Neurocomputing ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Analysis of classification margin for classification accuracy with applications ☆

Qutang Cai [a,*], Changshui Zhang [a], Chunyi Peng [b]

[a] State Key Laboratory on Intelligent Technology and Systems, Tsinghua National Laboratory for Information Science and Technology (TNList), Department of Automation, Tsinghua University, Beijing, China

[b] Microsoft Research Asia, 49 Zhichun Road, Haidian District, Beijing, China

ARTICLE INFO

Article history:
Received 15 April 2007
Received in revised form
25 November 2007
Accepted 3 March 2008
Communicated by J. Zhang

Keywords:
Classifier ensemble
Classification margin
Bagging
Optimal bound

ABSTRACT

Classification margin is commonly used for describing the classification capability of a committee of classifiers. In this paper, we study the relation between classification margin and misclassification error, focusing on exploring useful information about misclassification error from the known classification margin. We propose a max–min type bound concerning the minimal misclassification rate, and present some useful properties. Finally, we seek the way to improve classification performance by incorporating the classification margins, and devise an algorithm for improving average classification accuracy based on the proposed bound. Experimental results show the effectiveness of the proposed algorithm and also validate our analytic results.

© 2008 Elsevier B.V. All rights reserved.

1. Introduction

In the past, machine learning and statistical techniques performing classification tasks focused on designing a single classifier, like neural networks, decision trees, Bayesian classifiers and linear discrimination analysis algorithms (LDA) [4]. As has been observed, for single classifier systems, it is likely to overfit training data, and also difficult to make a good trade-off between complexity and generalizability. One possible way to avoid overfitting of single classifiers but making full use of the training data is by generating and combining multiple classifiers. Boosting [18] and bagging [1], which are also covered by the adaptive resampling and combining techniques (Arcing) [2], are two popular methods for the purpose. The recent proposed boosting methods, such as the well-known AdaBoost procedure [5,17], generate weak learners via dynamically reweighing training instances based on current classification results. The validity and efficiency of boosting methods has been extensively studied both in experiments and in theory [8,6]. The original bagging algorithms, including their related variants such as random forests [3] and random subspace methods [10], train base classifiers from data subsets or feature subspaces, and then output the classification results by voting or averaging. The effectiveness of bagging methods is known to reduce the variance of base classifiers [7].

One way of characterizing the strength of the combination of the resulting weak classifiers generated by boosting and bagging is by classification margin, which has been used in some previous research. Schapire et al. [17] observe that AdaBoost helps to maximize the number of examples with large margins. Breiman [3] used classification margin to study the correlation and validity of random forests. Other works concerning classification margin like [9,15] are mainly concerned with how to maximize the margin for special purposes or under additional assumptions. Meanwhile, for bagging-type algorithms where the training sets for training base classifiers are randomly constructed, the trained base classifiers are then inherently random. In other words, these trained classifiers can be treated to be drawn from the base classifier space according to some underlying probability distribution. Classification margin can then be viewed as the exceedance probability of correct classifiers, and, in general, the classification margin of Bagging can be empirically calculated using out-of-bag estimation. Currently, the connection between classification margin and classification error has not been fully investigated. We will focus on exploring the connections between the classification margin and misclassification error, assuming that the distribution of classification margin is known in advance. As we will show in later sections, the connections can be used to bound the error of optimal subensemble, to estimate average classification error, and to improve the classification accuracy.

☆ This work is supported by National 863 project (No. 2006AA10Z210).
* Corresponding author. Tel.: +86 10 62775783; fax: +86 10 62786911.
E-mail address: qutangcai@gmail.com (Q. Cai).

0925-2312/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2008.03.015

The remainder of this paper is organized as follows. In Section 2, we formulate the problems after introducing necessary definitions and notations. In Section 3, we investigate the relationship between classification margin and misclassification error. In Section 4, an optimization task is proposed for improving classification accuracy, and a detailed algorithm is developed. Conclusions are made in Section 5.

2. Problem formulation

Let X be the feature space and Y be the set of class labels. Let $\mathscr{D}$ denote the dataset, and each instance in $\mathscr{D}$ is represented by a feature-label pair $(x;y)$, where $x \in X, y \in Y$. In addition, we assume that samples are generated i.i.d. from an unknown underlying distribution $\mathscr{D}$ over $X \times Y$. Throughout this paper, we will use $P(\cdot)$ and $E(\cdot)$ as the probability function and expectation, respectively.

In a classification task, a classifier can be viewed as a parameterized mapping from the feature space X to Y. For example, the Fisher linear classifier for binary classification problems can be parameterized by its projection vector and a separating point. Therefore, we can write each individual classifier as a parameterized mapping $h(x;\theta)$, abbreviated by h_θ, where θ is the corresponding parameter, and x is the input feature. Thus, if the input is x, the classifier with parameter θ will predict the label with $h(x;\theta)$. Let the range of the classifier parameters be denoted by Θ. Then the base classifier space consists of all classifiers with parameters in Θ. We also use the same notation to represent the base classifier space when it does not introduce additional confusion.

As noted above, the classifier parameters are allowed to be random. For example, the classifiers built in the bagging algorithms vary with the random bootstrapped training sets. Therefore, we can assume that the classifiers are drawn for combining according to some unknown probability distribution over Θ, and write the distribution by ϑ. Now we introduce some definitions about the classification margin, which coincide with the definitions off Schapire et al. [17] and that of Breiman for random forests [3].

Definition 1 (*Margin function for ensemble classifiers*). For k base classifiers $h(\cdot;\theta_1), h(\cdot;\theta_2), \ldots, h(\cdot;\theta_k)$, the margin function for the ensemble is defined as

$$mg(x,y;\theta_1,\theta_2,\ldots,\theta_k) \stackrel{\text{def}}{=} \frac{1}{k}\left(\sum_{i=1}^{k} I(h(x;\theta_i)=y) - \max_{\substack{j\neq y \\ j\in Y}} \sum_{i=1}^{k} I(h(x;\theta_i)=j)\right), \quad (1)$$

where $I(\cdot)$ is the indicator function.

Definition 2 (*Margin function for parameter space*). The margin function for the classifiers in parameter space Θ is a function from $X \times Y$ to $[-1,1]$

$$mr(x,y) : X \times Y \mapsto [-1,1],$$

$$mr(x,y) \stackrel{\text{def}}{=} P_\vartheta(h(x,\theta)=y) - \max_{\substack{j\neq y \\ j\in Y}} P_\vartheta(h(x,\theta)=j). \quad (2)$$

The margin in (1) indicates the capability for correctly classifying (x,y) by majority voting using the given classifiers. The pair (x,y) is correctly classified if and only if $mg(x,y;\theta_1,\theta_2,\ldots,\theta_k) > 0$, so the misclassification error is

$$P_D(mg(x,y;\theta_1,\theta_2,\ldots,\theta_k) \leqslant 0). \quad (3)$$

Moreover, the margin in (1) also reflects the confidence in the classification. The larger the margin, the more the confidence.

In the definition of (2), the margin function $mr(x,y)$ maps $X \times Y$ into $[-1,1]$. This margin function can be viewed as the generalization of the definition in (2) to the base classifier space. The classification margin we mention below refers to (2) if there is no further emphasis. Moreover, since (x,y) is randomly generated, then $mr(x,y)$ is a random variable taking value in $[-1,1]$, and possesses a probability distribution P_m whose cumulative distribution function (cdf) is denoted by $F_m(\cdot)$. Thus, $F_m(\alpha) = P_D(\{(x,y) : mr(x,y) \leqslant \alpha\})$. In practical situations, F_m can be approximately calculated by empirical estimation. For example, in bagging algorithms, one can use an out-of-bag estimation to estimate F_m. Therefore, we will assume that $F_m(\cdot)$ is known in advance in this paper.

