Boosting decision stumps to do pairwise classification

Xie Jun, Yu Lu, Zhu Lei and Xue Hui

Pairwise classification is a task which predicts whether two samples belong to the same class or not. Boosting provides a way of combining many weak classifiers to produce a strong one and has been regarded as one of the most successful classification methodologies. The problem of pairwise classification is addressed by boosting decision stumps, the simplest weak classifier. Based on gentle AdaBoost, pairwise gentle AdaBoost of decision stumps is proposed to do pairwise classification. To make the classifier deal with a pair of inputs, sample-weighted linear discriminant analysis (LDA) is proposed, which is tailored to boosting the framework. For pairwise classification, the proposed algorithm shows better performance than traditional boosting of decision stumps on two UCI data sets.

Introduction: As one of the most important classification methodologies, boosting provides a way of combining the performance of many weak classifiers to produce a powerful committee [1]. Boosting works by sequentially applying a classification algorithm to reweighted versions of the training data. For many classification algorithms, this simple strategy results in dramatic improvements in performance. The simplest weak classifier is a form of decision tree with a single node, known as a 'decision stump'. Owing to their simplicity, decision stumps have been commonly used in a boosting framework [2].

Pairwise classification is a task which predicts whether two samples belong to the same class or not. A typical pairwise classification task arises in face verification, where we are required to decide whether two photos come from the same person. Researches on pairwise classification generally focus on all kinds of supervised distance learning or metric learning.

Since boosting has been attracting wide attention since its birth, and decision stumps are easy to implement and widely used, we hope to boost decision stumps to solve pairwise classification problems. However, traditional boosting of decision stumps does not provide a pairwise classifier, that is, input to the classifier is one single sample, instead of a pair of samples. In this Letter, we develop a pair boost classifier based on gentle AdaBoost of decision stumps, called pairwise gentle AdaBoost, to do pairwise classification.

From gentle AdaBoost to pairwise gentle AdaBoost: Since we choose to base our algorithm on gentle AdaBoost, which is a robust and stable version of boosting, we start with a brief review of gentle AdaBoost [1]. For a two-class classification problem, the final model of boosting has the additive form as

$$H(\boldsymbol{v}) = \mathrm{sign}\left[\sum_{m=1}^{M} h_m(\boldsymbol{v})\right] \tag{1}$$

where $\boldsymbol{v}$ is a sample (feature vector), M is the number of boosting rounds and h_m is the weak classifier learned in the mth round. In gentle AdaBoost, h_m is trained to minimise the weighted least-squared error:

$$J_m = \sum_{i=1}^{N} w_i^{(m)}(z_i - h_m(\boldsymbol{v}_i))^2 \tag{2}$$

where N is the number of training samples, $z_i \in \{+1, -1\}$ is the class label of $\boldsymbol{v}_i$ (sample indexed by i). $w_i^{(m)}$ is the weight of $\boldsymbol{v}_i$ in the mth round and weights of the samples are updated by

$$w_i^{(m+1)} = w_i^{(m)} \mathrm{e}^{-z_i h_m(\boldsymbol{v}_i)}, \quad i = 1, 2, \ldots, N \tag{3}$$

Owing to its simplicity, decision stumps become the commonly used weak classifiers in a boosting framework. Decision stumps have the form

$$h(\boldsymbol{v}) = aI(\boldsymbol{v}^{\mathrm{f}} > \theta) + bI(\boldsymbol{v}^{\mathrm{f}} \le \theta) \tag{4}$$

where $\boldsymbol{v}^{\mathrm{f}}$ denotes the fth component of sample feature vector $\boldsymbol{v}$, θ is a threshold and I is the indicator function. To learn $h(\boldsymbol{v})$ defined in (4), we search over all possible components f and thresholds θ; given f and θ, a and b can be estimated by minimising (2), then we have

$$a = \frac{\sum_i w_i z_i I(\boldsymbol{v}_i^{\mathrm{f}} > \theta)}{\sum_i w_i I(\boldsymbol{v}_i^{\mathrm{f}} > \theta)} \quad b = \frac{\sum_i w_i z_i I(\boldsymbol{v}_i^{\mathrm{f}} \le \theta)}{\sum_i w_i I(\boldsymbol{v}_i^{\mathrm{f}} \le \theta)} \tag{5}$$

Here we drop superscript m for simplicity.

