Published in:
Pattern Analysis and Applications
6, 2003, 22-31

Limits on the Majority Vote Accuracy in Classifier Fusion

L.I. Kuncheva, C.J. Whitaker, C.A. Shipp
School of Informatics,
University of Wales, Bangor, Gwynedd LL57 1UT, UK,
{l.i.kuncheva,c.j.whitaker,c.a.shipp}@bangor.ac.uk,

R.P.W. Duin
Faculty of Applied Sciences,
Delft University of Technology
P.O. Box 5046, 2600 GA Delft, The Netherlands,
bob@ph.tn.tudelft.nl

June 8, 2003

Limits on the Majority Vote Accuracy in Classifier Fusion

Abstract. We derive upper and lower limits on the majority vote accuracy with respect to individual accuracy p, the number of classifiers in the pool (L), and the pairwise dependence between classifiers, measured by Yule's Q statistic. Independence between individual classifiers is typically viewed as an asset in classifier fusion. We show that the majority vote with dependent classifiers can potentially offer a dramatic improvement both over independent classifiers and over the individual accuracy p. A functional relationship between the limits and the pairwise dependence Q is derived. Two patterns of the joint distribution for classifier outputs (correct/incorrect) are identified to derive the limits: the *pattern of success* and the *pattern of failure*. The results support the intuition that negative pairwise dependence is beneficial although not straightforwardly related to the accuracy. The pattern of success showed that for the highest improvement over p, all pairs of classifiers in the pool should have the same negative dependence.

Keywords: Classifier combination, classifier fusion, majority vote, limits on majority vote, independence and dependence, diversity.

1 Introduction

Let $\mathcal{D} = \{D_1, \ldots, D_L\}$ be a set (called also pool, team, ensemble, mixture, etc.) of classifiers such that $D_i : \Re^n \to \Omega$, where $\Omega = \{\omega_1, \ldots, \omega_c\}$, assigns $\mathbf{x} \in \Re^n$ a class label $\omega_j \in \Omega$. The majority vote method of combining classifier decisions, one of many methods in this important research area is to assign the class label ω_j to $\mathbf{x}$ that is supported by the majority of the classifiers D_i.

Finding independent classifiers is one aim of classifier fusion methods for the following reason. Let L be odd, $\Omega = \{\omega_1, \omega_2\}$, and all classifiers have the same classification accuracy p. The majority vote method with independent classifier decisions gives an overall correct classification accuracy calculated by the binomial formula

$$P_{maj} = \sum_{m=0}^{\lfloor L/2 \rfloor} \left(\begin{array}{c} L \\ m \end{array} \right) p^{L-m}(1-p)^m, \tag{1}$$

where $\lfloor a \rfloor$ denotes the largest integer less than or equal to a. The majority vote method with independent classifiers is guaranteed to give a higher accuracy than individual classifiers when $p > 0.5$ [12, 13]. The probability of a correct classification for $p = 0.6, 0.7, 0.8, 0.9$ and $L = 3, 5, 7, 9$ is shown in Table 1.

Table 1: Tabulated values of the majority vote accuracy of L independent classifiers with individual accuracy p

	$L = 3$	$L = 5$	$L = 7$	$L = 9$
$p = 0.6$	0.6480	0.6826	0.7102	0.7334
$p = 0.7$	0.7840	0.8369	0.8740	0.9012
$p = 0.8$	0.8960	0.9421	0.9667	0.9804
$p = 0.9$	0.9720	0.9914	0.9973	0.9991

Can we do better than that if the classifiers were dependent? The notion of *dependence* between classifiers can be perceived as *lack of independence* but there are various ways of further interpretation associated with diversity, orthogonality, complementarity, etc. [11,14]. It has been recognized that quantifying and studying the dependencies is an important issue in combining classifiers [11]. Numerous measures of dependence and diversity have been proposed in the literature. We can summarize the current results as follows:

1. When classifiers output estimates of the posterior probabilities $\hat{P}(\omega_s|\mathbf{x})$, and the outputs for each class are combined by averaging, or by an order statistic such as minimum, maximum or median, the classification error rate above the Bayes error (called the added error) depends on the correlation between the estimates (see [25,26]). Positively correlated classifiers only slightly reduce the added error, uncorrelated classifiers reduce the added error by a factor of $1/L$, and negatively correlated classifiers reduce the error even further.

2. When classifiers output class labels, the classification error can be decomposed into bias and variance terms [2, 9] or into bias and spread terms [3]. In both cases the second term accounts for the diversity of the ensemble. These results have been used to study the behavior of classifier ensembles in terms of the bias-variance trade-off.