With $F_m(\cdot)$ being known, if $P_m(mr(\cdot)=1)=1$, one can infer that, with probability one, simply single classifier $h(x,\theta)$, $\theta \in \Theta$, is sufficient for the classification task and can achieve the zero error rate. However, there are many more general cases that are not so extremal, i.e., $mr(\cdot)$ need not be 1. What can we say about the misclassification error for the general cases? What does the classification margin imply? Can one use the classification margin for further improving the classification accuracy? The remainder of this paper will mainly deal with these problems. To sum up, we will obtain the following interesting results:

1. There is a bound for minimal classification error of the ensemble classifiers with the given classification margin.
2. The bound for minimal classification error is attainable. In other words, this bound is a tight bound.
3. One can improve the classification results by making use of classification margins.

3. Classification margin and misclassification error

The main purpose of this section is to study the relationship between classification margin and misclassification error. We first deduce a bound for minimal ensemble error rate from the margin distribution, and then show its different faces concerning classification performance. For simplicity, we only consider two-class classification problems, i.e., $Y=\{-1,+1\}$, and assume that the committee size in voting is always odd to avoid undecidable cases.

3.1. Bounds for minimal misclassification error

The base classifier space Θ can be divided into two types according to the number of the classifiers it contains (namely, the size of Θ): finite base classifier space and infinite base classifier space. There are also two ways of drawing classifiers from Θ: drawing with replacement and drawing without replacement. Drawing with replacement is more general in practice, since for many ensemble algorithms like bagging and AdaBoost, the base classifiers are allowed to be duplicated. Drawing without replacement does not allow the component classifiers to be duplicated. Though we are mainly interested in drawing with replacement, we also include the case of drawing without replacement for a complete discussion.

3.1.1. Drawing with replacement

Consider classifier space Θ with distribution ϑ. We obtain the following result for drawing with replacement, using a probabilistic method.

Proposition 1. *For any k, there exists a subset of size k, $\{\theta_{i_1^*},\ldots,\theta_{i_k^*}\} \subset \Theta$, such that the ensemble error* (3) *satisfies*

$$P_D(mg(x,y;\theta_{i_1^*},\ldots,\theta_{i_k^*}) \leqslant 0) \leqslant \int_{-1}^{1} B(\alpha,k)\,\mathrm{d}F_m(\alpha), \quad (4)$$

Please cite this article as: Q. Cai, et al., Analysis of classification margin for classification accuracy with applications, Neurocomputing (2008), doi:10.1016/j.neucom.2008.03.015

where the integral is Lebesgue–Stieltjes integral, $B(\alpha,k)\doteq\sum_{i=\lceil k/2\rceil}^{k}\binom{k}{i}((1-\alpha)/2)^i((1+\alpha)/2)^{k-i}$, *and* $\lceil k/2\rceil$ *represents the minimal integer not less than* $k/2$.

Proof. We draw k component classifiers' parameters $\theta_1,\ldots,\theta_k$ i.i.d. according to ϑ. For each (x,y) that $mg(x,y)=\alpha$, the number of classifiers in $\{h_{\theta_1},\ldots,h_{\theta_k}\}$ that correctly classified (x,y) is a binomial random variable with parameters k and $(1+\alpha)/2$. Thus, the probability that (x,y) is misclassified by the combination of $h_{\theta_1},\ldots,h_{\theta_k}$ is

$$B(\alpha,k)\doteq\sum_{i=\lceil k/2\rceil}^{k}\binom{k}{i}\left(\frac{1-\alpha}{2}\right)^i\left(\frac{1+\alpha}{2}\right)^{k-i}. \tag{5}$$

With the aid of Fubini's theorem (see [16]),

$$\begin{aligned}
&E_{\theta_1,\ldots,\theta_k\sim\vartheta}(P_{(x,y)\sim D}(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0))\\
&\quad=E_{\theta_1,\ldots,\theta_k\sim\vartheta}(E_{(x,y)\sim D}(I(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)|\theta_1,\ldots,\theta_k))\\
&\quad=E_{(x,y)\sim D;\theta_1,\ldots,\theta_k\sim\vartheta}(I(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0))\\
&\quad=E_{(x,y)\sim D}(E_{\theta_1,\ldots,\theta_k\sim\vartheta}(I(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)|mr(x,y)))\\
&\quad=E_{(x,y)\sim D}(B(mr(x,y),k))\\
&\quad=\int_{-1}^{1}B(\alpha,k)\,\mathrm{d}F_m(\alpha).
\end{aligned}\tag{6}$$

Therefore,

$$E_{\theta_1,\ldots,\theta_k\sim\vartheta}(P_{(x,y)\sim D}(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0))=\int_{-1}^{1}B(\alpha,k)\,\mathrm{d}F_m(\alpha). \tag{7}$$

Thus there must exist one choice $\{\theta_{i_1^*},\ldots,\theta_{i_k^*}\}\subseteq\Theta$, such that the ensemble error rate does not exceed $\int_{-1}^{1}B(\alpha,k)\,\mathrm{d}F_m(\alpha)$. □

3.1.2. Drawing without replacement

For drawing without replacement, if the space Θ contains no atom (an element that takes place with positive probability), the previous result can still be applied without further modification. Otherwise, if Θ contains some atoms, then the atoms are at most infinitely countable, since the total probability should not exceed 1. Since the drawing without replacement is out of our interest, we only consider one special case that might be useful. We consider that Θ contains only finite atoms $\theta_1,\ldots,\theta_n$; and these atoms occur equally with probability $1/n$. For this case, the method of drawing without replacement is to draw $\{\theta_{i_1},\ldots,\theta_{i_k}\}$, where k is the specified size and $1\leqslant i_1<i_2<\cdots<i_k\leqslant n$.

Proposition 2. *For any* k, *there exists a subset* $\{\theta_{i_1^*},\ldots,\theta_{i_k^*}\}\subset\Theta$, *where* $1\leqslant i_1^*<i_2^*<\cdots<i_k^*\leqslant n$, *such that the ensemble error rate* (3) *satisfies*

$$P_D(mg(x,y;\theta_{i_1^*},\ldots,\theta_{i_k^*})\leqslant 0)\leqslant\int_{-1}^{1}f(n,k,\alpha)\,\mathrm{d}F_m(\alpha), \tag{8}$$

where

$$f(n,k,\alpha)=\sum_{i=\lceil k/2\rceil}^{\min(k,n_e)}\binom{n_e}{i}\binom{n_c}{k-i}\Big/\binom{n}{k}, \tag{9}$$

and $n_c=n(1+\alpha)/2$, $n_e=n(1-\alpha)/2$.