The detailed algorithm of gentle AdaBoost of decision stumps can be found in [2].

If we use classifier $H(\boldsymbol{v}):R^{\mathrm{d}} \to \{+1, -1\}$ (d is the dimension of feature vector) to do pairwise classification for a pair of samples, we have to concatenate them or take their difference. However, concatenation cannot reveal the correspondence between the same components of the two samples, whereas the difference of the two may lose useful information. Therefore, for pairwise classification, the desired classifier is a pairwise classifier whose input is a pair of samples instead of one single sample.

In this Letter, we develop a pairwise classifier based on gentle AdaBoost of decision stumps, called pairwise gentle AdaBoost of decision stumps, which maps a pair of samples to $\{+1, -1\}$, that is

$$\begin{aligned} &H(\boldsymbol{u}, \boldsymbol{v}):[R^{\mathrm{d}} \times R^{\mathrm{d}}] \to \{+1, -1\} \\ &H(\boldsymbol{u}, \boldsymbol{v}) = \mathrm{sign}\left[\sum_{m=1}^{M} h_m(\boldsymbol{u}, \boldsymbol{v})\right] \end{aligned} \tag{6}$$

Recall the optimisation of decision stumps: we would search over all possible components, and for each given component, we would search over all possible thresholds to split on. To make decision stumps fit the pairwise classification task, we develop pairwise decision stumps, which can deal with a pair of inputs. For each given component, we obtain N two-dimensional (2D) vectors, corresponding to N pairs of samples. Some projection method is used to project these 2D vectors (points on a 2D plane) onto a line. Then, we would search over all possible thresholds, as we do in the optimisation of traditional decision stumps.

The chosen projection method must be a discriminant projection method which is fitted to a boosting framework. As is known, the essential idea in boosting is that samples misclassified by one of the weak classifiers are given greater weight in the training of the next classifier. To be consistent with this idea, optimal objective of projection method should rely on the samples' weights, and should be updated in each round. However, none of the existing projection methods meets this requirement, since they are not designed for boosting. In this Letter, we extend the linear discriminant analysis (LDA) to sample-weighted LDA, which is tailored to boosting framework.

From LDA to sample-weighted LDA: It should be noted that sample-weighted LDA proposed here is entirely different from those proposed in [3, 4]. In the latter, different class pairs are weighted according to their contributions to the overall criterion, whereas in the sample-weighted LDA here, samples are assigned to different weights according to their previous classification performance. This is consistent with the idea of boosting.

For a sample of weight w, its contribution to the mean vector and scatter matrix in LDA should be increased by w times. Therefore we calculate the mean vector and scatter matrix of sample-weighted LDA by the following:

$$\begin{aligned} &\boldsymbol{m}_i = \frac{\sum_{\boldsymbol{x}_j \in D_i} w_j \boldsymbol{x}_j}{\sum_{\boldsymbol{x}_j \in D_i} \boldsymbol{x}_j}, \qquad \boldsymbol{m} = \frac{\sum_{j=1}^{N} w_j \boldsymbol{x}_j}{\sum_{j=1}^{N} \boldsymbol{x}_j} \\ &\boldsymbol{S}_{\mathrm{B}} = \sum_{i=1}^{C} \sum_{\boldsymbol{x}_j \in D_i} w_j (\boldsymbol{m}_i - \boldsymbol{m})(\boldsymbol{m}_i - \boldsymbol{m})^t \\ &\boldsymbol{S}_{\mathrm{W}} = \sum_{i=1}^{C} \sum_{\boldsymbol{x}_j \in D_i} w_j (\boldsymbol{x}_j - \boldsymbol{m}_i)(\boldsymbol{x}_j - \boldsymbol{m}_i)^t \end{aligned} \tag{7}$$

where $\boldsymbol{m}_i$ is the weighted mean of samples belonging to the ith class, $\boldsymbol{m}$ is the total mean of samples, $\boldsymbol{S}_{\mathrm{B}}$ is the weighted between-class scatter matrix, $\boldsymbol{S}_{\mathrm{W}}$ is the weighted within-class scatter matrix, N is the number of samples, C is the number of classes, D_i is the subset of samples belonging to the ith class and w_j is the weight of sample $\boldsymbol{x}_j$.

