ESANN'2005 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 27-29 April 2005, d-side publi., ISBN 2-930307-05-6.

Boosting by weighting boundary and erroneous samples*

Vanessa Gómez-Verdejo, Manuel Ortega-Moral,
Jerónimo Arenas-García and Aníbal R. Figueiras-Vidal
Department of Signal Theory and Communications

Universidad Carlos III de Madrid
Avda. Universidad 30, 28911 Leganés (Madrid) SPAIN.
{*vanessa,ortegam,jarenas,arfv*}*@tsc.uc3m.es*

Abstract. This paper shows that new and flexible criteria to resample populations in boosting algorithms can lead to performance improvements. Real Adaboost emphasis function can be divided into two different terms, the first only pays attention to the quadratic error of each pattern and the second takes only into account the "proximity" of each pattern to the boundary. Here, we incorporate an additional degree of freedom to this fixed emphasis function showing that a good tradeoff between these two components improves the performance of Real Adaboost algorithm. Results over several benchmark problems show that an error rate reduction, a faster convergence and overfitting robustness can be achieved.

1 Introduction

Multi-net systems are a good approach to solve difficult tasks which usually require a very complex net, overcoming sizing and training difficulties. Consequently, during the last years there has been an intensive research work to design Neural Networks (NN) ensembles, following different approaches, such as bagging or boosting [10].

Among the different methods that have been proposed, boosting procedures [8], and in particular Real Adaboost (RA) algorithm, have become very popular as a way to obtain advantage of "weak" learners. Concretely, RA works by adding sequentially a new base learner trained with an emphasized population, mainly paying its attention on the most erroneous samples (a detailed description can be found in [9]).

Breiman's work [2] points out that boosting schemes work because of focusing on the problematic patterns, independently of the explicit form of the emphasis function. Nevertheless, in [4] we showed that the RA emphasis function really combines in a fixed way two emphasis terms: one pays attention to the quadratic error of each pattern, and another takes into account its "proximity" to the boundary. Furthermore, we also tested how the performance of classical RA schemes can be improved focusing directly on the samples near the boundary.

*This work has been partly supported by grant CICYT TIC2002-03713. The work of V.Gómez-Verdejo was also supported by the Chamber of Madrid Community and European Social Fund by a grant.

In this paper we incorporate an additional degree of freedom to this fixed emphasis function, by means of a parameter that lets us combine the attention paid to the erroneous and boundary samples; in this way, we will show that usually the best way to resample the population is emphasizing neither most erroneous samples nor boundary ones, but a particular tradeoff between them.

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 weighted emphasis functions in some benchmark problems. Finally, in Section 5, conclusions and future research lines will be presented.

2 Real Adaboost

The fundamental idea of RA is to combine several "weak" learners in such a way that the ensemble improves its performance. To build up an RA classifier, at each round $t = 1, ..., T$ a new base learner is added implementing a function $o_t(\mathbf{x}_i) : X \longrightarrow [-1, 1]$ aimed to minimize the following error function

$$E_t^2 = \sum_{i=1}^{l} D_t(i)(t_i - o_t(\mathbf{x}_i))^2 \tag{1}$$

where l is the number of training patterns, $t_i \in \{-1, 1\}$ is the target for pattern $\mathbf{x}_i$, $o_t(\mathbf{x}_i)$ is the "weak" learner output for $\mathbf{x}_i$, and $D_t(i)$ is the weight that the t-th learner emphasis function assigns to $\mathbf{x}_i$. Initially, all weights have the same value $D_1(i) = 1/l$, $\forall i = 1, ..., l$, and they are then updated according to

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

where Z_t is a normalization factor assuring that $\sum_{i=1}^{l} D_t(i) = 1$, and α_t is the weight assigned to the t-th weak learner. Overall output of the net $f_T(\mathbf{x}_i)$ is calculated as the weighed combination of all learners:

$$f_T(\mathbf{x}_i) = \sum_{t=1}^{T} \alpha_t o_t(\mathbf{x}_i) \tag{3}$$

Values α_t are calculated in each round according to

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

where $r_t = \sum_{i=1}^{l} D_t(i) o_t(\mathbf{x}_i) t_i$. This choice of α_t values assures that the following training error bound is minimized

$$E_{\text{train}} = \sum_{i=1}^{l} | \operatorname{sign}(f(\mathbf{x}_i)) \neq t_i | \leq \sum_{i=1}^{l} \exp(-f_t(\mathbf{x}_i) t_i) \tag{5}$$

Additionaly, in [6] is showed that the same criterion maximizes the classification margin defined as $\rho = \min_{i=1\ldots l} f_t(\mathbf{x}_i)t_i$.