3. For the case of classifier outputs in the form of a correct/incorrect vote, four levels of diversity are detailed in [22]: Level 1, where no more than one classifier is wrong on each data point. Level 2, where for each data point up to $\lfloor L/2 \rfloor$ could be wrong (the majority is always correct). Level 3, where at least one classifier is correct for each data point, and Level 4, where there might be points for which none of the classifiers is correct.

4. It is recognized that a negative correlation should be pursued when designing classifier ensembles. The negative correlation training of neural networks is one such method [15–17, 21].

Practically, there is no unique choice of a measure of diversity or dependence. There are pairwise measures which are calculated for each pair of classifiers in $\mathcal{D}$ and then averaged [4, 7, 8, 23–26]; measures that use the idea of entropy or correlation of individual outputs with the averaged output of $\mathcal{D}$ [2, 3, 9, 10, 18, 21], and also measures which base the calculations on the distribution of "difficulty" of the data points [5, 6, 14, 19, 20, 27].

The Yule's Q statistic is derived as the equivalent of the correlation coefficient for binary (correct/incorrect) valued measurements. So that positive Q values show positive dependency, negative values show negative dependency and zero shows no dependency. As with the correlation coefficient the range is from -1 to $+1$. Since the correlation coefficient is thought of as the most natural choice of a dependence measure for continuous-valued classifier outputs, we chose Q for the case of binary outputs.

Even though there is an abundance of diversity measures, there is also a notable lack of studies that relate diversity and accuracy. Here we are interested in establishing theoretical limits on the majority vote accuracy and finding a functional relationship between the limits and the diversity of the team. The main finding to date is that of Lam and Suen [12, 13] who have studied the accuracy of the majority vote method for the special case of equally accurate and *independent* classifiers for oddand even L. They find that the majority vote is guaranteed to do better than an individual classifier when the classifiers have an accuracy greater than 0.5. We extend the results of Lam and Suen for equally accurate classifiers and odd L but without the restriction that the classifiers be independent.

The paper follows two lines: the first one is based on a synthetic example where we demonstrate that accuracy of the majority vote over three classifiers, each of accuracy 60 %, can vary between 40 % and 90 %. The example also shows that it is impossible to identify a straightforward relationship between the Q statistic for the pairs of individual classifiers and and the majority vote. The second line of study defines and analyses two probability distributions over the combinations of correct/incorrect votes of L classifiers. In Section III the *pattern of success* and *pattern of failure* are defined. Based on these we obtain the upper and lower bounds on the majority vote accuracy P_{maj} as a function of the individual accuracy p and the number of classifiers L in the pool $\mathcal{D}$. We also calculate the pairwise dependences for the two patterns, as functions of p. Section IV offers an analysis, and Section V, our conclusions.

2 Dependence between classifiers

2.1 Q statistics for pairwise dependence

Let $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$ be a labeled data set, $\mathbf{z}_j \in \Re^n$ coming from the classification problem in question. For each classifier D_i we design an N-dimensional output vector $\mathbf{y}_i = [y_{1,i}, \ldots, y_{N,i}]^T$ of *correct classification*, such that $y_{j,i} = 1$, if D_i recognizes correctly $\mathbf{z}_j$, and 0, otherwise. There are various statistics to assess the similarity of D_i and D_k [1]. Yule [28] suggested that the Q statistic be used as a measure of association. The Q statistic for two classifiers is

$$Q_{i,k} = \frac{N^{11}N^{00} - N^{01}N^{10}}{N^{11}N^{00} + N^{01}N^{10}}, \tag{2}$$

where N^{ab} is the number of elements $\mathbf{z}_j$ of $\mathbf{Z}$ for which $y_{j,i} = a$ and $y_{j,k} = b$ (see Table 2).

Table 2: A 2×2 table of the relationship between a pair of classifiers

	D_k correct (1)	D_k wrong (0)
D_i correct (1)	N^{11}	N^{10}
D_i wrong (0)	N^{01}	N^{00}

Total, $N = N^{00} + N^{01} + N^{10} + N^{11}$.

For statistically **independent** classifiers, $Q_{i,k} = 0$. Q varies between -1 and 1. The correlation between two binary classifier outputs (correct/wrong) $\mathbf{y}_i$ and $\mathbf{y}_j$ is

$$\rho_{i,k} = \frac{N^{11}N^{00} - N^{01}N^{10}}{\sqrt{(N^{11} + N^{10})(N^{01} + N^{00})(N^{11} + N^{01})(N^{10} + N^{00})}}. \tag{3}$$

For any two classifiers, Q and ρ have the same sign, and it can be proved that $|\rho| \leq |Q|$. We chose Q to measure the dependency because it has been designed for 2 by 2 contingency tables. It is also simpler to calculate from the table entries.

