ELSEVIER

Information Fusion 3 (2002) 245–258

INFORMATION FUSION

www.elsevier.com/locate/inffus

An experimental study on diversity for bagging and boosting with linear classifiers

L.I. Kuncheva [a,*], M. Skurichina [b], R.P.W. Duin [b]

[a] School of Informatics, University of Wales Bangor, Dean Street, Bangor, Gwynedd LL57 1UT, UK

[b] Pattern Recognition Group, Department of Applied Physics, Delft University of Technology, P.O. Box 5046, 2600 GA Delft, The Netherlands

Received 16 November 2001; received in revised form 18 April 2002; accepted 12 August 2002

Abstract

In classifier combination, it is believed that diverse ensembles have a better potential for improvement on the accuracy than non-diverse ensembles. We put this hypothesis to a test for two methods for building the ensembles: Bagging and Boosting, with two linear classifier models: the nearest mean classifier and the pseudo-Fisher linear discriminant classifier. To estimate diversity, we apply nine measures proposed in the recent literature on combining classifiers. Eight combination methods were used: minimum, maximum, product, average, simple majority, weighted majority, Naive Bayes and decision templates. We carried out experiments on seven data sets for different sample sizes, different number of classifiers in the ensembles, and the two linear classifiers. Altogether, we created 1364 ensembles by the Bagging method and the same number by the Boosting method. On each of these, we calculated the nine measures of diversity and the accuracy of the eight different combination methods, averaged over 50 runs. The results confirmed in a quantitative way the intuitive explanation behind the success of Boosting for linear classifiers for increasing training sizes, and the poor performance of Bagging in this case. Diversity measures indicated that Boosting succeeds in inducing diversity even for stable classifiers whereas Bagging does not.

Keywords: Combining classifier; Diversity; Bagging; Boosting

1. Introduction

A classifier is any function D by which we assign a class label ω from a set of predefined labels $\Omega = \{\omega_1, \ldots, \omega_c\}$ to an object represented as a data point $\mathbf{x}$ in a real n-dimensional space $\Re^n$. In the general case, the classifier output is a c-dimensional vector $[d_1(\mathbf{x}), \ldots, d_c(\mathbf{x})]^{\mathrm{T}}$ where $d_i(\mathbf{x})$ is the degree of "support" given by classifier D to the hypothesis that $\mathbf{x}$ comes from class ω_i, $i = 1, \ldots, c$. Without loss of generality we can restrict $d_i(\mathbf{x})$ within the interval $[0, 1]$, and call the classifier outputs "soft labels". Most often $d_i(\mathbf{x})$ is an estimate of the posterior probability $P(\omega_i|\mathbf{x})$. In some cases, "crisp" class labels are required, i.e., $d_i(\mathbf{x}) \in \{0, 1\}$, and $\sum_{i=1}^{c} d_i(\mathbf{x}) = 1$. These can be obtained by "hardening" the soft labels by assigning the largest value to 1 (the winning class label), and the remaining values to 0. Ties are resolved arbitrarily.

Classifier combination aims at a higher accuracy than that of a single D. The literature on classifier combination highlights the necessity of measuring and using the degree of diversity, independence, orthogonality, complementarity, etc., which are intuitively desirable characteristics of a classifier team [5,13,16,23,29,34]. Theoretically, a group of *independent* classifiers will improve upon the single classifier when majority vote combination is used. A dependent set of classifiers may be either better or worse [22]. Sometimes the difference is beneficial to the ensemble and yet sometimes it might be harmful. There is no consensus on what a "good" measure of diversity should be. The conceptual difficulty in defining diversity can be illustrated by an example. Assume that we have tested the L classifiers forming an ensemble on a data set of $N = 100$ ($N > L \geqslant 3$) objects (data points); each classifier recognizes all but one data points; and each classifier fails on a different point. Thus the estimated individual accuracy of each classifier is 0.99. Obviously, the classifier outputs are highly related, as a large amount of coincident decisions occur: the decisions of every pair of classifiers coincide in 98 out of 100 cases. Intuitively, this means that the diversity is

* Corresponding author. Tel.: +44-1248-383661; fax: +44-1248-361429.

E-mail address: l.i.kuncheva@bangor.ac.uk (L.I. Kuncheva).

1566-2535/02/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.
PII: S1566-2535(02)00093-3

low, and there is no gain in combining the classifiers. From another point of view, however, if we combine the classifier outputs, e.g., by taking the majority vote, we will arrive at a correct decision in all 100 cases. Thus the small outstanding improvement of 1% on the individual accuracy can be achieved through combining these "not too diverse" classifiers. So the potential for improvement is small, but this is all that is needed in this case. If we want diversity to measure the potential for improvement, what should its value be for this example, high or low? To account for this variety of viewpoints, in our experiments we use nine measures of diversity.

Bagging, Boosting, [1] Arcing [2] and the Random subspace method are guidelines for constructing classifier ensembles by varying the inputs. In this study we chose Bagging and Boosting which have shown good performance on various data sets [1,6].

Once the ensemble is put together, different combination methods can be used to derive the final class label of an object from the individual classifier outputs. In this study we used eight simple methods: minimum, maximum, product, average, simple majority, weighted majority, Naive Bayes and decision templates. These were selected with the idea to explore the potential of the ensemble beyond the traditional simple majority voting for Bagging and weighted majority voting for Boosting. We were interested whether diversity of the ensembles constructed by Bagging and Boosting would exhibit a relationship with the accuracy of some of the combination methods.

The rest of the paper is organized as follows. Section 2 explains Bagging and Boosting. Section 3 introduces the nine measures of diversity and the eight combination methods. The experiments are described in Section 4. Section 5 offers our conclusions.

2. Bagging and Boosting

2.1. Bagging

Bagging and Boosting are strategies for creating classifier ensembles, similar by the concept, yet with fundamental differences [9]. *Bagging* was proposed by Breiman [2] and extended further to *Arcing* [3,4] to accommodate the adaptive incremental construction of the ensemble which underlies the Boosting method (explained later). Bagging creates the classifiers in the ensemble by taking random samples with replacement (bootstrap sampling [7]) from the data set and building one classifier on each bootstrap sample. The final classification decision for an unlabeled data point $\mathbf{x}$ is made by taking the majority vote over the class labels produced by the L classifiers.

The true strength of Bagging is for *unstable* classifiers, such as neural networks and decision trees. Unstable classifiers are sensitive to small alterations in the data set. Thus, training the same classifier model on two slightly different training sets might result in substantially different classifiers. The classifiers might have similar overall accuracies but the parameters (e.g., the weights of the Neural Network) will differ, leading to a natural ensemble diversity. Ideally, this diversity will appear by the two classifiers recognizing correctly different objects from the data set, i.e., having "expertise" in different regions in the feature space. Bootstrap sampling is used to provide the random small alterations of the data set.

Bagging has been found to be inefficient for linear classifiers trained on large data sets, as these are *stable* classifiers [1,2,6,32]. This means that if linear classifiers (e.g., the nearest mean classifier, NMC) are trained on two very similar large data sets (e.g., bootstrap samples) the two classifiers will be virtually identical. Small differences in the data, will not lead to much difference in the estimates of the class means. So the inefficiency of Bagging for this case can be attributed to the lack of diversity in the ensemble. Linear classifiers might also become unstable if the training size is small. Then any alteration in the data set will have a major effect on the result, thus making the classifiers different from each other. The danger here is that, if we resort to very small training sample sizes (as we do in this study), the overall accuracy of the members of the ensemble will be low, and so will be the combined one. Thus the combination might not even reach the performance of a single linear classifier trained on the whole training data set.