Analyzing in detail emphasis function (2), it can be showed that it does not only pay attention to the error of each pattern but also to its "proximity" to the boundary, as we explained in [4] by rewritting (2) in the following manner

$$D_{t+1}(\mathbf{x}_i) = \frac{1}{Z'_t} \exp\left(-\frac{1}{2}\right) \exp\left(\frac{(f_t(\mathbf{x}_i) - t_i)^2}{2}\right) \exp\left(-\frac{f_t^2(\mathbf{x}_i)}{2}\right) \tag{6}$$

Thus, it can be divided into two different factors:

$$\text{Error emphasis} \qquad \exp\left(\frac{(f_t(\mathbf{x}_i) - t_i)^2}{2}\right) \tag{7}$$

$$\text{Boundary emphasis} \qquad \exp\left(-\frac{f_t^2(\mathbf{x}_i)}{2}\right) \tag{8}$$

3 A weighted emphasis function

In the light of (6), one may wonder if this fixed combination of emphasis terms is optimal in all situations. So, in this paper we study the effect of using an emphasis function that combines the error term (7) and the boundary one (8) by means of a weighting parameter λ ($0 \leq \lambda \leq 1$),

$$D_{\lambda,t+1}(i) = \frac{1}{Z_t} \exp\left(\lambda \cdot (f_t(\mathbf{x}_i) - t_i)^2 - (1-\lambda) \cdot f_t^2(\mathbf{x}_i)\right) \tag{9}$$

This flexible formulation allows us to pay more or less attention to the boundary "proximitity" or to the quadratic error of each sample by selecting different values λ. We can remark three special values of the weighting parameter:

- $\lambda = 0$: only the "proximity" to the boundary is taken into account.
- $\lambda = 0.5$: we get the classical RA emphasis function.
- $\lambda = 1$: the emphasis function only pays attention to the quadratic error.

In [4] we studied the first two particular cases[1], showing that focusing directly on the samples near the boundary ($\lambda = 0$) we can speed up the convergence, and even avoid the well-known overfitting problem of RA. In other cases, focusing on the most erroneous patterns works better. However, we will show next that the best way to emphasize the population is frequently neither of previous ones, but intermediate values of weighting parameter, depending on the particular problem we are solving.

[1] We have tested a normalized version of (9) when $\lambda = 0$ and $\lambda = 0.5$.

4 Experiments

To show how an adequate emphasis selection can improve the performance of boosting methods we have built a series of ensembles according to (9) for different values of the weighting parameter (λ) and we have evaluated their performance over several binary problems. In particular, we have selected six binary problems from [1]: *Abalone* (a multiclass problem converted to binary according to [7]), *Contraceptive*, *Image*, *Spam*, *Tictactoe* and *Waveform*, and also a synthetic problem from [5]: *Kwok*. In Table 1 we have summarized their main features (number of dimensions (dim), number of samples of each class (C_1/C_{-1}) in the training and test set). Some of the problems had a predefined test set; when this was not the case, ten random partitions with 40% of the data set have been selected to test the performance of the classifier.

Problem	dim	# Train samples	# Test samples
Abalone	8	1238/1269	843/827
Contraceptive	9	506/377	338/252
Image	18	821/1027	169/293
Kwok	2	300/200	6120/4080
Spam	57	1673/1088	1115/725
Tictactoe	9	199/376	133/250
Waveform	21	2694/1306	659/341

Table 1: Main features of the benchmark problems.

To build the ensembles we have used as base learners Multi Layer Perceptrons (MLPs) with different representational power, different number of hidden units (M). Each of these MLPs has been trained to minimize cost function (1) by means of a back-propagation algorithm with learning steps $\mu = 0.1$ and $\mu = 0.01$ for the hidden and output layer, respectively.

We have built different ensembles for λ in the range $[0, 1]$, using a 0.1 step. In all the cases, the selection of the ensemble output weights, α_t, is done according to (4). In this way, we are still minimizing a bound on the training error, and simultaneously maximizing the classifier margin, as RA does.