Proof. Using the similar arguments as the proof of Proposition 1, we draw a k-size subset by uniformly random selection (without replacement). For each (x,y) that $mg(x,y)=\alpha$, the number of classifiers in Θ that correctly classify (x,y) is n_c, so the number of k-subsets in Θ that misclassifies (x,y) is $\sum_{i=\lceil k/2\rceil}^{\min(k,n_e)}\binom{n_e}{i}\binom{n_c}{k-i}$. An application of the Polya urn model [11] shows that each k-subset that misclassifies (x,y) is drawn with probability $1/\binom{n}{k}$. Hence the probability that (x,y) is misclassified is

$$f(n,k,\alpha)=\sum_{i=\lceil k/2\rceil}^{\min(k,n_e)}\binom{n_e}{i}\binom{n_c}{k-i}\Big/\binom{n}{k}. \tag{10}$$

As in (6), we have

$$E_{\theta_1,\ldots,\theta_k\sim\vartheta}(P_{(x,y)\sim D}(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0))=\int_{-1}^{1}f(n,k,\alpha)\,\mathrm{d}F_m(\alpha). \tag{11}$$

From (11), there exists at least one choice of k-subset in Θ, $\{\theta_{i_1^*},\ldots,\theta_{i_k^*}\}$, where $1\leqslant i_1^*<i_2^*<\cdots<i_k^*\leqslant N$, such that (8) holds. □

3.1.3. Application to one classifier pruning problem

One direct application of Propositions 1 and 2 is to answer one problem raised in the classifier pruning scenario: for a large committee of trained classifiers, if one wants to reduce the committee size to an acceptable amount, how small can the misclassification error be achieved?

The problem of identifying an optimal subset is a typical combinatorial problem, and is NP-complete [20]. However, since the margin function of the base classifier space can be calculated, $F_m(\cdot)$ can be obtained, and our result can then provide an upper bound for the error rate of the optimal subensemble. If the reduced committee classifiers are allowed to be duplicated, Proposition 1 gives an upper bound for the possible minimal error rates. If the reduced committee classifiers are not allowed to be duplicated, Proposition 2 provides an upper bound for the minimal misclassification error.

3.2. The tightness property

A close look at (4) reveals that the bound is distribution free with respect to ϑ, and depends only on $F_m(\cdot)$ and k. A further exploration below shows that it is actually tight in the sense that, for some fixed $F_m^*(\cdot)$, the bound equals the possible minimal ensemble error for some sample space which possesses the same classification margin. (Similarly, we can also show that (8) cannot be further improved.) Let $err^*(\Theta,k)$ denote the minimal ensemble error of the k classifiers chosen from the base classifier space Θ. We have the following proposition.

Proposition 3. *There are some cumulative distribution functions* $F_m^*(\cdot)$, *such that*

$$\begin{aligned}
&\int_{-1}^{1}B(\alpha,k)\,\mathrm{d}F_m^*(\alpha)\\
&\quad=\sup\{err^*(\Theta,k):\ \textit{the } F_m(\cdot)\textit{ of }\Theta\textit{ equals }F_m^*(\cdot)\}.
\end{aligned}\tag{12}$$

Proof. Firstly, by (4), for all F_m^*,

$$\begin{aligned}
&\int_{-1}^{1}B(\alpha,k)\,\mathrm{d}F_m^*(\alpha)\\
&\quad\geqslant\sup\{err^*(\Theta,k):\text{the }F_m(\cdot)\text{ of }\Theta\text{ equals }F_m^*(\cdot)\}.
\end{aligned}\tag{13}$$

We only need to show that F_m^* exists such that

$$\begin{aligned}
&\int_{-1}^{1}B(\alpha,k)\,\mathrm{d}F_m(\alpha)\\
&\quad\leqslant\sup\{err^*(\Theta,k):\text{the }F_m(\cdot)\text{ of }\Theta\text{ equals }F_m^*(\cdot)\}.
\end{aligned}\tag{14}$$

We will show that the family of cdfs that equal 0 at the origin all satisfy (14). We prove this by construction. Since $F_m^*(0)=0$, then for almost all instance (x,y), $mr(x,y)>0$ (a.k.a., $P_D(mr(x,y)>0)=1$). We consider the infinite base classifier space Θ, and make the feature space X and the sample distribution D satisfy the following two conditions:

Discriminability: $\forall\theta\in\Theta$, the conditional expectation

$$E_D(I(h(x,\theta)=y)|mr(x,y)=\alpha)=\frac{1+\alpha}{2}. \tag{15}$$

Please cite this article as: Q. Cai, et al., Analysis of classification margin for classification accuracy with applications, Neurocomputing (2008), doi:10.1016/j.neucom.2008.03.015

Independence: $\forall\theta_1,\theta_2,\ldots,\theta_n$ in Θ such that $\theta_1\neq\cdots\neq\theta_n$,

$$P_D(I(h(x,\theta_1)=y),\ldots,I(h(x,\theta_n)=y)|mr(x,y)=\alpha)=\prod_{i=1}^{n}P_D(I(h(x,\theta_i)=y)|mr(x,y)=\alpha). \quad (16)$$

These conditions state that all the base classifiers have the same classification ability as (15), and the outputs of the base classifiers are independent in the sense of (16). It can be verified that the two conditions are consistent, and the examples meeting the two conditions exist.

To complete the proof, we will show that under the previous construction, for fixed $\alpha>0$, $\forall\theta_1,\theta_2,\ldots,\theta_k$ (the θ's here need not be distinct),

$$P_D(mg(x,y;\theta_1,\theta_2,\ldots,\theta_k)<0|mr(x,y)=\alpha)\geqslant B(\alpha,k). \quad (17)$$

Thus
$err=E_D(P_D(mg(x,y;\theta_1,\ldots,\theta_k)<0|mr(x,y)=\alpha))\geqslant\int_{-1}^{1}B(\alpha,k)\,\mathrm{d}F_m^*(\alpha)$, and then $err^*\geqslant\int_{-1}^{1}B(\alpha,k)\,\mathrm{d}F_m^*(\alpha)$, so F_m^* meets (14).

(1) If $\theta_1,\ldots,\theta_k$ are all distinct from each other, then by (15) and (16),

$$P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0|mr(x,y)=\alpha))=B(\alpha,k). \quad (18)$$

(2) If some of $\theta_1,\ldots,\theta_k$ are duplicated, we denote the committee of these base classifiers by C, and rewrite them by distinct parameters $\theta'_1,\ldots,\theta'_m$, and let $t_1,\ldots,t_m$ denote their duplicated times, respectively. Thus $\sum_{i=1}^{m}t_i=k$. Now we add new different classifiers $\theta'_{m+1},\ldots,\theta'_k$ to $\{\theta'_1,\ldots,\theta'_m\}$, and obtain a new committee of k base classifiers, denoted by C'. Then as in (18), $P_D(mg(x,y;C')\leqslant 0|mr(x,y)=\alpha)=B(\alpha,k)$. Since

$$\begin{aligned}&P_D(mg(x,y;C)\leqslant 0|mr(x,y)=\alpha)\\&\quad-P_D(mg(x,y;C')\leqslant 0|mr(x,y)=\alpha)\\&\quad=P(mg(x,y;C)<0,mg(x,y;C')>0|mr(x,y)=\alpha)\\&\qquad-P(mg(x,y;C)>0,mg(x,y;C')<0|mr(x,y)=\alpha),\end{aligned} \quad (19)$$

we only need to show that

$$\begin{aligned}&P(mg(x,y;C)<0,mg(x,y;C')>0|mr(x,y)=\alpha)\\&\quad>P(mg(x,y;C)>0,mg(x,y;C')<0|mr(x,y)=\alpha).\end{aligned} \quad (20)$$

The event $\{(x,y):mg(x,y;C)<0,mg(x,y;C')>0,mr(x,y)=\alpha\}$ can be decomposed into the following events:

$$\begin{aligned}\omega_{(I_1,\ldots,I_k)}\overset{\mathrm{def}}{=}\{(x,y):&I(h_{\theta'_1}(x)=y)\\&=I_1,\ldots,I(h_{\theta'_k}(x)=y)=I_k,mr(x,y)=\alpha\},\end{aligned}$$

where I_i's take value in $\{0,1\}$, and

$$\sum_{i=1}^{m}I_i*t_i>k/2\quad\text{and}\quad\sum_{i=1}^{k}I_i<k/2. \quad (21)$$