2.2. Boosting

Boosting has been proposed and refined in a series of works by Freund and Schapire [8], leading to its most successful implementation called AdaBoost (Adaptive Bootstrapping). While Bagging relies on *random* and *independent* changes in the training data implemented by bootstrap sampling, Boosting advocates *guided* changes of the training data to direct further classifiers toward more "difficult cases". In this way, a certain desirable diversity is induced in the classifier team. At each step one classifier is added to the team. The training set for this classifier is obtained from the data set $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$, using a coefficient (*weight*) for each individual sample point. This coefficient at step k, say, $W_k(i)$, corresponds to the "difficulty" in recognizing point $\mathbf{z}_i$ by the previously added member of the ensemble. The higher the coefficient, the higher the difficulty. The coefficients are modified at each step. Boosting has two major implementations: Boosting by *resampling* and Boosting by *reweighting*. In the resampling version, the

[1] From "Bootstrap aggregating".
[2] From "Adaptive reweighting and combining".

coefficients $W_k(i)$ are used as a probability distribution on $\mathbf{Z}$ and the bootstrap sample is drawn from this distribution. Thus, multiple copies of the "difficult" points are likely to appear in the next training set, focusing the "expertise" of the classifier onto a problematic region in the feature space. In the reweighting implementation we assume that the training algorithm can take in individual data weights, and so we feed $W_k(i)$, and use the whole of $\mathbf{Z}$. There has been a debate about the merits of both implementations and their theoretical backup [1,3,9] and the evidence so far is inconclusive either way. In this study we use the reweighting method only.

There are many versions of the Boosting algorithm including variants of the classical AdaBoost. Below we describe the version used in the current study. Initially, all coefficients are set to $W_1(i) = 1/N$, $i = 1, \ldots, N$. We start with an empty classifier ensemble $\mathscr{D} = \emptyset$ and initialize the iterate counter $k = 1$. At iterate k, we first construct the classifier D_k to be added to the ensemble ($\mathscr{D} = \mathscr{D} \cup \{D_k\}$), and then calculate its error E_k using the weights W_k

$$E_k = \sum_{i=1}^{N} W_k(i)(1 - y_{i,k}), \tag{1}$$

where $y_{i,k} = 1$ if D_k gives the correct label of $\mathbf{z}_i$, and $y_{i,k} = 0$, otherwise. In a sense, $y_{i,k}$ is an "oracle" output, and is only applicable to the labeled data set $\mathbf{Z}$. Note the oracle output for later when diversity measures are considered. If $E_k = 0$ or $E_k \geqslant 0.5$, the weights $W_k(i)$ are reinitialized to 1. Next we calculate

$$\beta_k = \sqrt{\frac{1 - E_k}{E_k}}, \quad E_k \in (0, 0.5), \tag{2}$$

to be used in the weighted voting, and subsequently update the individual weights

$$W_{k+1}(i) = \frac{W_k(i)\beta_k^{(1-y_{i,k})}}{\sum_{j=1}^{N} W_k(j)\beta_k^{(1-y_{j,k})}}, \quad i = 1, \ldots, N. \tag{3}$$

The ensemble construction procedure terminates at iterate L, and the gradually built ensemble is $\mathscr{D} = \{D_1, \ldots, D_L\}$. The final decision for a new object $\mathbf{x}$ is made by weighted voting between the L classifiers. First, all classifiers give labels for $\mathbf{x}$ and then for all D_k that gave label ω_t, we calculate the support for that class by

$$\mu_t(\mathbf{x}) = \sum_{D_k(\mathbf{x}) = \omega_t} \ln(\beta_k). \tag{4}$$

The class with the maximal support is chosen for $\mathbf{x}$.

3. Combination methods

Let $\mathscr{D} = \{D_1, D_2, \ldots, D_L\}$ be the set of trained classifiers and $\Omega = \{\omega_1, \ldots, \omega_c\}$ be the set of class labels. Denote by $d_{i,j}(\mathbf{x})$ the support given by classifier D_i for the hypothesis that the given input $\mathbf{x}$ comes from class ω_j, $i = 1, \ldots, L$, $j = 1, \ldots, c$. The L classifier outputs $D_1(\mathbf{x}), \ldots, D_L(\mathbf{x})$ are then combined to get a label for $\mathbf{x}$. Depending on the type of the classifier outputs and the combination rule, we can get a soft final output $D(\mathbf{x}) = [\mu_1(\mathbf{x}), \ldots, \mu_c(\mathbf{x})]^{\mathrm{T}}$ or a crisp one $D(\mathbf{x}) \in \Omega$.

Here we consider eight combination methods which include the most popular choices: majority vote (Bagging), weighted majority vote (Boosting), average, minimum, maximum, product, Naive Bayes and decision templates [19].

For the majority vote combination, the class label assigned to $\mathbf{x}$ is the one that is most represented in the set of L crisp class labels obtained from $\mathscr{D}$. The weighted majority uses (4) with coefficients $\ln(\beta_k)$ calculated as in (2). In this case, E_k is the error of classifier D_k using (1) with $W_k(i) = 1/N$.

For the remaining simple combination methods,

$$\mu_j(\mathbf{x}) = \mathscr{O}\big(d_{1,j}(\mathbf{x}), \ldots, d_{L,j}(\mathbf{x})\big), \quad j = 1, \ldots, c, \tag{5}$$

where $\mathscr{O}$ is the respective operation (maximum, minimum, average or product). For the case of two classes, maximum is always equivalent to minimum [18]. [3] This equivalence is valid only if for any classifier D_i, $d_{i,1}(\mathbf{x}) + d_{i,2}(\mathbf{x}) = C$, where C is a constant. If $d_{i,j}$ are estimates of the posterior probabilities, then $C = 1$.

Naive Bayes combination considers crisp class labels $D_i(\mathbf{x}) \in \Omega$ and assumes that the probability that the true class is ω_k for a given $\mathbf{x}$ is proportional to the product of the probabilities $P(\omega_k|D_i(\mathbf{x}))$, $i = 1, \ldots, L$. The participating conditional probabilities are estimated from the training data.

Decision templates method uses soft class labels [19]. The classifier outputs can be conveniently organised in the so called decision profile as the matrix $\mathrm{DP}(\mathbf{x}) = [d_{i,j}(\mathbf{x})]$, $i = 1, \ldots, L$, $j = 1, \ldots, c$. Given L (trained) classifiers, c decision templates are calculated from the data, one template per class

$$\mathrm{DT}_i = \frac{1}{N_i} \sum_{\substack{\mathbf{z}_j \in \omega_i \\ \mathbf{z}_j \in \mathbf{Z}}} \mathrm{DP}(\mathbf{z}_j), \quad i = 1, \ldots, c, \tag{6}$$

where N_i is the number of elements of $\mathbf{z}$ from ω_i.

DT_i can be regarded as the expected $\mathrm{DP}(\mathbf{x})$ for class ω_i. The support for the class offered by the combination of the L classifiers, $\mu_i(\mathbf{x})$, is then found using a measure of (*dis*)*similarity* between the current $\mathrm{DP}(\mathbf{x})$ and DT_i, e.g.,

$$d_E(\mathrm{DP}(\mathbf{x}), \mathrm{DT}_i) = \sum_{j=1}^{c} \sum_{k=1}^{L} (d_{k,j}(\mathbf{x}) - dt_i(k, j))^2, \tag{7}$$

where $dt_i(k, j)$ is the k, j-th entry in decision template DT_i. Here we use the squared Euclidean distance but

[3] There were small differences in our experiments due to the random tie break.

Table 1
A summary of the eighth combination methods used

Method	Notation	Classifier outputs	Extra parameters
Majority vote	MAJ	Class labels	None
Weighted majority vote	WMAJ	Class labels	Weights for the classifiers
Naive Bayes	NB	Class labels	Conditional probabilities
Maximum	MAX	Soft labels	None
Minimum	MIN	Soft labels	None
Average	AVR	Soft labels	None
Product	PRO	Soft labels	None
Decision templates	DT	Soft labels	Decision templates

other measures of (dis)similarity can also be applied [17]. The highest similarity value will determine the class label of $\mathbf{x}$. With the Euclidean distance, we can think of the decision templates combination method as a NMC in the $L \times c$-dimensional space of the soft classifier outputs. The DT_i's are the means of the classes, and the winning label is determined by the distance to the nearest mean.

A summary of the eight classifier combination methods is given in Table 1.