In Table 2 we have displayed the test errors for RA and the best result that was achieved when varying the weighting parameter (λ_0), averaged over 50 independent runs. We have used a different number of rounds (T) to assure a complete convergence of the ensemble. Furthermore, to measure the statistical importance of these approaches, λ_{RA} and λ_0, we have used the Wilcoxon Rank Test (WRT) [3], where a value p lower than 0.1 indicates that the differences between them are significant[2]; on the contrary, p is close to 1 when there is no

[2]Values of p lower than 0.001 have been rounded down to 0.

statistical difference between the two rates.

Problem	M	T	E_{RA}	λ_0	E_{λ_0}	WRT (p)
Abalone	9	100	19.42	**0.1**	**19.16**	0
	2	250	19.56	**0.3**	**19.27**	0
Contraceptive	6	20	28.91	**0.1**	**28.58**	0.16
	3	50	29.16	**0**	**28.39**	0.0028
Image	3	100	**2.74**	0.5	2.74	1
	2	150	**2.86**	0.5	2.86	1
Kwok	4	100	11.82	**0.4**	**11.70**	0
	2	200	12.27	**0.3**	**12.24**	0.34
Spam	3	100	**5.83**	0.5	5.83	1
	2	150	**5.92**	0.5	5.92	1
Tictactoe	4	150	2.92	**0.4**	**2.02**	0.0012
	2	900	7.78	**0.3**	**6.60**	0
Waveform	13	100	8.85	**0**	**8.16**	0
	6	100	8.27	**0**	**7.75**	0

Table 2: Test errors for Real Adaboost (RA) and the best value of the weighting parameter λ_0.

It can be seen that $\lambda = 0.5$, corresponding to RA, is only the best setting for *Image*; *Spam* results are independent of λ value for a wide range, to be more specific the error rate remains unchanged for λ from 0.2 to 0.6; and better results have been obtained for a different value in five out of the seven benchmark problems. In addition to this, we have observed some other important effects that are summarized next:

- In *Abalone*, *Contraceptive* and *Waveform*, emphasis functions which focus mainly on boundary samples ($\lambda \leq 0.3$) not only reduce the test error but also provide a faster convergence: for instance, in *Contraceptive* the same final error rate of RA was obtained with only three rounds.
- For some problems, certain selections of λ resulted in a much faster initial convergence although the final error is higher than the results displayed in Table 2. This effect is clearly shown in *Tictactoe* when emphasis is centered on erroneous samples ($\lambda \geq 0.8$).
- RA performance is frequently degraded due to overfitting during training. When a more appropiate value of λ was used, this problem was reduced drastically.

5 Conclusions and future work

In this paper we have studied the performance of boosting methods when using different tradeoffs between error and boundary emphasis. In particular, we have showed that a good selection of the weighting parameter can reduce the error rate, accelerate the convergence of the ensemble, and even avoid the overfitting problem.

These evidences suggest the appropriateness of designing automatic methods to select an optimum value of the weighting parameter for each classification problem. Although cross-validation is a straightforward choice, it would be more interesting designing a method that let us adapt easier λ during the ensemble growing. This constitutes a promising research line where we are currently working.

References

[1] C. L. Blake and C. J. Merz. UCI repository of machine learning databases. http://www.ics.uci.edu/ mlearn/MLRepository.html, 1998. University of California, Irvine, Dept. of Information and Computer Sciences.

[2] L. Breiman. Prediction games and arcing algorithms. Technical Report 504, Statistics Department, University of California, December 1997.

[3] J. D. Gibbons. *Nonparametric Statistical Inference*. Basel : Marcel Dekker, New York, 4th edition, 2003.

[4] V. Gómez-Verdejo, M. Ortega-Moral, J. P. Cabrera, J. Arenas-García, and A. Figueiras-Vidal. Boosting by emphasizing boundary samples. In *Proc. of the Learning'04 Intl. Conf.*, pages 67–72, Elche, Spain, 2004.

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

[6] R. Meir and G. Ratsch. An introduction to boosting and leveraging. In S. Mendelson and A. Smola, editors, *Advanced Lectures on Machine Learning*, LNCS, pages 119–184. Springer Verlag, 2003.

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

[8] R. E. Schapire. The strength of weak learnability. In *30th Annual Symposium on Foundations of Computer Science*, pages 28–33, 1989.

[9] R. E. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, 37(3):297–336, 1999.

[10] A. J. C. Sharkey. *Combining Artificial Neural Nets. Ensemble and Modular Multi-Net Systems*. Springer-Verlag, London, UK, 1999.