Similarly, $\{(x,y):mg(x,y;C)>0,mg(x,y;C')<0,mr(x,y)=\alpha\}$ can be decomposed into the following events:

$$\begin{aligned}\omega'_{(I_1,\ldots,I_k)}\overset{\mathrm{def}}{=}\{(x,y):&I(h_{\theta'_1}(x)=y)\\&=I_1,\ldots,I(h_{\theta'_k}(x)=y)=I_k,mr(x,y)=\alpha\},\end{aligned}$$

where I_i's take value in $\{0,1\}$, and

$$\sum_{i=1}^{m}I_i*t_i<k/2\quad\text{and}\quad\sum_{i=1}^{k}I_i>k/2. \quad (22)$$

For all $(I_1,\ldots,I_k)$ satisfying (21), it can be verified that $(1-I_1,\ldots,1-I_k)$ also satisfies (22). Since

$$P_D(\omega_{(I_1,\ldots,I_k)}|mr(x,y)=\alpha)=\binom{k}{\sum_{i=1}^{k}I_i}\left(\frac{1+\alpha}{2}\right)^{\sum_{i=1}^{k}I_i}\left(\frac{1-\alpha}{2}\right)^{k-\sum_{i=1}^{k}I_i}$$

and

$$P_D(\omega'_{(1-I_1,\ldots,1-I_k)}|mr(x,y)=\alpha)=\binom{k}{\sum_{i=1}^{k}I_i}\left(\frac{1+\alpha}{2}\right)^{k-\sum_{i=1}^{k}I_i}\left(\frac{1-\alpha}{2}\right)^{\sum_{i=1}^{k}I_i},$$

then for $\alpha\geqslant 0$,

$$P_D(\omega_{(I_1,\ldots,I_k)}|mr(x,y)=\alpha)>P_D(\omega'_{(1-I_1,\ldots,1-I_k)}|mr(x,y)=\alpha). \quad (23)$$

Therefore,

$$\begin{aligned}&P(mg(x,y;C)<0,mg(x,y;C')>0|mr(x,y)=\alpha)\\&\quad=\sum_{(I_1,\ldots,I_k)\ \text{meets (21)}}P_D(\omega_{(I_1,\ldots,I_k)}|mr(x,y)=\alpha)\\&\quad>\sum_{(I_1,\ldots,I_k)\ \text{meets (21)}}P_D(\omega_{(1-I_1,\ldots,1-I_k)}|mr(x,y)=\alpha)\\&\quad=\sum_{(I_1,\ldots,I_k)\ \text{meets (22)}}P_D(\omega'_{(I_1,\ldots,I_k)}|mr(x,y)=\alpha)\\&\quad=P(mg(x,y;C)>0,mg(x,y;C')<0|mr(x,y)=\alpha)\end{aligned} \quad (24)$$

and (20) holds.

Consequently, by (18) and (20), (17) holds. □

3.3. Expectation and rate of convergence

In the proof of Proposition 1, we have shown in (7) that

$$\int_{-1}^{1}B(\alpha,K)\,\mathrm{d}F_m(\alpha)=E_{\theta_1,\ldots,\theta_k\sim\vartheta}(P_{(x,y)\sim D}(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)). \quad (25)$$

This equality leads to the following proposition.

Proposition 4. *Drawing k base classifiers independently according to distribution ϑ, the expectation of the ensemble error rate is $\int_{-1}^{1}B(\alpha,k)\,\mathrm{d}F_m(\alpha)$.*

Since $\int_{-1}^{1}B(\alpha,k)\,\mathrm{d}F_m(\alpha)$ can be viewed as the expectation of $B(\alpha,k)$, where α is a random variable with distribution $F_m(\alpha)$, we write $\int_{-1}^{1}B(\alpha,k)\,\mathrm{d}F_m(\alpha)$ by $E_D(B(\alpha,k))$ for later use. A direct application of Proposition 4 shows that the average classification error of Bagging algorithms is $E_D(B(\alpha,k))$.

Furthermore, $E_D(B(\alpha,k))$ can be used to dominate the following rate of convergence.

Proposition 5. *For $\alpha_0\in[-1,1]$, $\varepsilon>0$,*

(a) $$P_{\theta_1,\ldots,\theta_k\overset{\text{i.i.d}}{\sim}\vartheta}(P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)\geqslant F_m(\alpha_0)+\varepsilon)\leqslant\frac{1}{\varepsilon}\int_{\alpha_0^+}^{1}B(\alpha,k)\,\mathrm{d}F_m(\alpha),$$

(b) $P_{\theta_1,\ldots,\theta_k \overset{\text{i.i.d}}{\sim} \vartheta}(P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)$
$\leqslant F_m(\alpha_0)-\varepsilon)\leqslant \frac{1}{\varepsilon}\int_0^{\alpha_0^-} 1-B(\alpha,k)\,\mathrm{d}F_m(\alpha).$

Proof. For fixed $\theta_1,\ldots,\theta_k$, observe that the event $\{(x,y): P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)\geqslant F_m(\alpha_0)+\varepsilon\}$ can be decomposed into two parts

$$\begin{aligned}&\{(x,y): mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0\}\\&\quad=\{(x,y): mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0, mr(x,y)>\alpha_0\}\\&\qquad\cup\{(x,y): mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0, mr(x,y)\leqslant\alpha_0\}.\end{aligned}\tag{26}$$

Thus

$$\begin{aligned}&P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)\\&\quad=P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0; mr(x,y)>\alpha_0)\\&\qquad+P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0; mr(x,y)\leqslant\alpha_0).\end{aligned}\tag{27}$$

Besides,

$$\begin{aligned}&P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0; mr(x,y)\leqslant\alpha_0)\\&\quad\leqslant P_D(mr(x,y)\leqslant\alpha_0)=F_m(\alpha_0).\end{aligned}\tag{28}$$

Therefore,

$$\begin{aligned}&P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)\leqslant P_D(mg(x,y;\theta_1,\ldots,\theta_k)\\&\quad\leqslant 0; mr(x,y)>\alpha_0)+F_m(\alpha_0).\end{aligned}\tag{29}$$

With (29) and Chebychev's inequality,

$$\begin{aligned}&P_{\theta_1,\ldots,\theta_k \overset{\text{i.i.d}}{\sim}\vartheta}(P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)\geqslant F_m(\alpha_0)+\varepsilon)\\&\quad\leqslant P_{\theta_1,\ldots,\theta_k \overset{\text{i.i.d}}{\sim}\vartheta}(P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0; mr(x,y)>\alpha_0)\geqslant\varepsilon)\\&\quad\leqslant\frac{1}{\varepsilon}E_{\theta_1,\ldots,\theta_k \overset{\text{i.i.d}}{\sim}\vartheta}(P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0; mr(x,y)>\alpha_0))\\&\quad=\frac{1}{\varepsilon}E_D(P_{\theta_1,\ldots,\theta_k \overset{\text{i.i.d}}{\sim}\vartheta}(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0|mr(x,y)>\alpha_0))\\&\quad=\frac{1}{\varepsilon}\int_{\alpha_0^+}^1 B(\alpha,k)\,\mathrm{d}F_m(\alpha).\end{aligned}\tag{30}$$

Thus (a) holds. With a similar argument, (b) also holds. □

3.4. Numerical and asymptotic properties

We present here some numerical and asymptotic properties concerning the deduced bound, which provide an alternative way for calculation and characterize the limiting behavior. Before we go further, we need the following properties for $B(\alpha,k)$.