4. Measures of diversity

Diversity may be interpreted differently, as suggested in the introduction. Hence, there are different diversity measures in the literature. Some of these measures, such as the Q-statistic and the correlation coefficient have come directly from mainstream statistics, others have their origins in software engineering and comparing of software versions, and yet another group of measures have been proposed specifically for the problems of multiple classifier systems.

4.1. Pairwise measures

The joint output of two classifiers, D_i and D_k, can be represented in a 2×2 table as shown in Table 2.

In this study we use four pairwise measures of diversity.

The Q-statistic (Q). Yule's Q-statistic [37] for two classifiers, e.g., D_i and D_k, is

$$Q_{i,k} = \frac{ad - bc}{ad + bc}. \tag{8}$$

Table 2
The 2×2 relationship table with probabilities

	D_k correct (1)	D_k wrong (0)
D_i correct (1)	a	b
D_i wrong (0)	c	d

Total, $a + b + c + d = 1$.

Q varies between -1 and 1, and for statistically independent classifiers it is 0. For a set of L classifiers, we calculate the averaged Q of all pairs.

The correlation coefficient (ρ). The correlation between two binary classifier outputs is

$$\rho_{i,k} = \frac{ad - bc}{\sqrt{(a+b)(c+d)(a+c)(b+d)}}. \tag{9}$$

For any two classifiers, Q and ρ have the same sign, and it can be proved that $|\rho| \leqslant |Q|$.

The disagreement measure (D) (used in [14,31])

$$\mathrm{D}_{i,k} = b + c. \tag{10}$$

The double-fault measure (DF) (used in [14,31])

$$\mathrm{DF}_{i,k} = d. \tag{11}$$

All these pairwise measures have been proposed as measures of similarity or dissimilarity (in a different context) in the numerical taxonomy literature (e.g., [33]).

4.2. Non-pairwise measures

Six non-pairwise measures of diversity are described below. Consider again the labeled data set $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$ sampled from the classification problem in question. Recall the "oracle" output of a classifier D_i and organize it as an N-dimensional binary vector $\mathbf{y}_i = [y_{1,i}, \ldots, y_{N,i}]^{\mathrm{T}}$, such that $y_{j,i} = 1$, if D_i recognises correctly $\mathbf{z}_j$, and 0, otherwise, $i = 1, \ldots, L$.

Kohavi–Wolpert variance (kw). Denote by $l(\mathbf{z}_j)$ the number of classifiers from $\mathscr{D}$ that correctly recognise $\mathbf{z}_j$, i.e., $l(\mathbf{z}_j) = \sum_{i=1}^{L} y_{j,i}$. Taking the formula for the variance from [15] and applying simple manipulations, the diversity measure becomes

$$\mathrm{kw} = \frac{1}{NL^2} \sum_{j=1}^{N} l(\mathbf{z}_j)(L - l(\mathbf{z}_j)). \tag{12}$$

The entropy measure (Ent). The highest diversity among classifiers for a particular $\mathbf{z}_j \in \mathbf{Z}$ is manifested by $\lfloor L/2 \rfloor$ of the votes in $\mathbf{y}_j$ with the same value (0 or 1) and the other $L - \lfloor L/2 \rfloor$ with the alternative value. If they all were 0's or all were 1's, there is no disagreement, and the classifiers cannot be deemed diverse. One possible measure of diversity based on this concept is

$$\mathrm{Ent} = \frac{1}{N} \sum_{j=1}^{N} \frac{1}{(L - \lceil L/2 \rceil)} \min \left\{ \sum_{i=1}^{L} y_{j,i}, L - \sum_{i=1}^{L} y_{j,i} \right\}. \tag{13}$$

Ent varies between 0 and 1, where 0 indicates no difference and 1 indicates the highest possible diversity. While value 0 is achievable for any number of classifiers L and any p, value 1 can only be attained for $p \in [(L-1)/2L, (L+1)/2L]$. Ent is similar up to a non-linear monotonic transformation to the entropy measure proposed in [5].

The measure of difficulty (θ). The idea for this measure came from a study by Hansen and Salomon [12]. We define a discrete random variable X taking values in $\{0/L, 1/L, \ldots, 1\}$ and denoting the proportion of classifiers in the ensemble that correctly classify an input $\mathbf{x}$ drawn randomly from the distribution of the problem. The measure of *difficulty*, θ, is defined as

$$\theta = \mathrm{Var}(X). \tag{14}$$

The higher the value of θ, the worse the classifier team.

The next two measures of diversity came from the literature on comparing different versions of software for analysis of reliability.

Generalised diversity (GD). This measure has been proposed in [28]. Let Y be a random variable expressing the proportion of classifiers (out of L) that *fail* on a randomly drawn object $\mathbf{x} \in \Re^n$. Denote by p_i the probability that $Y = i/L$. (Note that $Y = 1 - X$, where X is the variable introduced for θ). Denote by $p(i)$ the probability that i randomly chosen classifiers will fail on a randomly chosen $\mathbf{x}$. Then

$$p(1) = \sum_{i=1}^{L} \frac{i}{L} p_i, \tag{15}$$

and

$$\mathrm{p}(2) = \sum_{\mathrm{i}=1}^{\mathrm{L}} \frac{\mathrm{i}}{\mathrm{L}} \frac{(\mathrm{i}-1)}{(\mathrm{L}-1)} \mathrm{p_i}. \tag{16}$$

The generalised diversity measure, GD, is

$$\mathrm{GD} = 1 - \frac{p(2)}{p(1)}. \tag{17}$$

Coincident failure diversity (CFD). This is a modification of GD proposed in [27]

$$\mathrm{CFD} = \begin{cases} 0, & p_0 = 1.0; \\ \frac{1}{1-p_0} \sum_{i=1}^{L} \frac{L-i}{L-1} p_i, & p_0 < 1. \end{cases} \tag{18}$$

4.3. Grouping of the measures

Beside pairwise and non-pairwise, we can group the measures in two other ways. First, with respect to what the high values indicate, the measures are

- *Ascending*: measures looking for diversity: the higher the value the more diverse ($\uparrow$).
- *Descending*: measures looking for similarity: the higher the value the less diverse ($\downarrow$).

The second important grouping feature is the symmetry of the measures with respect to the correct and the incorrect outputs [30]. Intuitively, measures of diversity should be *symmetrical* (we use the symbol '■') because a set of identical classifiers should be classed as non-diverse, regardless of whether they are all correct or all wrong. That is, if the 0's (incorrect votes) and the 1's (correct votes) in the classifier outputs are swapped, the measure of diversity should be the same. Otherwise, the diversity measure is *non-symmetrical* (marked by '□').

Table 3 shows a summary of the nine measures of diversity including their types, and literature sources.

Table 4 shows the *toy example*. Given are $L = 2$ classifiers, each of accuracy 0.99. Assume that on the training set $\mathbf{Z}$ consisting of $N = 100$ labeled data points, the two classifiers err on two different objects. Then $a = 0.98$, $b = c = 0.01$, and $d = 0$ (refer to Table 2). The ranges of the measures for $L = 2$ classifiers are shown first *regardless of the individual accuracy*, and then *for an individual accuracy of* $p = 0.99$ as in our toy example [20]. The value of the measures for two independent classifiers each of accuracy $p = 0.99$ is shown in column three. The measures for the toy example are shown in the fourth column. At a first glance, the possible ranges and the diversity values for the example hold conflicting views. Judging by columns 3 and 6 only, some of the measures class the toy ensemble as highly diverse (compare the value to the *possible* ranges), (Q, DF, GD and CFD), another group of measures find the ensemble to be close to independent (ρ and θ), and a third group find it non-diverse (D, Ent, kw). Looking at the ranges

Table 3
Summary of the nine measures of diversity

Name	Notation	↑ / ↓	P/N[a]	S[b]	Source
Q-statistic	Q	(↓)	P	■	[37]
Correlation coefficient	ρ	(↓)	P	■	[33]
Disagreement measure	D	(↑)	P	■	[14,31]
Double-fault measure	DF	(↓)	P	□	[10]
Kohavi–Wolpert variance	kw	(↑)	N	■	[15]
Entropy measure	Ent	(↑)	N	■	[21]
Measure of difficulty	θ	(↓)	N	□	[12]
Generalised diversity	GD	(↑)	N	□	[28]
Coincident failure diversity	CFD	(↑)	N	□	[27]

[a] P stands for "pairwise", N stands for "non-pairwise".
[b] The column S shows symmetry: (■) *symmetrical* measures; (□) *non-symmetrical* measures.