2.2 A synthetic example

Let $\mathcal{D} = \{D_1, D_2, D_3\}$ and $N = |\mathbf{Z}| = 10$. We assume that all three classifiers have the same individual accuracy of correct classification, $p = 0.6$. This is manifested by each classifier labeling correctly 6 of the 10 elements of $\mathbf{Z}$. Given these requirements, *all* possible combinations of distributing 10 elements into the 8 combinations of outputs of the three classifiers are shown in Table 3. For a correct overall decision by the majority vote for some $\mathbf{z}_j \in \mathbf{Z}$, at least two of the three outputs $\mathbf{y}_i$ should be 1. The last column of Table 3 shows the majority vote accuracy of each of the 28 possible combinations. It is obtained as the proportion (out of 10 elements) of the sum of the entries in columns '111', '101', '011' and '110' (two or more correct votes). The best and the worst cases are highlighted in the table.

To clarify the entries in Table 3, consider as an example the first row. The number 2 in the column under the heading '101', displayed vertically, means that exactly 2 elements of $\mathbf{Z}$ are correctly recognized by D_1 and D_3 (the top and the bottom 1's of the heading) and misclassified by D_2 (the zero in the middle).

Table 3: All possible combinations of correct/incorrect classification of 10 objects by three classifiers so that each classifier recognizes exactly 6 objects. The entries in the table are the number of occurrences of the specific binary output of the three classifiers in the particular combination. The majority vote accuracy P_{maj} is shown in the last column.

No	1 1 1	1 0 1	0 1 1	0 0 1	1 1 0	1 0 0	0 1 0	0 0 0	P_{maj}
1	0	2	2	2	4	0	0	0	0.8
2	0	2	3	1	3	1	0	0	0.8
3	**0**	**3**	**3**	**0**	**3**	**0**	**0**	**1**	**0.9**
4	1	1	1	3	4	0	0	0	0.7
5	1	1	2	2	3	1	0	0	0.7
6	1	2	2	1	2	1	1	0	0.7
7	1	2	2	1	3	0	0	1	0.8
8	2	0	0	4	4	0	0	0	0.6
9	2	0	1	3	3	1	0	0	0.6
10	2	0	2	2	2	2	0	0	0.6
11	2	1	1	2	2	1	1	0	0.6
12	2	1	1	2	3	0	0	1	0.7
13	2	1	2	1	2	1	0	1	0.7
14	2	2	2	0	2	0	0	2	0.8
15	3	0	0	3	2	1	1	0	0.5
16	3	0	0	3	3	0	0	1	0.6
17	3	0	1	2	1	2	1	0	0.5
18	3	0	1	2	2	1	0	1	0.6
19	3	1	1	1	1	1	1	1	0.6
20	3	1	1	1	2	0	0	2	0.7
21	**4**	**0**	**0**	**2**	**0**	**2**	**2**	**0**	**0.4**
22	4	0	0	2	1	1	1	1	0.5
23	4	0	0	2	2	0	0	2	0.6
24	4	0	1	1	1	1	0	2	0.6
25	4	1	1	0	1	0	0	3	0.7
26	5	0	0	1	0	1	1	2	0.5
27	5	0	0	1	1	0	0	3	0.6
28	6	0	0	0	0	0	0	4	0.6

The table offers at least two interesting facts

- There is a case where the majority vote produces 90 % correct classification. Although purely hypothetical, this vote distribution is *possible* and offers a dramatic increase over the individual rate $p = 0.6$.

- Combining classifiers using the majority vote is beneficial or "neutral" in a great deal of the cases. In this example, in 12 of the 28 cases (42.9 %) the combined accuracy is greater than the limit for independent classifiers ($P_{maj} \geq 0.7$). For another 11 cases (39.3 %), the accuracy did not improve on the individual rate ($P_{maj} = p = 0.6$). In the remaining 5 cases (17.8 %) the overall accuracy was below the individual error rate ($P_{maj} < 0.6$). It is unknown which of these 28 distributions is most likely to occur in a real-life experiment. Therefore, even though most of the cases are no worse than the individual classifiers, improvement over p is *not guaranteed*.