Lemma 6. (1) $B(\alpha,k)=betainc((1-\alpha)/2,\lceil k/2\rceil,\lfloor k/2\rfloor+1)$, *where betainc is the normalized incomplete beta function.*

$$betainc(t,a,b)\doteq\int_0^t u^{a-1}(1-u)^{b-1}\,\mathrm{d}u\Big/\int_0^1 u^{a-1}(1-u)^{b-1}\,\mathrm{d}u.$$

(2) *For* $\alpha>0$, $B(\alpha,k)\leqslant\min(\exp(-\alpha^2k/8),(2/\alpha\sqrt{k})\exp(-\alpha^2k/2))$.
For $\alpha<0$, $1-B(\alpha,k)\leqslant\min(\exp(-\alpha^2k/8),(2/\alpha\sqrt{k})\exp(-\alpha^2k/2))$.
(3) $B(\alpha,k)$ *is monotonically nonincreasing with* α, $\lim_{\alpha\to 1}B(\alpha,k)=0$, *and*

$$\lim_{k\to\infty}B(\alpha,k)=\begin{cases}0, & \alpha>0,\\ 1, & \alpha<0.\end{cases}$$

Remark 1. Lemma 6(1) is useful for numerical computation purpose, since $\binom{k}{i}$ in (5) will be large when k is large and a direct computation of (5) will encounter floating point overflow problems.

Remark 2. Lemma 6(3) shows the monotonicity and limit property of $B(\alpha,k)$. Lemma 6(2) gives an asymptotic bound for how fast $B(\alpha,k)$ tends to its limit with k increasing, from which we note that the larger $|\alpha|$, the faster the $B(\alpha,k)$ converges.

Proof. (1) A direct computation or using the properties of Beta function leads to

$$\int_0^1 u^{\lceil k/2\rceil}(1-u)^{\lfloor k/2\rfloor+1}\,\mathrm{d}u=\frac{\left(\left\lceil\frac{k}{2}\right\rceil-1\right)!\left\lfloor\frac{k}{2}\right\rfloor!}{k!}.\tag{31}$$

By integration by parts,

$$\begin{aligned}&\int_0^{(1-\alpha)/2}u^{\lceil k/2\rceil}(1-u)^{\lfloor k/2\rfloor+1}\,\mathrm{d}u\\&\quad=\sum_{i=\lceil k/2\rceil}^{k}\frac{\left(\left\lceil\frac{k}{2}\right\rceil-1\right)!\left\lfloor\frac{k}{2}\right\rfloor!}{i!(k-i)!}\left(\frac{1-\alpha}{2}\right)^i\left(\frac{1+\alpha}{2}\right)^{k-i}.\end{aligned}\tag{32}$$

By (31) and (32), the equation holds.

(2) We only consider the case that $\alpha>0$, the other case follows in the same way. Note that $\binom{k}{i}((1-\alpha)/2)^i((1+\alpha)/2)^{k-i}$ equals $Pr(X=i)$ in the binomial distribution $X\sim Bin(k,(1-\alpha)/2)$. The Hoeffding bound gives that

$$B(k,\alpha)\leqslant\exp\left(-\frac{\alpha^2k}{8}\right).\tag{33}$$

An improved bound of Lévy's bound [12] gives that

$$B(k,\alpha)\leqslant\frac{2}{\alpha\sqrt{k}}\exp\left(-\frac{\alpha^2k}{2}\right).\tag{34}$$

(3) The monotonicity and limiting behavior is a direct result of (1) and (2). □

Corollary 7. (1) *Law of large numbers*: $P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)$ *converges to* $F_m(0)$ *with probability* 1.

(2) *PAC-type convergence*: *For* $\delta>0$, *there exists a constant* k_0, *such that* $\forall k\geqslant k_0$, *with probability at least* $1-\delta$, $P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)\leqslant F_m(0)+\delta$. *Moreover*, k_0 *can be taken to be*

$$k_0=\frac{16}{\alpha_0^2}\log\frac{\sqrt{2}}{\delta}\quad\textit{where}$$
$$\alpha_0=\sup\left\{\alpha: F_m(\alpha)\leqslant F_m(0)+\frac{\delta}{2}\right\}.\tag{35}$$

Proof. (1) For arbitrary $\varepsilon>0$, applying Proposition 5 with $\alpha=0$,

$$\begin{aligned}&P_{\theta_1,\ldots,\theta_k \overset{\text{i.i.d}}{\sim}\vartheta}(P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)\\&\quad\geqslant F_m(0)+\varepsilon)\leqslant\frac{1}{\varepsilon}\int_{0^+}^1 B(\alpha,k)\,\mathrm{d}F_m(\alpha),\\&P_{\theta_1,\ldots,\theta_k \overset{\text{i.i.d}}{\sim}\vartheta}(P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)\\&\quad\leqslant F_m(0)-\varepsilon)\leqslant\frac{1}{\varepsilon}\int_{-1}^{0^-}1-B(\alpha,k)\,\mathrm{d}F_m(\alpha).\end{aligned}$$

By Lemma 6(2),

$$\begin{aligned}&P_{\theta_1,\ldots,\theta_k \overset{\text{i.i.d}}{\sim}\vartheta}(|P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)-F_m(0)|\geqslant\varepsilon)\\&\quad\leqslant\frac{1}{\varepsilon}\int_{[-1,1]\setminus\{0\}}\exp\left(-\frac{\alpha^2k}{8}\right)\mathrm{d}F_m(\alpha).\end{aligned}$$

By dominated convergence theorem,

$$\lim_{k\to\infty}\int_{[-1,1]/\{0\}}\exp\left(-\frac{\alpha^2k}{8}\right)\mathrm{d}F_m(\alpha)=0,$$

so $P_{\theta_1,\ldots,\theta_k \overset{\text{i.i.d}}{\sim}\vartheta}(|P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)-F_m(0)|\geqslant\varepsilon)\to 0$ and $P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)$ converges to $F_m(0)$ with probability 1.

Please cite this article as: Q. Cai, et al., Analysis of classification margin for classification accuracy with applications, Neurocomputing (2008), doi:10.1016/j.neucom.2008.03.015

(2) Let k_0, α_0 be the same as in (35). For $k \geqslant k_0$, by Proposition 5(a),

$$\begin{aligned}
&P_{\theta_1,\ldots,\theta_k \overset{\text{i.i.d}}{\sim} \vartheta}(P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)\geqslant F_m(0)+\delta)\\
&\leqslant P_{\theta_1,\ldots,\theta_k \overset{\text{i.i.d}}{\sim} \vartheta}\left(P_D(mg(x,y;\theta_1,\ldots,\theta_k)\leqslant 0)\geqslant F_m(\alpha_0)+\frac{\delta}{2}\right)\\
&\leqslant \frac{2}{\delta}\int_{\alpha_0^+}^{1} B(\alpha,k)\,\mathrm{d}F_m(\alpha)\\
&\leqslant \frac{2}{\delta}\int_{\alpha_0^+}^{1} \exp\frac{-\alpha^2 k}{8}\,\mathrm{d}F_m(\alpha)\\
&\leqslant \frac{2}{\delta}\int_{\alpha_0^+}^{1} \exp\frac{-\alpha_0^2 k_0}{8}\,\mathrm{d}F_m(\alpha)\\
&\leqslant \delta. \qquad \square
\end{aligned} \tag{36}$$

The above properties in Corollary 7 indicate that:

- For fixed k, the higher the probability that margin $mg(x,y)$ is large, the lower the expected classification error rate. This coincides with our intuition.
- As k increases, the law of large numbers implies that, with probability close to 1, the instances in $\{(x,y) : mg(x,y)>0\}$ will be correctly classified, and the instances in $\{(x,y) : mg(x,y)<0\}$ will be misclassified.
- If with a high probability, the margin $mg(x,y)$ is large, then the k needed to attain a small error rate is small. That is, using only a relatively small subset of classifiers can achieve an optimal classification capability.