Table 4
Ranges of the nine measures of diversity and values for the toy example ($p = 0.99$)

Notation	Type	Possible range for $L = 2$, any p	Restricted range for $L = 2$, $p = 0.99$	Independence value for $L = 2$, $p = 0.99$	Toy example for $L = 10$, $p = 0.99$
Q	(↓)	$[-1, 1]$	$[-1, 1]$	0.0000	−1.0000
ρ	(↓)	$[-1, 1]$	$[-0.0101, 1]$	0.0000	−0.0101
D	(↑)	$[0, 1]$	$[0, 0.0200]$	0.0198	0.0200
DF	(↓)	$[0, 1]$	$[0, 0.01]$	0.0001	0.0000
kw	(↑)	$[0, 0.5]$	$[0, 0.0050]$	0.0050	0.0050
Ent	(↑)	$[0, 1]$	$[0, 0.0200]$	0.0099	0.0200
θ	(↓)	$[0, 0.25]$	$[0, 0.0099]$	0.0050	0.0049
GD	(↑)	$[0, 1]$	$[0, 1]$	0.9900	1.0000
CFD	(↑)	$[0, 1]$	$[0, 1]$	0.9950	1.0000

for the specific p, however, the majority of the measures have marked the toy example as most diverse. This observation brings in the *Accuracy–Diversity* dilemma, stating generally that highly accurate classifiers cannot be very diverse. Indeed, this is demonstrated by the low values of D, Ent, and kw. The problem is that the interval of possible values of diversity is restricted by the high value of p. Some measures, however, do not share this problem, e.g., Q, DF, GD and CFD. There is no consensus on which group of measures is "better", which led us to use all the measures in our study, hoping to spot a relationship with the accuracy of the ensemble.

5. Experiments

We used the nine measures of diversity with the eight combination methods and the two ensemble building strategies: Bagging and Boosting. The experimental setup is described below. Table 5 contains a description of the seven datasets used.

1. *80-D correlated Gaussian data.* This is an 80-dimensional data set consisting of two Gaussian classes with equal covariance matrices; 500 vectors sampled from each class. The mean of the first class is zero for all the features. The mean of the second class is 6 for the first feature, and 0 for all the remaining features. The common covariance matrix has a variance of 41 for the first two features and a unit variance for all the other features. the covariances for x_1 and x_2 are 39, and all the remaining entries in the matrix are 0. Thus, there are two relevant features (needed jointly) and 78 irrelevant features.
2. *80-D rotated Gauss data.* In order to spread discrimination power evenly over all features, we have rotated the 80-dimensional correlated Gaussian data set in the whole 80-dimensional feature space. The rotation was performed by using a Hadamard matrix [11].

   The other five data sets:
3. Pima Indians Diabetes data;
4. Ionosphere data;
5. Wisconsin Diagnostic Breast Cancer data;
6. Sonar data; and
7. German data

are available at UCI Machine Learning Repository Database as indicated in Table 5. Table 6 shows the data sizes sampled as the training data from each data set. The testing was done on the remaining parts of the data sets. The results reported next are the average of 50 independent runs with each setup.

In all cases, 11 ensemble sizes L were used: $\{2, 3, 4, 5, 7, 10, 20, 50, 100, 200, 250\}$.

We wanted to keep the experiments as simple as possible and this led us to compromises and subsequent limitations:

1. All data sets have only two classes.
2. Two linear classifier models have been used as base classifiers: the NMC and the *pseudo-Fisher linear dis-*

Table 5
Summary of the seven two-class data sets used

Database	n	N	$\hat{P}_{\max}$ (%)	Past usage (%)	Availability
80-D correlated Gauss	80	1000	50.00	93	Delft
80-D rotated correlated Gauss	80	1000	50.00	93	Delft
Pima Indians Diabetes	8	768	65.10	80	UCI
Ionosphere	34	351	64.10	92	UCI
Wisconsin Diagnostic Breast Cancer	30	569	62.74	97.5	UCI
Sonar	60	198	56.06	90	UCI
German	24	1000	70.00	72	UCI

n: number of features; N: number of cases in the database; $\hat{P}_{\max}$: the largest class proportion; Past usage: shows the highest accuracy reported elsewhere; UCI: http://www.ics.uci.edu/~mlearn/MLRepository.html; *Delft*: The data is available from ⟨marina@ph.tn.tudelft.nl⟩.

Table 6
Training data sizes for the seven data sets

Correlated Gauss	3	5	10	15	20	30	40	50	100	200
Rotated correlated Gauss	3	5	10	15	20	30	40	50	100	200
Pima Indians Diabetes	3	4	7	10	20	50	100	200		
Ionosphere	3	5	10	17	30	50	100			
Wisconsin Diagnostic Breast Cancer	3	5	10	15	20	30	50	100	200	
Sonar	3	5	10	15	20	30	50	80		
German	3	5	10	12	15	20	30	50	100	200

criminant classifier [32] which might favor Boosting over Bagging. The classifiers that we used were unstable for small data sizes varying from three samples per class to the "critical" size (dimensionality of the feature space). The classifiers were stable for larger data sizes making Bagging inefficient for the latter case [32]. On the other hand, we measure diversity between the classifier *outputs*. Thus, if there is a strong relationship between diversity and accuracy it will show regardless of the classification model we used to produce the outputs.

3. The training set was constructed with equal amount of data point from each class, simulating equal prior probabilities, and this was not always close to reality.

We consider the strength of our study to be the scale of the experimentation: seven data sets, two classifier models, eight combination methods, nine diversity measures, 7–10 data sizes, and 11 ensemble sizes. The relatively small training sizes and our choice of simple base classifiers were prerequisites for a feasible experiment. The choice of the training sizes was dictated by the need to "destabilize" the base classifiers thereby making them suitable for Bagging and Boosting. Indeed, too simple classifiers trained on small data sets might generate poor members of the ensemble, and though diverse, they will not be much accurate altogether. This is one of the caveats in our study, hence we confine our conclusions to Bagging and Boosting of linear classifiers.

Before setting up a study on the effect of the data size and the number of the classifiers in the ensemble on the relationship between diversity and accuracy, we decided to pool the results together and look for a general pattern of relationship between diversity and accuracy. We put together all ensembles created by varying the training data size for the seven data sets (62 combinations altogether, the number of the non-empty cells in Table 6) the number of classifiers L (11 values) and the two classifier models. Thus, a total of $62 \times 11 \times 2 = 1364$ ensembles were designed by Bagging, and the same number designed by Boosting. On each ensemble, we calculated the accuracies of the 8 combination methods and the nine measures of diversity. Tables 7 and 8 show the correlation in % between the eight accuracies for Bagging and Boosting, respectively. Given the high

Table 7
Correlation in % between the accuracies of the eight combination methods for BAGGING

	WMAJ	AVR	MIN	MAX	PRO	NB	DT
MAJ	99	99	63	62	67	87	98
WMAJ		99	67	67	70	89	99
AVR			68	67	71	89	99
MIN				100	99	87	70
MAX					98	86	70
PRO						89	72
NB							90

Table 8
Correlation in % between the accuracies of the eight combination methods for BOOSTING

	WMAJ	AVR	MIN	MAX	PRO	NB	DT
MAJ	90	94	88	88	92	93	88
WMAJ		98	99	99	99	98	99
AVR			98	98	100	99	98
MIN				100	99	97	99
MAX					99	97	99
PRO						98	99
NB							98

