE Trans. Pattern Anal. and Mach. Intell. 2 (1) (1980) 101–111.

[27] L. Valiant, Theory of the learnable, Commun. ACM 27 (1984) 1134–1142.

Vanessa Gómez-Verdejo was born in Madrid, Spain, in 1979. She received the Telecommunication Engineer degree in 2002 from Universidad Politécnica de Madrid, Madrid, Spain. Presently, she is pursuing the Ph.D. degree at the Department of Signal Theory and Communications, Universidad Carlos III de Madrid. Her present research interests are centred in the fields of Adaptive Signal Processing and Machine Learning, mainly Neural Networks ensembles, and their applications.

Manuel Ortega-Moral was born in Burgos, Spain, in 1980. He received the Telecommunication Engineer degree in 2003 from Centro Politécnico Superior de Zaragoza. He is currently pursuing the Ph.D. degree at the Department of Signal Theory and Communications, Universidad Carlos III de Madrid, focusing his research interests on the fields of Neural Networks, Adaptive Signal Processing and their applications.

Jerónimo Arenas-García was born in Seville, Spain, in 1977. He received the Telecommunication Engineer degree in 2000 from Universidad Politécnica de Madrid (ranked number 1; National Award to graduation), and his Ph.D. degree from Universidad Carlos III de Madrid, where he is now an Assistant Professor. His present research interests are focused in the fields of Adaptive Signal Processing, Machine Learning and their applications.

Anibal R. Figueiras-Vidal received the Telecommunication Engineer degree from Universidad Politécnica de Madrid, Madrid, Spain, in 1973 (ranked number 1; National Award to graduation) and the Doctor degree (Honors) from Universidad Politécnica de Barcelona, Barcelona, Spain, in 1976. He is a Professor of signal theory and communications with Universidad Carlos III, Madrid. His research interests are digital signal processing, digital communications, neural networks, and learning theory. He has (co)authored more than 300 journal and conference papers in these areas. Dr. Figueiras received an ''Honoris Causa'' Doctorate degree in 1999 from Universidad de Vigo, Vigo, Spain. He is currently General Secretary of the Royal Academy of Engineering of Spain.