4. Boosting the accuracy

In this section, we will seek the way to improve classification accuracy by incorporating classification margins. We will devise an algorithm for minimizing the proposed bound, and the experimental results indicate that the proposed algorithm can reduce the classification error. We also observe that:

- One can grow classifiers from any subspaces of the feature space by bagging. The idea of growing classifiers from feature subspaces has also been used in the random subspace method by Ho [10].
- The classifiers constructed from different feature subspaces will likely behave diversely, and often a notable portion of instances can easily be correctly classified in some feature subspaces but obscure in other subspaces.

By these two observations, it seems possible to improve classification accuracy by combining classifiers trained from different feature subspaces and taking advantages of the diverse classification capability.

4.1. Algorithm framework

4.1.1. Combining strategy

To improve the classification accuracy, it is a natural way to construct a new base classifier space based on the prescribed feature subspaces. We will employ a probabilistic technique which makes the margin function in the new base classifier space a linear combination of margin functions of the base classifier spaces grown from the feature subspaces.

Our method for constructing the new base classifier space is by assigning weights to the base classifier spaces. Let the base classifier spaces be denoted by $\Theta_1,\ldots,\Theta_n$, with their margin functions $mr_1(\cdot),\ldots,mr_n(\cdot)$, respectively. Then the new base classifier space is $\bigcup_{i=1}^{n}\Theta_i$. We want the margin function $mr(\cdot)$ in the new space to be a linear combination of $mr_1(\cdot),\ldots,mr_n(\cdot)$:

$$mr(\cdot) = w_1 * mr_1(\cdot) + w_2 * mr_2(\cdot) + \cdots + w_n * mr_n(\cdot), \tag{37}$$

where w_i is the weight assigned to Θ_i. In (37), w_i's can be further restricted to be nonnegative since one can reverse the output of all classifiers in Θ_i to make w_i nonnegative. In addition, w_i's are made to meet the normalized condition that $\sum_{i=1}^{n} w_i = 1$. We use the following two steps to achieve $mr(\cdot)$: for each base classifier θ:

Step 1: Randomly draw one index s from $\{1,\ldots,n\}$ with $P(s=i)=w_i$.

Step 2: Draw θ randomly from Θ_s according to ϑ_s.

By these two steps,

$$\begin{aligned}
mr(x,y) &= P(h(x;\theta)=y) - P(h(x;\theta)\neq y))\\
&= \sum_{i=1}^{n}[P(h(x;\theta)=y|\theta\in\Theta_s)\\
&\quad - P(h(x;\theta)\neq y|\theta\in\Theta_s)] * P(s=i)\\
&= \sum_{i=1}^{n} w_i * mr_i(x,y),
\end{aligned} \tag{38}$$

which is the desired margin function in (37).

4.1.2. The objective function

To convert the previous ideas into an optimization task, we need an objective function. By the analysis of classification margin, it is a natural way to use $E_D(B(\alpha,k))$ as the objective function. This brings three benefits:

- It is the average error rate of k base classifiers randomly drawn. Reducing $E_D(B(\alpha,k))$ will reduce the average error rate by Proposition 4.
- With $F_m(\cdot)$ fixed, $E_D(B(\alpha,k))$ is the pessimistic bound for minimal error rates. A further careful choice of the k base classifiers may be possible to achieve lower error rates.
- $E_D(B(\alpha,k))$ also dominates the PAC-type convergence rate. When $E_D(B(\alpha,k))$ is small, the error rate is expected to converge fast to its limit as the committee size increases.

To construct the new classifier space, the following optimization problem is to be solved:

$$\begin{aligned}
&\min \int_{-1}^{1} B(\alpha,k)\,\mathrm{d}F_m(\alpha),\\
&\text{where } F_m(\alpha) = P\left(\sum_{i=1}^{n} w_i * mr_i(x,y)\leqslant\alpha\right)\\
&\text{and } \sum_{i=1}^{n} w_i = 1,\ \ w_i\geqslant 0.
\end{aligned} \tag{39}$$

4.1.3. A suboptimal algorithm

Let the instances from the training set be denoted by $(x_1,y_1),\ldots,(x_m,y_m)$, then the discrete version of (39) is

$$\sum_{j=1}^{m} B\left(\sum_{i=1}^{n} w_i * mr(x_j,y_j),k\right). \tag{40}$$

The summands does not possesses "good" properties such as monotonicity or convexity for the free parameters w_i's , and (40) is difficult to be globally minimized. We use an approximate technique for minimizing (40).

As have been shown, as $\alpha\uparrow 1$, $B(\alpha,k)$ monotonically tends to 0. Thus we expect that maximizing the number of instances whose classification margin exceeds some specified level is helpful for reducing (40). We carry this out by solving the following problem,

Please cite this article as: Q. Cai, et al., Analysis of classification margin for classification accuracy with applications, Neurocomputing (2008), doi:10.1016/j.neucom.2008.03.015

for some $\gamma \geqslant 0$:

$$\begin{aligned} \min \quad & \sum_{j=1}^{m} \delta_j \\ \text{s.t.} \quad & \sum_{i=1}^{n} w_i * mr_i(x_j, y_j) \geqslant \gamma - \delta_j \\ & \text{for } j = 1, \ldots, m, \\ & \sum_{i=1}^{n} w_i = 1, \quad \delta_j \geqslant 0, \ \ w_i \geqslant 0 \\ & \text{for } i = 1, \ldots, n \text{ and } j = 1, \ldots, m. \end{aligned} \tag{41}$$

In (41), δ_j's can be viewed as penalty if the resulting $mr(x_j, y_j)$ is lower than the prescribed value γ. The optimization problem is tractable since it is a linear optimization problem and can be globally minimized efficiently via linear programming [19]. The solution for (41) only depends on γ, and we tune γ by grid searching for minimizing (40). The detailed procedure of our algorithm is summarized in Table 1.

4.2. Experimental results

To illustrate the effectiveness of the proposed algorithm, we compare the proposed algorithm with some other well-known related algorithms, including AdaBoost [5], bagging [1], random forest [3], and the random subspace method [10]. We choose these methods for comparison because: (A) AdaBoost is undoubtedly one of the most popular and well-known ensemble methods that can produce good results; (B) Bagging is one well known algorithm that is able to induce some classification margins, and our algorithm in the following experiments depends heavily on bagging; (C) Random forest is a successful variant of bagging; (D) Random subspaces methods also incorporate the same ideas of growing classifiers from randomly chosen feature subspaces as our methods; in fact, the random subspaces methods can also be viewed as a special type of bagging where the bootstrap resampling is cast on the features of the instances.

The datasets we used are chosen from the UCI Repository of machine learning databases [13], which have also been used extensively in related works. Since we only study the binary classification problem, we selected the two largest categories in each dataset for the classification task. The details of the datasets are presented in Table 2, including the numbers of attributes and continuous attributes, missing value information, and instance number. The attributes of these datasets consist of continuous (numerical) and categorical attributes, and seven datasets contain missing values.

Since decision trees can handle well both categorical and numerical attributes as well as missing values, we use decision trees as the base classifiers. We implement all the algorithms using Weka [21], and use the C4.5 decision tree [14] as the base classifier. Observe that in (37), if the margin functions are identical or close to each other, the combined margin function can hardly be changed. Therefore, for each dataset, we randomly draw a maximum of 25 subspaces from the original feature space with dimension of about $\frac{2}{3}$ of the dimension of entire space. By doing this, we expect to obtain classifier spaces in some of which the classifiers are not too weak and the margin functions are not too close to each other. For the margin functions in each base classifier spaces, we use an empirical out-of-bag estimation for their approximation.