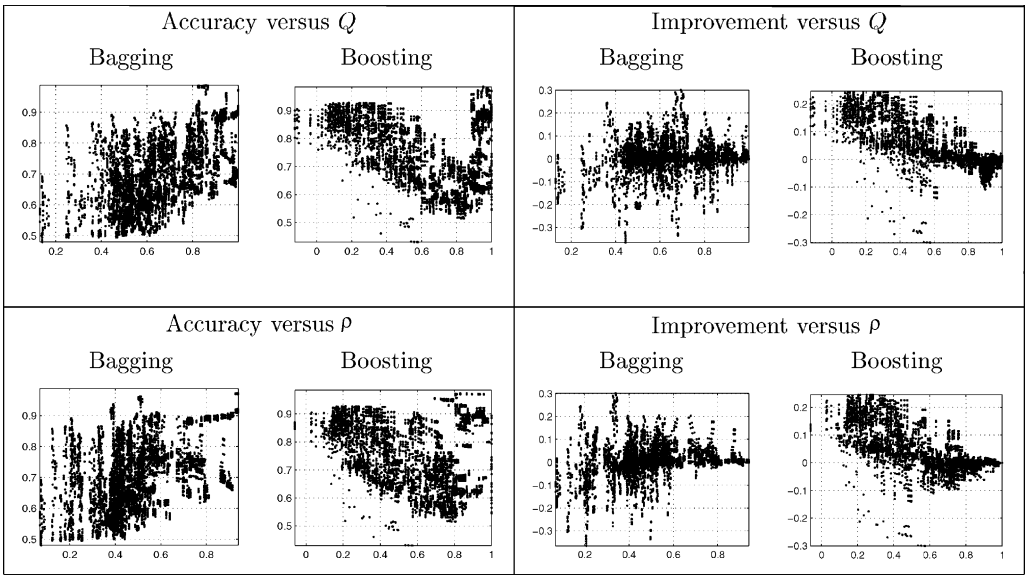

Fig. 1. (■) Symmetrical (↓) measures of diversity.

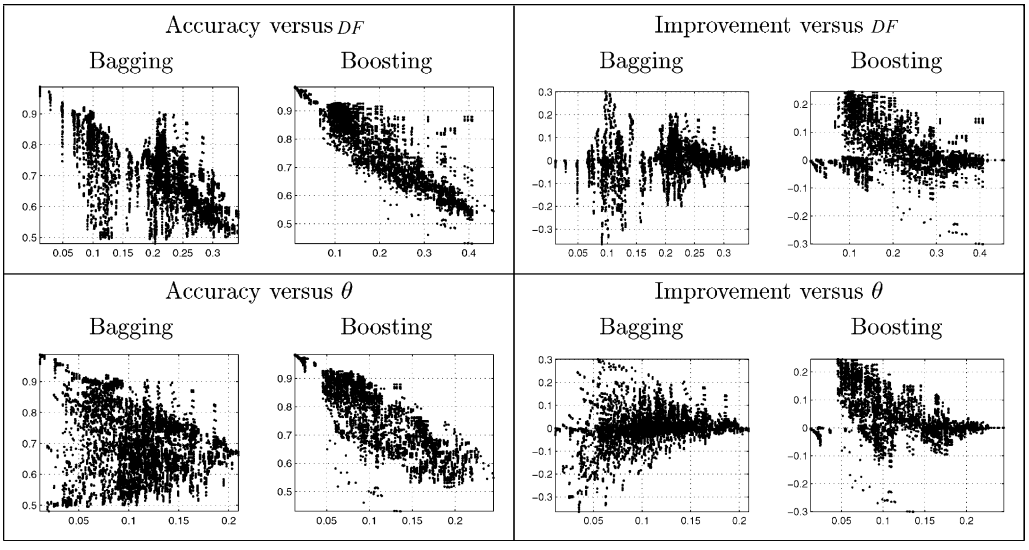

Fig. 2. (□) Non-symmetrical (↓) measures of diversity.

positive values, especially for Boosting, we decided to pool the results for the combination methods as well. This led to a total of 10,944 pairs of values diversity/accuracy for each diversity measure for Bagging, and the same amount for Boosting.

Figs. 1–4 show the scatterplots of the accuracy versus diversity for Bagging and Boosting. Each point in the plots corresponds to one ensemble. Together with the accuracy, we calculated the *improvement on the single* classifier trained on the entire (respective) training set. The resultant plots of the improvement versus diversity are also shown.

Figs. 5 and 6 show an excerpt from the results for the two traditional combination rules: the majority voting for Bagging and the weighted majority voting for Boosting, separately for the seven data sets. In Fig. 5, we have plotted the testing accuracy versus diversity measure Q for different values of the training samples. The line joins the diversity–accuracy points "starting" from the smallest training size (denoted by a star) and "ending" with the largest size (denoted by a gray circle). The coordinates of each points are the averages across all *ensemble sizes*, L (recall that each of these estimates is itself an average of 50 independent runs). Fig. 6 depicts the diversity–accuracy plots with respect to the ensemble sizes. The coordinates of each points are the averages across all training sizes. The figures show how the Bagging and Boosting manage to induce diversity, and whether the (traditional) ensemble accuracy benefits from this diversity.

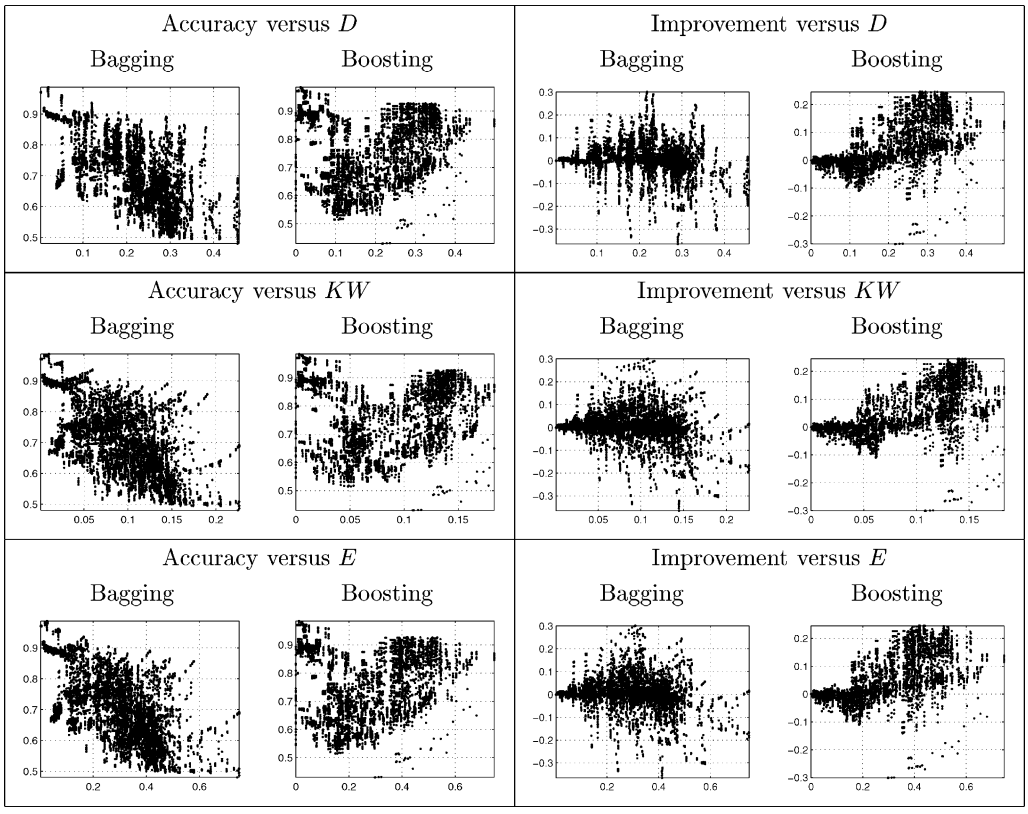

Fig. 3. (■) symmetrical (↑) measures of diversity.