In pairwise decision stumps, samples are of 2D, and then the optimal projection vector is calculated by

$$\boldsymbol{p}^* = \arg\max_{\boldsymbol{p}\in R^2} \frac{\boldsymbol{p}^t \boldsymbol{S}_B \boldsymbol{p}}{\boldsymbol{p}^t \boldsymbol{S}_W \boldsymbol{p}} \tag{8}$$

Pairwise gentle AdaBoost of decision stumps: The full algorithm of pairwise gentle AdaBoost of decision stumps is shown in Table 1.

Table 1: Pairwise gentle AdaBoost of decision stumps

input:$(\boldsymbol{u}_i, \boldsymbol{v}_i, z_i)$, $i = 1, 2, \ldots, N$, $\boldsymbol{u}_i, \boldsymbol{v}_i \in R^d$, $z_i \in \{+1, -1\}$,
($z_i = 1$ means that $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ belong to the same class.)
output: classifier $H(\boldsymbol{u}, \boldsymbol{v})$:$[R^d \times R^d] \to \{+1, -1\}$

(1) Initialise the weights by setting $w_i^{(1)} = 1$, $i = 1, \ldots, N$
(2) For $m = 1, 2, \ldots, M$
(2.1) For $r = 1, 2, \ldots, d$
(2.1.1) calculate $\boldsymbol{S}_B$, $\boldsymbol{S}_W$ by (7) with samples
$\boldsymbol{x}_i = [\boldsymbol{u}_i^r \boldsymbol{v}_i^r]^t$ and weights $w_i^{(m)}$, $i = 1, 2, \ldots, N$
($\boldsymbol{u}_i^r$ denotes the rth component of sample $\boldsymbol{u}_i$)
(2.1.2) find the best projection vector $\boldsymbol{p}_r \in R^2$ by (8)
(2.2) Train classifier
$h_m(\boldsymbol{u}, \boldsymbol{v}) = aI([\boldsymbol{u}^f \boldsymbol{v}^f] \cdot \boldsymbol{p}_f > \theta) + bI([\boldsymbol{u}^f \boldsymbol{v}^f] \cdot \boldsymbol{p}_f \le \theta)$
by minimising $J_m = \sum_{i=1}^{N} w_i^{(m)} (z_i - h_m(\boldsymbol{u}_i, \boldsymbol{v}_i))^2$
(The optimisation of f, θ, a and b can be found in the second Section.)
(2.3) Update weights of each training sample by
$w_i^{(m+1)} = w_i^{(m)} e^{-z_i h_m(\boldsymbol{u}_i, \boldsymbol{v}_i)}$
and renormalise.
(3) Output the final classifier

$$H(\boldsymbol{u}, \boldsymbol{v}) = \text{sign}\left[\sum_{m=1}^{M} h_m(\boldsymbol{u}, \boldsymbol{v}) + \sum_{m=1}^{M} h_m(\boldsymbol{v}, \boldsymbol{u})\right]$$

As we know, a natural requirement for a pairwise classification is that the order of the two input samples should not influence the classification result. To meet this requirement, we enforce the final classifier $H(\boldsymbol{u}, \boldsymbol{v})$ to be symmetric by step (3) in Table 1.

Experimental results: To evaluate the performance of the proposed algorithm, we compared it with traditional boosting of decision stumps with two kinds of input: concatenation and difference of the two samples. To show the effect of sample-weighted LDA in pairwise boosting of decision stumps, we also did experiments of pairwise boosting with traditional LDA.

Experiments have been done on four data sets from UCI machine learning repository [5]. The description of the four datasets can be found in Table 2 (second to fourth columns). We randomly select single samples from each class and construct pairs of samples of the same classes and different classes. The number of pairs of samples in the experiments is shown in Table 2 (the last two columns). To make the results more reasonable, we did each experiment for several tens of times with different training and test samples. In each experiment, each single sample is not used more than once and the random selection is exactly fair to each class and each pair of classes. In the experiments, the dimension of the samples is reduced by principal component analysis (PCA) before the samples are sent to the boosting classifiers, the dimension are shown in Table 2 (fifth column). Rounds in the boosting are set to $M = 400$. The results are the average of 50 random experiments.