Since we use a random sampling technique in choosing feature subspace to grow base classifiers, we should use a relative large k (compared with the number of feature subspaces) to achieve stable classification results. In our experiments, we set the committee size to be 100. We use a 10-fold cross-validation for calculating the average classification error, and the experiments on each dataset are run 100 times independently. The experimental results are given in Table 3. We note that for most datasets, the average error rates of the proposed algorithm are lower than the others. Our algorithm achieves the lowest misclassification error in 13 out of 19 datasets. A 95% confidence t-test also shows that the proposed algorithm statistically outperforms the others: our algorithm statistically beats AdaBoost in 15 datasets, bagging in 13 datasets, random forest in 12 datasets and the random subspace method in 17 datasets. In addition, the average error rates in the experiments are all close to the estimated expectation error given in Proposition 4. These validate that using $E_D(B(\alpha, k))$ as the objective function is effective, and our approximate optimization algorithm can successfully utilize the classification ability in different base classifier spaces to achieve the lower misclassification error.

Though our algorithm achieves better performance in the experiments, compared with bagging, it requires additional computational power in our experiments. Although the margin functions are presumably known, they require pre-estimation like out-of-bag estimation; however, since the margin functions are assumed to be known in advance, it will not be covered in the training steps. In the training procedure in Table 1, compared with bagging, the main extra computational costs are introduced by the grid search linear programming optimization. By the theory of linear programming, the computational cost of linear programming for (41) for one γ depends only on the size of the dataset m

Table 1
Algorithmic procedure for training

Input: Training set $\{(x_1, y_1), \ldots, (x_m, y_m)\}$; feature subspaces $S_1, \ldots, S_n$ as well as their margin function $mr_1(\cdot), \cdots, mr_n(\cdot)$; committee size k.
Training:
Optimization (grid search): Solving (41) based on $mr_i(\cdot)$ for different γ's, and obtain several candidate coefficients. Pick one solution $w_1^*, \ldots, w_n^*$ that minimizes (39). (E.g., let γ take $0.05 * i$, $i = 1, \ldots, 20$. Optimize (41) and obtain 20 groups of coefficients. Pick the group that maximizes (39) as the final choice for the coefficients of the feature subspaces.)
Constructing committee: Grow k classifiers independently from S_i's with probability w_i^*'s. For $l = 1, \ldots, k$,
Choosing feature subspace: randomly draw a feature subspace S_{i_l} from S_i's with probability w_i^*'s;
Training base classifier: train the base classifier θ_l from feature subspace S_{i_l}.
Output: The committee of classifiers $\theta_1, \ldots, \theta_k$.

Table 2
Description of datasets

Dataset	Attributes	Cont. attr.	Missing	Instances
Balance	4	0	No	576
Breast Wisc	9	9	Yes	699
Bupa	6	6	No	345
Credit-g	20	8	No	1000
Crx	15	6	Yes	690
Echocardio	10	8	Yes	131
Glass	9	9	No	146
Hayes Roth	4	0	No	129
Heart Cleve	13	5	Yes	303
Hepatitis	19	6	Yes	155
Horse Colic	26	8	Yes	366
House Votes	16	0	Yes	435
Ionosphere	34	32	No	351
Pima	8	8	No	768
Promoters	57	0	No	106
Sonar	60	60	No	208
Tic-tac-toe	9	0	No	958
Vehicle	18	18	No	435
Yeast	8	8	No	792

Please cite this article as: Q. Cai, et al., Analysis of classification margin for classification accuracy with applications, Neurocomputing (2008), doi:10.1016/j.neucom.2008.03.015

Table 3
Experimental results

Dataset	A	B	F	R	O	(E)	A/B/F/R/E
Balance	18.70 ± 4.32	15.54 ± 5.05	14.34 ± 4.30	7.91 ± 4.27	$\mathbf{5.89 \pm 3.44}$	(5.75)	+/+/+/+/0
Breast Wisc	3.31 ± 1.91	4.41 ± 2.59	3.53 ± 1.85	3.76 ± 2.50	$\mathbf{2.79 \pm 1.95}$	(2.73)	+/+/+/+/0
Bupa	30.43 ± 7.70	26.66 ± 7.17	28.09 ± 7.30	28.06 ± 7.73	$\mathbf{26.57 \pm 7.33}$	(25.80)	+/0/+/+/0
Credit-g	25.23 ± 3.55	25.70 ± 4.11	24.61 ± 3.75	24.06 ± 4.10	$\mathbf{22.87 \pm 4.26}$	(23.24)	+/+/+/+/0
Crx	13.72 ± 4.07	13.75 ± 4.05	14.28 ± 3.84	13.49 ± 4.15	$\mathbf{11.62 \pm 3.68}$	(12.14)	+/+/+/+/0
Echocardio	11.10 ± 7.66	9.42 ± 7.17	9.59 ± 6.98	10.17 ± 7.73	$\mathbf{9.33 \pm 7.33}$	(8.50)	+/0/0/+/0
Glass	$\mathbf{11.70 \pm 8.69}$	17.27 ± 8.53	12.83 ± 8.05	13.47 ± 7.81	12.27 ± 7.80	(11.37)	0/+/0/+/0
Hayes Roth	23.11 ± 10.45	21.0 ± 10.57	22.09 ± 10.92	23.46 ± 9.74	$\mathbf{18.39 \pm 9.24}$	(18.1)	+/+/+/+/0
Heart Cleve	19.06 ± 6.82	21.21 ± 6.30	18.81 ± 6.33	17.76 ± 6.36	$\mathbf{15.58 \pm 5.84}$	(16.39)	+/+/+/+/0
Hepatitis	16.19 ± 8.44	17.12 ± 9.84	16.30 ± 8.18	16.38 ± 9.80	$\mathbf{13.06 \pm 8.80}$	(13.25)	+/+/+/+/0
Horse Colic	17.12 ± 5.59	$\mathbf{14.49 \pm 6.70}$	15.52 ± 5.53	20.97 ± 6.60	14.60 ± 6.32	(14.34)	+/0/+/+/0
House Votes	4.90 ± 3.55	3.24 ± 2.76	3.52 ± 2.40	5.60 ± 3.09	$\mathbf{2.96 \pm 2.49}$	(3.21)	+/0/+/+/0
Ionosphere	6.01 ± 4.21	7.29 ± 4.42	6.60 ± 4.10	5.74 ± 4.15	$\mathbf{5.49 \pm 3.87}$	(5.65)	+/+/+/0/0
Pima	26.21 ± 4.94	24.26 ± 4.37	24.12 ± 4.82	25.27 ± 4.88	$\mathbf{23.73 \pm 4.59}$	(23.89)	+/0/0/+/0
Promoters	8.53 ± 10.09	12.55 ± 10.64	9.39 ± 9.26	8.55 ± 8.75	$\mathbf{6.73 \pm 7.93}$	(6.55)	+/+/+/+/0
Sonar	$\mathbf{13.52 \pm 7.46}$	23.43 ± 10.37	16.29 ± 8.43	20.43 ± 9.60	18.76 ± 9.40	(17.20)	−/+/−/+/0
Tic-tac-toe	$\mathbf{0.89 \pm 1.03}$	3.93 ± 2.22	2.79 ± 1.78	11.17 ± 3.30	3.44 ± 2.14	(3.48)	−/+/−/+/0
Vehicle	$\mathbf{1.72 \pm 1.85}$	4.68 ± 3.34	2.28 ± 2.22	2.80 ± 2.50	1.75 ± 2.06	(1.38)	0/+/0/+/0
Yeast	37.26 ± 5.05	$\mathbf{32.31 \pm 4.77}$	32.53 ± 5.55	33.01 ± 4.67	32.42 ± 4.22	(32.02)	+/0/0/0/0
#Best	4	2	0	0	13		
#Second	3	4	3	5	4		
#Moderate	2	1	12	2	2		
#Bad	5	3	3	8	0		
#Worst	5	9	1	4	0		