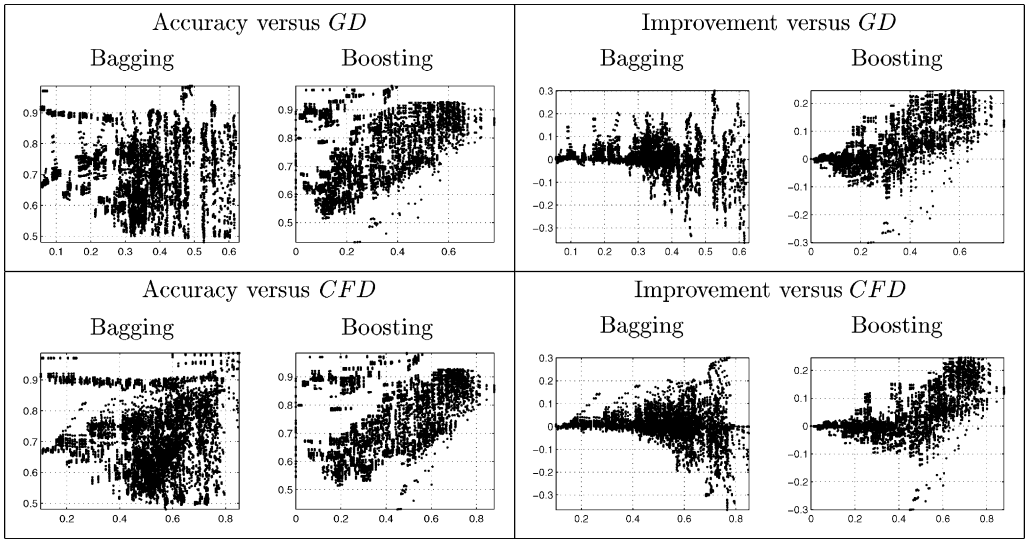

Fig. 4. (□) Non-symmetrical (↑) measures of diversity.

6. Discussion

The purpose of the experiment was to allow us to spot visually any pattern between diversity and accuracy that can guide our further studies. This is why we grouped the diversity measures once by their type ((↓) and (↑)) and second by the symmetry characteristic.

The first conspicuous observation from the scatterplots 1–4 is that there is no *strong* relationship between diversity and accuracy, whether it be linear or non-linear. There is however a general trend shown by the Boosted ensembles whereby higher diversity means *generally* higher accuracy and also higher improvement on the single accuracy. The most visually pleasing

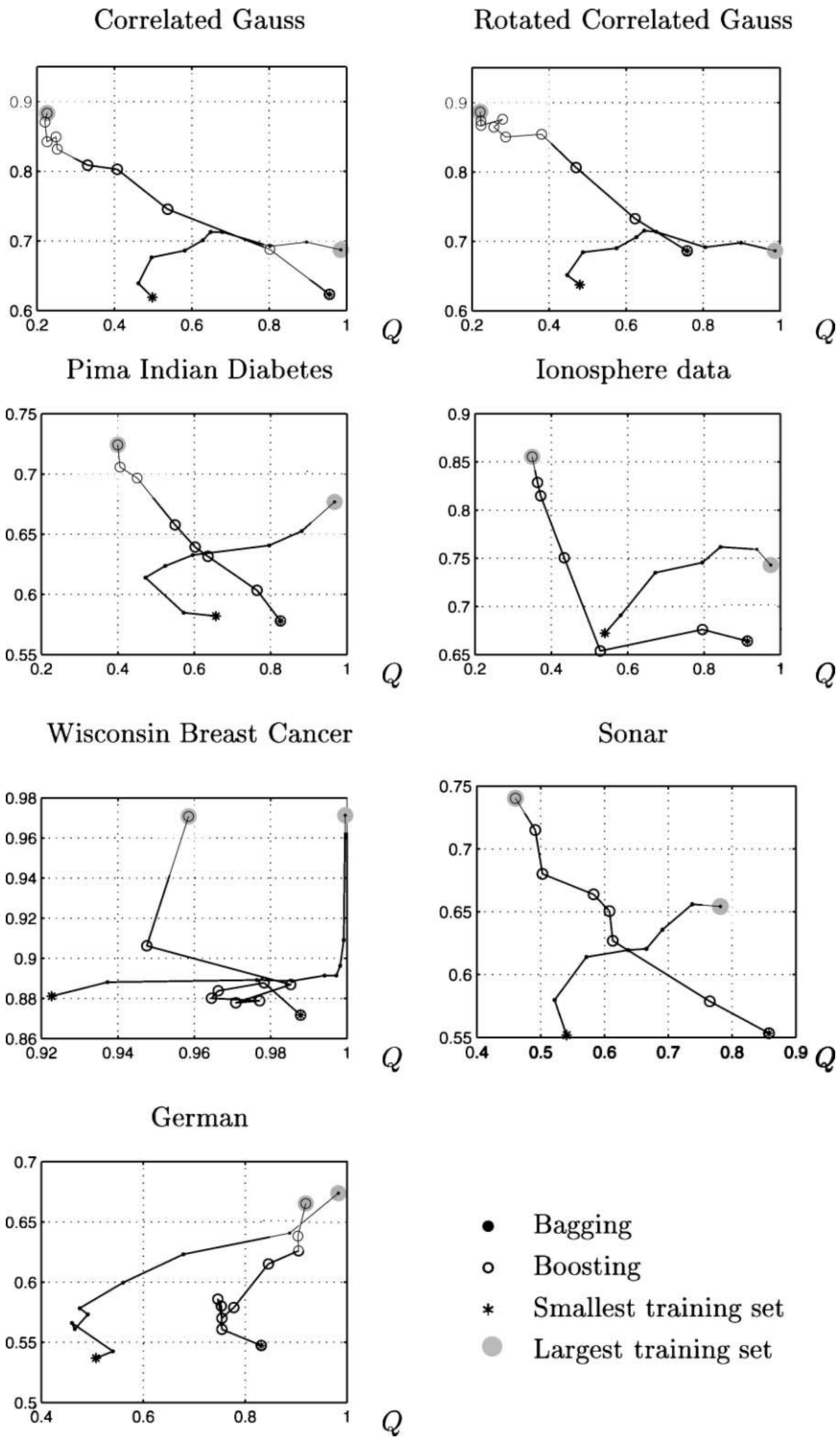

Fig. 5. Accuracy–diversity plots for training sizes (all L together).

relationship is found between DF and θ on the one hand and the Boosting accuracy on the other hand. The "clouds" of points are stretched along a negative slope expressing the relationship "the higher the diversity (low values of the measures), the higher the accuracy". However, this relationship is not mirrored on the right hand side where the *improvement* over the single classifier is plotted. We have to note here that DF and θ are non-symmetrical ($\square$) measures, which suggests that they are indirectly related to the accuracy of the team. Indeed, both measures have an accuracy connotation by definition. Whether or not this is a reasonable approach