Table 2: Dataset information and experiment setting

	Dataset information			Experiment information		
Datasets	Number of instances	Number of classes	Number of attributes	Sample dimension	Pairs of samples of the same class	Pairs of samples of different classes
MF	2000	10	649	10	460	540
LR	20 000	26	16	16	1300	1300
Covtype	581 012	7	54	15	840	840
Optdigits	5620	10	64	15	1260	1260

'MF': multiple features UCI dataset.
'LR': letter recognition UCI dataset.
'Covtype': cover type UCI dataset.
'Optdigits': optical recognition of handwritten digits UCI dataset.

As shown in Table 3, the proposed pairwise gentle AdaBoost of decision stumps show obviously better performance than that of original boosting of concatenation and difference of the two samples. Compared with the LDA, the proposed sample-weighted LDA shows great advantage, since it is tailored to boosting framework.

Table 3: Results of several methods to boosting decision stumps

Datasets	Boost of concatenation	Boost of difference	Pairwise boost with LDA	Pairwise boost with swLDA
MF	0.506	0.797	0.749	**0.855**
LR	0.502	0.685	0.682	**0.734**
Covtype	0.503	0.664	0.64	**0.712**
Optdigits	0.503	0.799	0.737	**0.838**

'Boost of concatenation': traditional boost of decision stumps, with concatenation of the two samples as input.
'Boost of difference': traditional boost of decision stumps, with difference of the two samples as input.
'Pairwise boost with LDA': proposed pairwise boost of decision stumps except that traditional LDA, instead of sample-weighted LDA, is used as projection method.
'Pairwise boost with swLDA': proposed pairwise boost of decision stumps we proposed, sample-weighted LDA is used as projection method.

Conclusion: In this Letter, we propose an algorithm called pairwise gentle AdaBoost of decision stumps to do pairwise classification. Since the traditional boosting of decision stumps cannot deal with a pair of inputs, we propose sample-weighted LDA to do discriminant analysis before optimisation of decision stumps. Sample-weighted LDA is tailored to boosting framework and has samples' weight dependant optimal objective. The results show that the proposed algorithm has obvious advantage over traditional gentle AdaBoost of decision stumps. Thus we provide a simple and effective method to do pairwise classification by boosting of decision stumps.

Acknowledgments: This work has been supported by the National Natural Science Foundation of China (61101202, 61375057) and the Natural Science Foundation of Jiangsu Province of China under grant BK20131298.

14 January 2014
doi: 10.1049/el.2014.0128

Xie Jun (*College of Command Information System, PLA University of Science and Technology, Nanjing 210007, People's Republic of China*)

Yu Lu and Zhu Lei (*Institute of Communications Engineering, PLA University of Science and Technology, Nanjing 210007, People's Republic of China*)

E-mail: yulu_mail@263.net

Xue Hui (*School of Computer Science and Engineering Southeast University, Nanjing 210096, People's Republic of China*)

References

1 Friedman, J., Hastie, T., and Tibshirani, R.: 'Additive logistic regression: a statistical view of boosting', *Ann. Stat.*, 2000, **28**, (2), pp. 337–407
2 Torralba, A., Murphy, K.P., and Freeman, W.T.: 'Sharing visual features for multiclass and multiview object detection', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2007, **29**, (5), pp. 854–869
3 Loog, M., Duin, R.P.W., and Haeb-Umbach, R.: 'Multiclass linear dimension reduction by weighted pairwise fisher criteria', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2001, **23**, (7), pp. 762–766
4 Fu, Z., and Robles-Kelly, A.: 'Learning object material categories via pairwise discriminant analysis'. IEEE Conf. Computer Vision and Patter Recognition (CVPR), IEEE Computer Society, Minneapolis, MN, USA, June 2007, pp. 1–7
5 'UCI machine learning repository', 2013. [Online]. Available at http://www.archive.ics.uci.edu/ml