For each pool $\mathcal{D}$, there are $L(L-1)/2$ pairs of classifiers. Denote by $Q_{i,j}$ the Q value for classifiers D_i and D_j. The Q statistic was calculated for each pair of classifiers for each of the 28 combinations. For the winning combination ($P_{maj} = 0.9$), $Q_{1,2} = Q_{2,3} = Q_{1,3} = -0.5$. For the worst case ($P_{maj} = 0.4$), $Q_{1,2} = Q_{2,3} = Q_{1,3} = 0.333$. Table 4 shows the sorted P_{maj} and the corresponding $Q_{1,2}, Q_{2,3}$ and $Q_{1,3}$. As can be seen in the table, there is no clear pattern of relationship between P_{maj} and the Q's. For a general observation, we averaged separately the Q's for all 12 combinations for which $P_{maj} > 0.648$[1] (favorable) and the 16 combination for which $P_{maj} \leq 0.648$ (unfavorable). The averaged Q of the favorable combinations is -0.1227, and that of the unfavorable combinations is 0.2873. However, the values of the Q's for both groups: favorable and unfavorable, are scattered over the whole range from -1 to 1, and extracting a consistent relationship does not seem to be possible.

The same type of synthetic experiment was carried out for $N = 50$. From the total of 3037 possible combinations, 1217 (40.0 %) have $P_{maj} > 0.648$ (favorable group). The worst part of the unfavorable group, i.e., with $P_{maj} < 0.6$, consisted of 874 (28.8 %) combinations. The averaged values of Q for the two groups are similar to the values in our previous example, -0.1200 for the favorable group and 0.2370 for the unfavorable one. Figure 1 displays the histograms of all Q's for the favorable and unfavorable groups of classifier teams. In the top two plots, one Q per ensemble was considered as the mean of the three pairwise Q's. In the bottom two plots, all pairwise Q's were pooled, so the total count is $3 \times 3037 = 9111$. *Generally*, the favorable Q's tend to be more on the negative side. Again, we have to emphasize that these experiments do not correspond to any real classification problem. Here we assumed that each possible distribution of votes occurs once (or with the same probability). In real-life problems we can expect only a small part of these distributions to appear, most probably distributions corresponding to positively dependent classifier outputs.

The simulation was run for $L = 3$ classifiers (any number of classes c) with $N = 10$, 20, and 30 and with individual accuracy $p = 0.6$, 0.7, 0.8 and 0.9. Table 5 shows the minimum and the maximum values of P_{maj}.

As a measure of overall dependence for a pool of three classifiers we took the maximum and the average of the three Q's, and these are shown in the last two columns of Table 4. The relationship between Q_{max} and P_{maj} is shown in the left plot of Figure 2, and between Q_{avr}

[1] see Table 1 for $p = 0.6$ and $L = 3$

Table 4: Sorted by P_{maj} combination from Table 3, the corresponding pairwise Q's.

No	P_{maj}	$Q_{1,2}$	$Q_{1,3}$	$Q_{2,3}$	Q_{avr}	Q_{max}
21	0.4	0.33	0.33	0.33	0.33	0.33
15	0.5	0.88	-0.50	-0.50	-0.04	0.88
17	0.5	0.33	-0.50	0.33	0.05	0.33
22	0.5	0.88	0.33	0.33	0.51	0.88
26	0.5	0.88	0.88	0.88	0.88	0.88
8	0.6	1.00	-1.00	-1.00	-0.33	1.00
9	0.6	0.88	-1.00	-0.50	-0.21	0.88
10	0.6	0.33	-1.00	0.33	-0.11	0.33
11	0.6	0.33	-0.50	-0.50	-0.22	0.33
16	0.6	1.00	-0.50	-0.50	0.00	1.00
18	0.6	0.88	-0.50	0.33	0.24	0.88
19	0.6	0.33	0.33	0.33	0.33	0.33
23	0.6	1.00	0.33	0.33	0.55	1.00
24	0.6	0.88	0.33	0.88	0.70	0.88
27	0.6	1.00	0.88	0.88	0.92	1.00
28	0.6	1.00	1.00	1.00	1.00	1.00
4	0.7	0.88	-1.00	-1.00	-0.37	0.88
5	0.7	0.33	-1.00	-0.50	-0.39	0.33
6	0.7	-0.50	-0.50	-0.50	-0.50	-0.50
12	0.7	0.88	-0.50	-0.50	-0.04	0.88
13	0.7	0.33	-0.50	0.33	0.05	0.33
20	0.7	0.88	0.33	0.33	0.51	0.88
25	0.7	0.88	0.88	0.88	0.88	0.88
1	0.8	0.33	-1.00	-1.00	-0.56	0.33
2	0.8	-0.50	-1.00	-0.50	-0.67	-0.50
7	0.8	0.33	-0.50	-0.50	-0.22	0.33
14	0.8	0.33	0.33	0.33	0.33	0.33
3	0.9	-0.50	-0.50	-0.50	-0.50	-0.50

and P_{maj}, in the right plot. Both are calculated on all possible combinations of three votes for $N = 50$ objects.