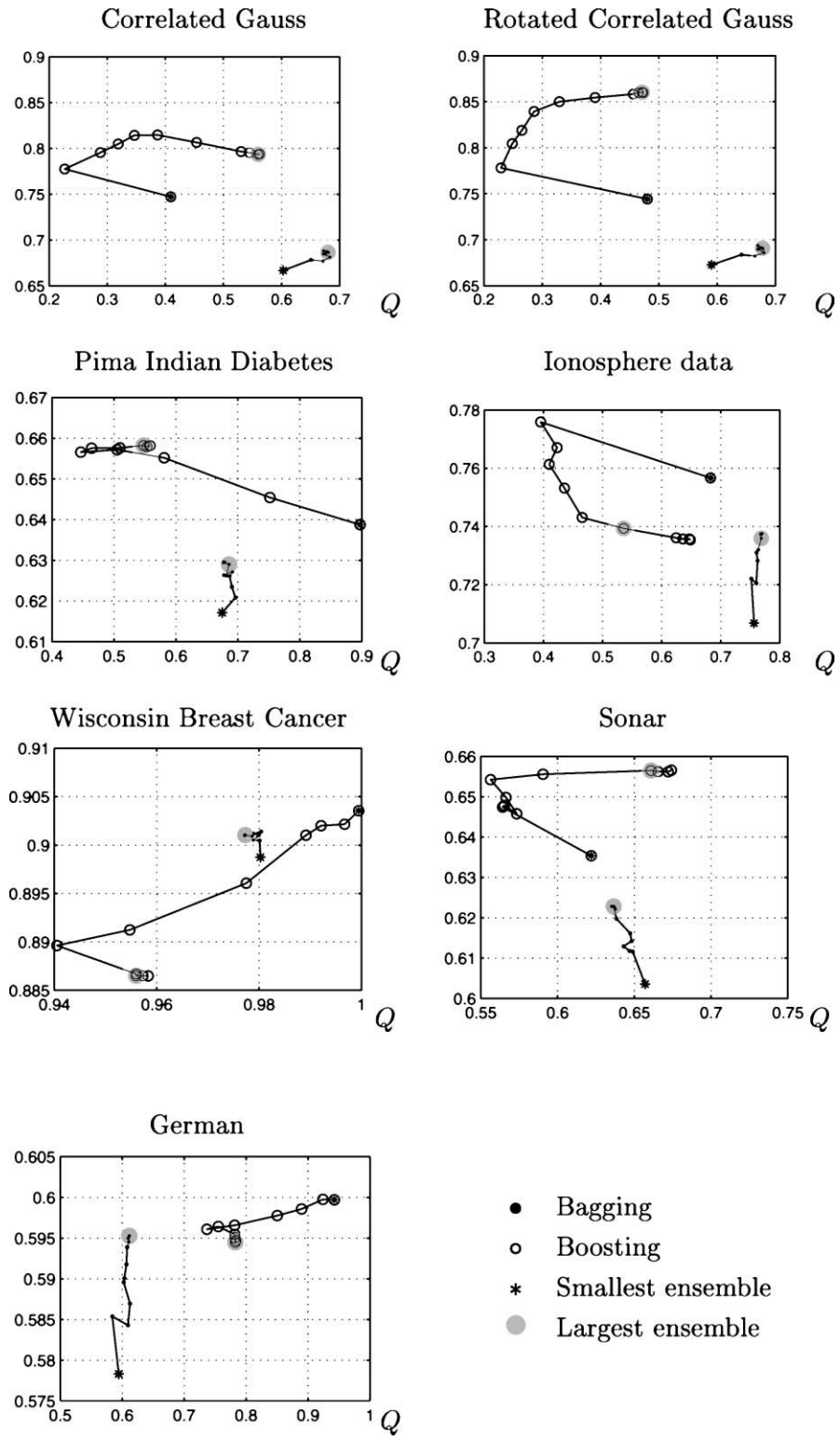

Fig. 6. Accuracy–diversity plots for ensemble sizes (all training sizes together).

for defining diversity remains an open issue. If we wanted a measurement of *accuracy*, why do we not measure the accuracy directly? Or why not use a simplified version of accuracy?

Boosting was expected to fare better because it *enforces* diversity by design, whereas Bagging has no such mechanism. It relies instead on the natural diversity of independent bootstrap samples.

Curiously, the accuracy trends for Bagging and Boosting are diametrically opposite for all symmetrical (■) diversity measures. For the group of non-symmetrical measures the Bagging accuracy plot has a peculiar cloud shape with little regularity in it. The opposite trends for the symmetrical measures suggest that *in the current experimental setup for Bagging*, the diverse ensembles consisted of weak classifiers, and so could not reach a reasonable ensemble accuracy, whereas non-diverse ensembles consisted of better classifiers and were more accurate altogether.

All diversity measures exhibit a "tornado" pattern for the improvement on the single accuracy for Boosting. At the lowest values of diversity, there is little or no scatter around the single accuracy: an ensemble of identical classifiers will not improve on the single classifier. The higher the diversity, the bigger the spread of the possible accuracies; some accuracies are below the single accuracy (due to "bad diversity") and majority of the accuracies are above the 0 line, indicating increasing improvement. Interestingly, the "tornado" pattern is present for the corresponding Bagging scatterplots but higher diversity in the ensemble is not coupled with higher accuracy; only the spread shows up. This suggests that we have not chosen a setup where Bagging is efficient, and possibly, another base classifier model would lead to a relationship pattern between diversity and accuracy similar to that for Boosting, however vague that might be.

The separate accuracy–diversity plots in Figs. 5 and 6 highlight the opposite ways in which Bagging and Boosting behave with respect to diversity. An almost consistent pattern through all data sets is observed in Fig. 5. With increasing the training size, Boosting makes use of the chance to induce diversity in the ensemble, whereas Bagging typically brings diversity down (Q approaching 1). Hence Boosting reaches higher accuracy except for the Wisconsin Breast Cancer data where the accuracies are very similar, and for the German data where Bagging achieves a slightly higher accuracy. For the German data, Boosting also fails to produce the pattern seen in the other plots. Interestingly, the plot for the Wisconsin Breast Cancer data also shows a setback in the increasing diversity, and Boosting levels up with the Bagging. The values of Q are noticeably higher than these for all other plots which shows the failure of the ensemble methods. For this data set, one single classifier build on a large training set will suffice (the end points of the Bagging line diagram, where $Q \approx 1$, signifying identical classifiers). Thus five of the seven plots show a clear behavioral pattern suggesting that Boosting benefits from larger training sizes by inducing diversity while Bagging drives the ensemble towards similar members of generally higher individual accuracy.

The relationship between diversity and accuracy with respect to L, the size of the ensembles, is rather blurred (Fig. 6). Our preliminary results (not displayed here) where plots were produced for a specific training size, did not show a consistent relationship either, for neither small nor large training sizes. Ideally, diversity should be helpful in deciding at what point adding new classifiers becomes *non-profitable* or even *deteriorating*. Such a pattern shows up for the Ionosphere data in Fig. 6 for the Boosting polygon. The most diverse ensemble had the highest accuracy, and increasing L beyond this point brought the accuracy down. The next best alternative is to be able to stop the training at a certain small L, assuming that the accuracy will not *improve much* if we add more classifiers. A certain tendency of this type appears on the plots for the Sonar data and the Pima Indian Diabetes data where the accuracy for Boosting levels off after a certain L is reached. On all the graphs in Fig. 6 Bagging shows little or no relationship with diversity when the ensemble size is concerned, so measuring diversity in this case is pointless.

7. Conclusions

This paper explores the relationship between diversity and accuracy on a large scale experiment. Bagging and Boosting have been nominated for the generation of ensembles of two models of linear base classifiers. Different sizes of the ensembles and the training data sets were considered using seven two-class data sets. Eight combination methods were applied with similar accuracies which led us to pool all the results so that we can plot and analyze a general "accuracy" versus different measures of diversity. We noted the conceptual differences between diversity measures available in the literature but found out later in the experiments that, with small exceptions, the measures worked rather in agreement. There were no strong relationships but some interesting patterns were identified and discussed in the text. It was found out that Boosting produced more diverse ensembles, and the general trend was that the higher the diversity, the higher the accuracy. At higher diversities, the improvement values are typically dispersed, with more of them on the positive side (accounting for "good diversity") but yet some on the negative side (due to "bad diversity"), indicating that highly diverse ensembles can be worse than the individual average.

The separate plots in Fig. 5 confirm the intuitive hypothesis that for linear base classifiers and large samples Boosting is better than Bagging. Our study suggested an answer to why this happens by looking at different training sizes and comparing the performances of the two ensemble building methods *with regard to their diversity*. Boosting works by inducing diversity (Q decreases and the accuracy of the ensemble increases)

whereas Bagging relies on good "all-round experts" and ultimately leads to ensembles of clones of a single most competent expert (Q is driven towards 1).

Two caveats. First, the conclusions here are valid for the chosen experimental setup and might differ if we used other classifier models and training sizes. Decision trees and neural networks are typical choices for base classifiers. For these choices, moderate data sets should be used because too small sets might lead to quick overtraining and poor generalization, and too large data sets will allow building a single good classifier, thereby dismissing the need for an ensemble. Training of a few thousand ensembles, some of which of 250 members, was not seen feasible resource-wise for our experimental plan.