Experimental results, comparing the error rate of AdaBoost (A), bagging (B), random forests (F), random subspace methods (R), our algorithm (O) together with its estimated expectation error (E). The standard deviations for the five algorithms are also presented. For each dataset, we put in emphasis the **best** algorithm(s). The last five rows count the number of times each algorithm counts, respectively, among the best, second, moderate, bad and worst. A 95% confidence t-test of the misclassification rate between proposed algorithm and other algorithms is given in the last five columns, where a plus sign designates a statistically significant win, a minus designates a statistically significant loss, and zero means no statistical significance.

and the number of feature subspaces n (more precisely, it depends on $m+n$). When $m+n$ is large (for example, $m+n=10\,000$), the linear programming procedure will be slow and even unaffordable. Fortunately, in the experiments, $m+n$ are all smaller than 1500, and linear programming problems below this scale can be readily solved. For example, the grid search optimization for Credit-g dataset with $\gamma = 0.05 * i,\ i = 1, \ldots, 20$, only requires 53.4 s using *linprog* function in Matlab 6.5 on a pentium 2.4 GHz class machine. In the classification stage, the computational power required is the same as other voting-based algorithms.

Besides the improvement of classification accuracy, one advantage of our proposed algorithm is that the classification accuracy is predictable. Another advantage of our algorithm is its flexibility. A closer look at our algorithms reveals that our algorithm only requires that there are attainable margin functions on the base classifier spaces, and does not put restrictions on the construction method for the base classifier spaces. This means that the algorithm not only can be applied wherever the bagging algorithms can be used, but also can be applied to other classifier spaces provided that they possess probability distributions as well as estimable margin functions.

5. Conclusions

We have studied the relationship between classification margin and misclassification error. We obtain an upper bound for the optimal ensemble error based on the classification margin. We also show that the proposed bound is actually a tight bound, and can serve as the average ensemble error rate. We also present other properties of this bound, such as the alternative calculation method and the limiting behavior.

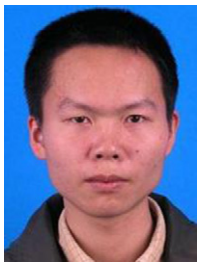
As a further step, we consider the possibility of improving classification accuracy based on margin functions, and develop a corresponding algorithm by minimizing the proposed bound. The experimental results show that reducing the bound helps to reduce the misclassification error, and the proposed algorithm outperforms some other related algorithms, including AdaBoost, bagging, random forests and random subspace methods. This also validates that it is possible to further improve the classification accuracy by taking the classification margins into account. Moreover, since our algorithm only requires that there are some margin functions on the base classifier spaces, we believe it is promising to be applicable to a wide range of classifier spaces. In a future exploration, we will extend our algorithm to large datasets, and consider combination strategies other than linear combination for incorporating classification margins.

References

[1] L. Breiman, Bagging predictors, Mach. Learn. 24 (2) (1996) 123–140.
[2] L. Breiman, Arcing classifiers, Ann. Stat. 26 (3) (1998) 801–849.
[3] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[4] R. Duda, P. Hart, D. Stork, Pattern Classification, Wiley, New York, 2000.
[5] Y. Freund, R. Schapire, A decision-theoretic generalization of online learning and an application to boosting, J. Comput. Syst. Sci. 55 (1) (1997) 119–139.
[6] J. Friedman, T. Hastie, R. Tibshirani, Additive logistic regression: a statistical view of boosting, Ann. Stat. 38 (2).
[7] Y. Grandvalet, Bagging equalizes influence, Mach. Learn. 55 (3) (2004) 251–270.
[8] T. Hastie, R. Tibshirani, J.H. Friedman, The Elements of Statistical Learning, Springer, Berlin, 2001.
[9] T. Hertz, A. Bar-Hillel, D. Weinshall, Boosting margin based distance functions for clustering, in: Proceedings of the 21st International Conference on Machine Learning, 2004.
[10] T.K. Ho, The random subspace method for constructing decision forests, IEEE Trans. Pattern Anal. Mach. Intell. 20 (8) (1998) 832–844.
[11] N.L. Johnson, S. Kotz, Urn Models and Their Applications: An Approach to Modern Discrete Probability Theory, Wiley, New York, 1977.
[12] O. Kraft, A note on exponential bounds for binomial probabilities, Ann. Inst. Statist. Math. 21 (1969) 219–220.
[13] C.B.D.J. Newman, S. Hettich, C. Merz, UCI Repository of Machine Learning Databases, 1998.

Please cite this article as: Q. Cai, et al., Analysis of classification margin for classification accuracy with applications, Neurocomputing (2008), doi:10.1016/j.neucom.2008.03.015

[14] R.J. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann, Los Altos, CA, 1993.
[15] G. Ratsch, M.K. Warmuth, Efficient margin maximizing with boosting, J. Mach. Learn. Res. 6 (2005) 2131–2152.
[16] W. Rudin, Real and Complex Analysis, third ed., McGraw-Hill, New York, 1987.
[17] R.E. Schapire, Y. Freund, P. Bartlett, W.S. Lee, Boosting the margin: a new explanation for the effectiveness of voting methods, Ann. Stat. 26 (5) (1998) 1651–1686.
[18] R. Schapire, A brief introduction to boosting, in: Proceedings of the 16th International Joint Conference on Artificial Intelligence, Morgan Kaufmann, Los Altos, CA, 1999, pp. 1401–1406.
[19] A. Schrijver, Theory of Linear and Integer Programming, Wiley, New York, 1986.
[20] C. Tamon, J. Xiang, On the boosting pruning problem, in: Proceedings of the 11th European Conference on Machine Learning, 2000.
[21] I.H. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques, second ed., Morgan Kaufmann, Los Altos, CA, 2005.



Qutang Cai was born in Fujian, China, in 1981. He received the B.E. degree with honors in Automation from Tsinghua University, Beijing, China, in 2002. He is now a Ph.D. student at the State Key Laboratory of Intelligent Technology and Systems, Department of Automation, Tsinghua University, Beijing. His research interests include machine learning, pattern recognition, signal processing, time series analysis and optimization theory.

Changshui Zhang received his B.S. degree in Mathematics from the Peking University, Beijing, China, in 1986, and Ph.D. degree from the Department of Automation, Tsinghua University, Beijing, China, in 1992. He is currently a Professor of the Department of Automation, Tsinghua University. He is an Associate Editor of the journal Pattern Recognition. His interests include artificial intelligence, image processing, pattern recognition, machine learning, evolutionary computation, etc.

Chunyi Peng received her B.E. degree in Automation and M.S. degree in Pattern Recognition and Intelligent System (both with honors) from Tsinghua University, Beijing, China, in 2002 and 2005. Her research interests include distributed computing, intelligent systems and signal processing.

Please cite this article as: Q. Cai, et al., Analysis of classification margin for classification accuracy with applications, Neurocomputing (2008), doi:10.1016/j.neucom.2008.03.015