Figure 2 shows that P_{maj} is not strongly associated with either Q_{max} or Q_{avr}. However, it is possible to identify a threshold on each Q such that any combination of votes which has a "more negative" Q value (Q_{max} or Q_{avr}) belongs to the favorable group, i.e., such combinations are *better than a pool of independent classifiers*. Shown in Table 6 are the thresholds Q_{max}^{thr} or Q_{avr}^{thr} for $N = 10$, 20, and 30, and for $p = 0.6$, 0.7, 0.8 and 0.9.

This example motivated our further study on the relationship between p, Q, and P_{maj}.

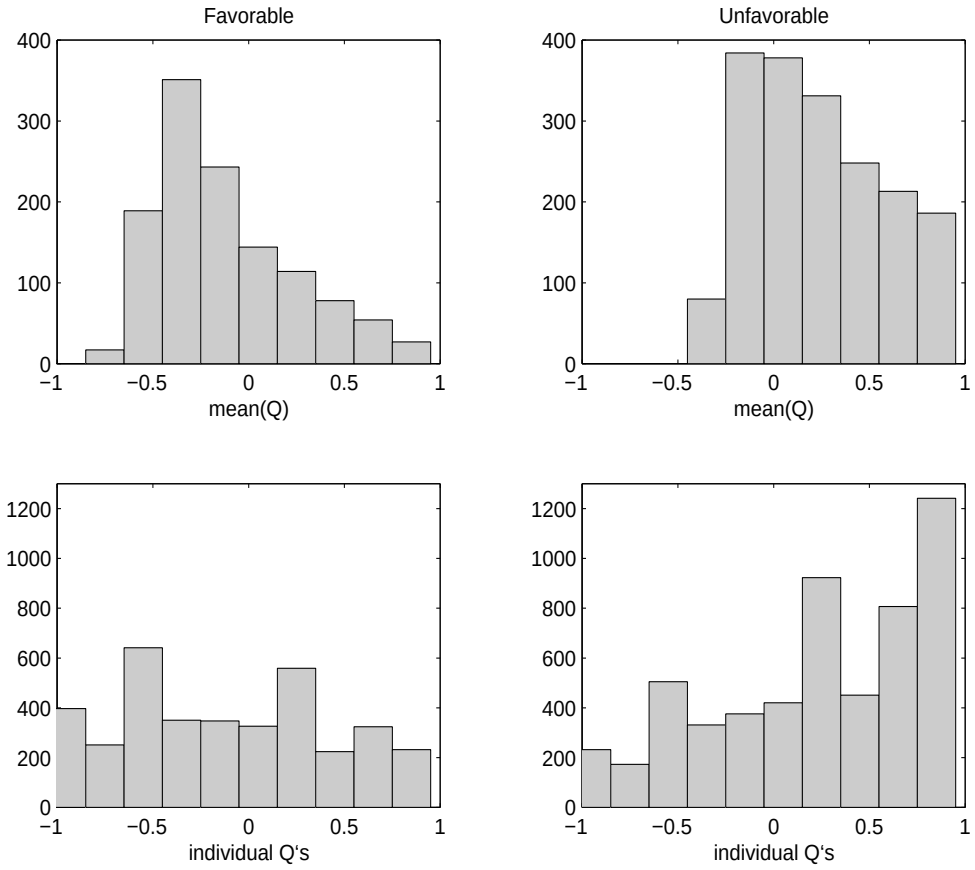

Figure 1: Histograms of the Q statistic for the "favorable" and "unfavorable" combinations of classifier outputs, $N = 50$.