Second, using the oracle classifier outputs $\mathbf{y}_i$ is only one way to try to quantify diversity. A drawback of this approach is that the diversity measures can be calculated only for *labeled* data. The next "level of detailization" of the classifier outputs (called "abstract level" in [36]) is to take the crisp labels and measure diversity on these. Some measures for this case have already been in use, and, in fact, we modified them to fit in our binary oracle-output model [5,15]. At the most detailed level ("measurement level" in [36]), the soft class labels can be used for calculating diversity. The most popular measure has been the correlation between $d_{i,j}$ and $d_{k,j}$ for class ω_j. A sum of correlations for ω_j weighted by the prior probabilities $P(\omega_j)$, $j = 1, \ldots, c$, has been used in [34,35] to derive a relationship between correlation and improvement on the accuracy when average combination method is used. The supposedly beneficial negative correlation has been the backbone of the negative correlation training of Neural Networks [24–26,29]. There is a spectrum of other possible measures of diversity for this case. It will be interesting to parallel the results found for oracle outputs with these at the abstract and measurement levels.

So, is the quest for quantifying and measuring diversity overemphasized? The lack of a general strong relationship would suggest a positive answer. However, the lack of a pattern might not be such a "negative" finding after all. Our study shows that diversity is *generally* beneficial but it is not a substitute for accuracy. And it need not be! So instead of trying to find an elusive relationship, maybe we should shift the focus of our study and consider diversity as an extra dimension in the search for better ensemble building methodologies.

References

[1] E. Bauer, R. Kohavi, An empirical comparison of voting classification algorithms: Bagging, boosting, and variants, Machine Learning 36 (1999) 105–142.

[2] L. Breiman, Bagging predictors, Machine Learning 26 (2) (1996) 123–140.

[3] L. Breiman, Arcing classifiers, The Annals of Statistics 26 (3) (1998) 801–849.

[4] L. Breiman, Combining predictors, in: A.J.C. Sharkey (Ed.), Combining Artificial Neural Nets, Springer-Verlag, London, 1999, pp. 31–50.

[5] P. Cunningham, J. Carney. Diversity versus quality in classification ensembles based on feature selection, Technical Report TCD-CS-2000-02, Department of Computer Science, Trinity College Dublin, 2000.

[6] T.G. Dietterich, Ensemble methods in machine learning, in: J. Kittler, F. Roli (Eds.), Multiple Classifier Systems, Lecture Notes in Computer Science, vol. 1857, Springer, 2000, pp. 1–15.

[7] B. Efron, R. Tibshirani, An Introduction to the Bootstrap, Chapman & Hall, NY, 1993.

[8] Y. Freund, R.E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, Journal of Computer and System Sciences 55 (1) (1997) 119–139.

[9] Y. Freund, R.E. Schapire, Discussion of the paper Arcing Classifiers by Leo Breiman, The Annals of Statistics 26 (3) (1998) 824–832.

[10] G. Giacinto, F. Roli, Design of effective neural network ensembles for image classification processes, Image Vision and Computing Journal 19 (9–10) (2001) 699–707.

[11] S.W. Golomb, L.D. Baumert, The search for Hadamard matrices, American Mathematics Monthly 70 (1963) 12–17.

[12] L.K. Hansen, P. Salamon, Neural network ensembles, IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (10) (1990) 993–1001.

[13] S. Hashem, B. Schmeiser, Y. Yih, Optimal linear combinations of neural networks: an overview, in: IEEE International Conference on Neural Networks, Orlando, Florida, 1994, pp. 1507–1512.

[14] T.K. Ho, The random space method for constructing decision forests, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (8) (1998) 832–844.

[15] R. Kohavi, D.H. Wolpert, Bias plus variance decomposition for zero-one loss functions, in: L. Saitta (Ed.), Machine Learning: Proc. 13th International Conference, Morgan Kaufmann, 1996, pp. 275–283.

[16] A. Krogh, J. Vedelsby, Neural network ensembles, cross validation and active learning, in: G. Tesauro, D.S. Touretzky, T.K. Leen (Eds.), Advances in Neural Information Processing Systems, vol. 7, MIT Press, Cambridge, MA, 1995, pp. 231–238.

[17] L.I. Kuncheva, Using measures of similarity and inclusion for multiple classifier fusion by decision templates, Fuzzy Sets and Systems 122 (3) (2001) 401–407.

[18] L.I. Kuncheva, A theoretical study on expert fusion strategies, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2) (2002) 281–286.

[19] L.I. Kuncheva, J.C. Bezdek, R.P.W. Duin, Decision templates for multiple classifier fusion: an experimental comparison, Pattern Recognition 34 (2) (2001) 299–314.

[20] L.I. Kuncheva, C.J. Whitaker, Ten measures of diversity in classifier ensembles: limits for two classifiers, in: Proc. IEE Workshop on Intelligent Sensor Processing, Birmingham, IEE, 2001, pp. 10/1–10/6.

[21] L.I. Kuncheva, C.J. Whitaker. Measures of diversity in classifier ensembles. Machine Learning, in press.

[22] L.I. Kuncheva, C.J. Whitaker, C.A. Shipp, R.P.W. Duin, Is independence good for combining classifiers?, in: Proc. 15th International Conference on Pattern Recognition, Barcelona, Spain, vol. 2, 2000, pp. 169–171.

[23] L. Lam, Classifier combinations: implementations and theoretical issues, in: J. Kittler, F. Roli (Eds.), Multiple Classifier Systems, Lecture Notes in Computer Science, vol. 1857, Springer, 2000, pp. 78–86.

[24] Y. Liu, X. Yao, Negatively correlated neural networks for classification, in: Proc. 3rd International Symposium on Artificial Life and Robotics (AROBIII'98), Japan, 1998, pp. 736–739.

[25] Y. Liu, X. Yao, Simultaneous learning of negatively correlated neural network, in: Proc 9th Australian Conference on Neural Networks (ACNN'98), Brisbane, Australia, 1998, pp. 183–187.
[26] Y. Liu, X. Yao, Ensemble learning via negative correlation, Neural Networks 12 (1999) 1399–1404.
[27] D. Partridge, W. Krzanowski, Distinct failure diversity in multiversion software, personal communication.
[28] D. Partridge, W.J. Krzanowski, Software diversity: practical statistics for its measurement and exploitation, Information and Software Technology 39 (1997) 707–717.
[29] B.E. Rosen, Ensemble learning using decorrelated neural networks, Connection Science 8 (3/4) (1996) 373–383.
[30] D. Ruta, B. Gabrys, Application of the evolutionary algorithms for classifier selection in multiple classifier systems with majority voting, in: J. Kittler, F. Roli (Eds.), Proc. Second International Workshop on Multiple Classifier Systems, Lecture Notes in Computer Science, vol. 2096, Springer-Verlag, 2001.
[31] D.B. Skalak, The sources of increased accuracy for two proposed boosting algorithms, in: Proc. American Association for Artificial Intelligence, AAAI-96, Integrating Multiple Learned Models Workshop, 1996.
[32] M. Skurichina, Stabilizing Weak Classifiers, Ph.D. thesis, Delft University of Technology, Delft, The Netherlands, 2001.
[33] P.H.A. Sneath, R.R. Sokal, Numerical Taxonomy, W.H. Freeman & Co, 1973.
[34] K. Tumer, J. Ghosh, Error correlation and error reduction in ensemble classifiers, Connection Science 8 (3/4) (1996) 385–404.
[35] K. Tumer, J. Ghosh, Linear and order statistics combiners for pattern classification, in: A.J.C. Sharkey (Ed.), Combining Artificial Neural Nets, Springer-Verlag, London, 1999, pp. 127–161.
[36] L. Xu, A. Krzyzak, C.Y. Suen, Methods of combining multiple classifiers and their application to handwriting recognition, IEEE Transactions on Systems, Man, and Cybernetics 22 (1992) 418–435.
[37] G.U. Yule, On the association of attributes in statistics, Philos. Trans., A 194 (1900) 257–319.