Table 5: The minimum and the maximum values of the majority vote P_{maj} for $L = 3$ classifiers of accuracy p with N objects

p	$N = 10$		$N = 20$		$N = 30$	
	P_{max}	P_{min}	P_{max}	P_{min}	P_{max}	P_{min}
0.6	0.9	0.40	0.9	0.40	0.9	0.40
0.7	1.0	0.60	1.0	0.55	1.0	0.56
0.8	1.0	0.75	1.0	0.70	1.0	0.70
0.9	1.0	0.90	1.0	0.85	1.0	0.86

3 Limits on the majority vote accuracy

In the example in Section II we enumerated all possibilities of correct/incorrect votes for three classifiers and N objects. Here we define and analyze two probability distributions over the

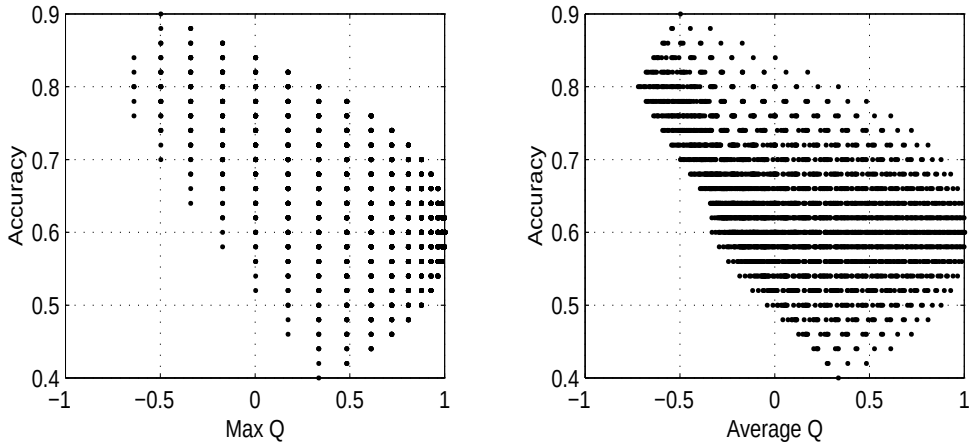

Figure 2: Plot of P_{maj} against Q_{max} and Q_{avr} for all possible combinations of 3 votes for $N = 50$ objects.

Table 6: Threshold dependence values guaranteeing that the combination is "favorable"

p	$N = 10$		$N = 20$		$N = 30$	
	Q_{avr}^{thr}	Q_{max}^{thr}	Q_{avr}^{thr}	Q_{max}^{thr}	Q_{avr}^{thr}	Q_{max}^{thr}
0.6	-0.375	-0.5	-0.375	-0.5	-0.43	-0.5
0.7	-0.63	-1.0	-0.45	-0.47	-0.52	-0.66
0.8	-1.0	-1.0	-0.6	-1.0	-0.7	-1.0
0.9	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0

possible combinations of L correct/incorrect votes.

3.1 The "pattern of success"

The three-classifier problem from the previous section can be visualized using two pairwise tables in Table 7 (see also Table 2)

This time we chose for convenience the entries in the table to be the probabilities of occurrence of the respective combination of correct and wrong outputs. For example, c is the probability of occurrence of the triple (011), i.e., D_1 wrong, D_2 correct, D_3 correct. Therefore,

$$a + b + c + d + e + f + g + h = 1. \tag{4}$$

The probability of correct classification of the majority vote of the three classifiers is (two or more correct)

Table 7: The probabilities in two 2-way tables illustrating a 3-classifier voting team.

D_3 correct (1)			D_3 wrong (0)		
$D_2 \rightarrow$			$D_2 \rightarrow$		
$D_1 \downarrow$	1	0	$D_1 \downarrow$	1	0
1	a	b	1	e	f
0	c	d	0	g	h

$$P_{maj} = a + b + c + e. \tag{5}$$

All three classifiers have the same individual accuracy p, which brings in the following three equations

$$\begin{array}{ll} a+b+e+f=p, & D_1 \text{ correct ;} \\ a+c+e+g=p, & D_2 \text{ correct ;} \\ a+b+c+d=p, & D_3 \text{ correct} \end{array} \tag{6}$$

Maximizing P_{maj} in (5) subject to conditions (4), (6) and $a, b, c, d, e, f, g, h \geq 0$, for $p = 0.6$, we obtain $P_{maj} = 0.9$ with the pattern highlighted in Table 3: $a = d = f = g = 0$, $b = c = e = 0.3$, $h = 0.1$. This example, optimal for 3 classifiers, indicates the possible characteristics of the best combination of L classifiers. The "pattern of success" and "pattern of failure" defined later follow the same intuition although we do not include in this study a formal proof for their optimality.

Consider the pool $\mathcal{D}$ of L (odd) classifiers, each with accuracy p. For the majority vote to give a correct answer we need $\lfloor L/2 \rfloor + 1$ or more of the classifiers to be correct. Intuitively, the best improvement over the individual accuracy will be achieved when exactly $\lfloor L/2 \rfloor + 1$ votes are correct. Any extra correct vote for the same $\mathbf{x}$ will be "wasted" because it is not needed to give the correct class label. Correct votes which participate in combinations not leading to a correct overall vote are also "wasted". To use the above idea we make the following definition

Definition 1. *The "pattern of success" is a distribution of the L classifier outputs for the pool $\mathcal{D}$ such that:*

1. *The probability of any combination of $\lfloor L/2 \rfloor + 1$ correct and $\lfloor L/2 \rfloor$ incorrect votes is α;*

2. *The probability of all L votes being incorrect is γ;*

3. *The probability of all other combinations is zero.*

For $L = 3$, the 2-table expression of the pattern of success is shown in Table 8.

Table 8: The *Pattern of Success*

D_3 correct (1)			D_3 wrong (0)		
$D_2 \rightarrow$			$D_2 \rightarrow$		
$D_1 \downarrow$	1	0	$D_1 \downarrow$	1	0
1	0	α	1	α	0
0	α	0	0	0	$\gamma = 1 - 3\alpha$

Here no votes are "wasted", the only combinations that occur are where all classifiers are incorrect or exactly $\lfloor L/2 \rfloor + 1$ are correct. To simplify notation, let $l = \lfloor L/2 \rfloor$. The probability of a correct majority vote (P_{maj}) for the pattern of success is the sum of the probabilities of each correct majority vote combination. Each such combination has probability α. There are $\begin{pmatrix} L \\ l+1 \end{pmatrix}$ ways of having $l+1$ correct out of L classifiers. Therefore

$$P_{maj} = \begin{pmatrix} L \\ l+1 \end{pmatrix} \alpha. \tag{7}$$

The pattern of success is only possible when $P_{maj} \leq 1$, i.e., when

$$\alpha \leq \frac{1}{\begin{pmatrix} L \\ l+1 \end{pmatrix}}. \tag{8}$$

If D_i gives a correct vote then the remaining $L-1$ classifiers must give l correct votes. There are $\begin{pmatrix} L-1 \\ l \end{pmatrix}$ ways in which the remaining $L-1$ classifiers can give this, each with probability α. So to have D_i with an overall accuracy p the following must hold

$$p = \begin{pmatrix} L-1 \\ l \end{pmatrix} \alpha. \tag{9}$$

Expressing α from (9) and substituting in (7) gives

$$P_{maj} = \frac{pL}{l+1}. \tag{10}$$

Feasible patterns of success have $P_{maj} \leq 1$, so (10) requires

$$p \leq \frac{l+1}{L}. \tag{11}$$

If $p > \frac{l+1}{L}$ then $P_{maj} = 1$ can be achieved, but there is an excess of correct votes to be distributed among combinations of classifiers with less than $l+1$ correct votes. The improvement over the individual p will not be as large as for the pattern of success but the majority vote accuracy will be 1 anyway. The final formula for P_{maj} is

$$P_{maj} = \min\left\{1, \frac{pL}{l+1}\right\}. \tag{12}$$

By definition, the pattern of success is symmetrical with respect to all classifiers. Hence all pairs of individual classifiers have the same 2-way tables, and therefore the same Q. Shown below is the 2-way table for the *pattern of success* containing the probabilities of correct/incorrect combinations of D_i and D_j from $\mathcal{D}$.

	$D_j \rightarrow$	
$D_i \downarrow$	1	0
1	$\begin{pmatrix} L-2 \\ l-1 \end{pmatrix}\alpha$	$\begin{pmatrix} L-2 \\ l \end{pmatrix}\alpha$
0	$\begin{pmatrix} L-2 \\ l \end{pmatrix}\alpha$	$1-3\alpha\begin{pmatrix} L-2 \\ l \end{pmatrix}$

The entries in the table are obtained by following similar patterns of combinatorial reasoning. For example, the probability that both D_i and D_j are correct is calculated by finding out the number of times D_i and D_j both cast correct votes in a "winning" combination. The number of possible combinations is the number of all combinations of $l-1$ classifiers (because D_i and D_j complete the required $l+1$ correct votes) out of the remaining $L-2$. Since the probability of any "winning" combination is α, the probability to have D_i and D_j both correct is $\begin{pmatrix} L-2 \\ l-1 \end{pmatrix}\alpha$. The same reasoning shows that when only one of D_i and D_j are correct, then for the other $L-2$ classifiers there must be l correct. As the probabilities in the four cells must sum to 1 then by using the fact that L is odd (and so equals $2l+1$), the probability that both D_i and D_j are incorrect is found.

Hence using (2) and (9) with these four probabilities, we obtain

$$Q = \frac{1-2p}{1-p}. \tag{13}$$

Note that Q is always negative for $p > 0.5$, and therefore the classifiers in the pattern of success are not independent but are *negatively dependent*. Finally

$$Q = \max\left\{-1, \frac{1-2p}{1-p}\right\}. \tag{14}$$

3.2 The "pattern of failure"

Definition 2. *The "pattern of failure" is a distribution of the L classifier outputs for the pool $\mathcal{D}$ such that:*

1. *The probability of any combination of $\lfloor L/2 \rfloor$ correct and $\lfloor L/2 \rfloor + 1$ incorrect votes is β;*

2. *The probability of all L votes being correct is δ;*

3. *The probability of all other combinations is zero.*

For $L = 3$, the 2-table expression of the pattern of failure is shown in Table 9.

Table 9: The *Pattern of Failure*

	D_3 correct (1)			D_3 wrong (0)		
	$D_2 \rightarrow$			$D_2 \rightarrow$		
$D_1 \downarrow$	1	0		$D_1 \downarrow$	1	0
1	$\delta = 1 - 3\beta$	0		1	0	β
0	0	β		0	β	0

The worst scenario is when the correct votes are "wasted", i.e., grouped in combinations of exactly l out of L correct (one short for the majority to be correct). The "excess" of correct votes needed to make up the individual p are also wasted by all the votes being correct together, while half of them plus one will suffice.

The probability of a correct majority vote (P_{maj}) is δ. As there are $\begin{pmatrix} L \\ l \end{pmatrix}$ ways of having l correct out of L classifiers, each with probability β, then

$$P_{maj} = \delta = 1 - \begin{pmatrix} L \\ l \end{pmatrix} \beta. \tag{15}$$

If D_i gives a correct vote then either all the remaining classifiers are correct (probability δ) or exactly $l - 1$ are correct out of the $L - 1$ remaining classifiers. For the second case there are $\begin{pmatrix} L-1 \\ l-1 \end{pmatrix}$ ways of getting this, each with probability β. To get the overall accuracy p for classifier D_i we sum the probabilities of the two cases

$$p = \delta + \begin{pmatrix} L-1 \\ l-1 \end{pmatrix} \beta. \tag{16}$$

Combining (15) and (16) gives

$$P_{maj} = \frac{pL - l}{l + 1}. \tag{17}$$

For values of individual accuracy $p > 0.5$, the pattern of failure is always possible.

As with the pattern of success, the pattern of failure is symmetrical with respect to all classifiers, by definition. Hence all pairs of individual classifiers have the same 2-way tables,

and therefore the same Q. The 2-way table in the case of L classifiers for the *pattern of failure* is shown below

	$D_i \rightarrow$	
$D_j \downarrow$	1	0
1	$1 - \binom{L}{l}\beta + \binom{L-2}{l-2}\beta$	$\binom{L-2}{l-1}\beta$
0	$\binom{L-2}{l-1}\beta$	$\binom{L-2}{l}\beta$

Using (2) and (16) with these four probabilities, we obtain

$$Q = \frac{2p-1}{p}. \tag{18}$$

4 Analysis

4.1 Best case (the pattern of success)

Using (12), Table 10 shows the individual accuracy required for a pool of $L = 3$, 5, 7, 9, and 11 classifiers so that $P_{maj} = 1$ is achievable. Interestingly, the largest individual accuracy needed to achieve $P_{maj} = 1$, is $p = 2/3 \approx 0.6667$ for *any* number of classifiers L. Beyond this value of p, the highest possible P_{maj} is 1, and there are "wasted" correct votes. Compare this result with the majority vote for *independent* classifiers in Table 1. Substituting the upper limits, the first entry ($p = 0.6, L = 3$) will be 0.9 and all the remaining will be 1's. Therefore it is theoretically possible to achieve a dramatic improvement over the individual accuracy and also over the independent vote accuracy, if we drop the independence requirement.

Table 10: Minimum individual accuracy p needed by a pool of $L = 3$, 5, 7, 9, and 11 classifiers so that $P_{maj} = 1$ is achievable.

L	3	5	7	9	11
p	0.6667	0.6000	0.5714	0.5556	0.5455

The dependence Q for the pattern of success (13), for any $p > 2/3$, will be -1. Eliminating p from (10) and (13), and solving for P_{maj}, we obtain

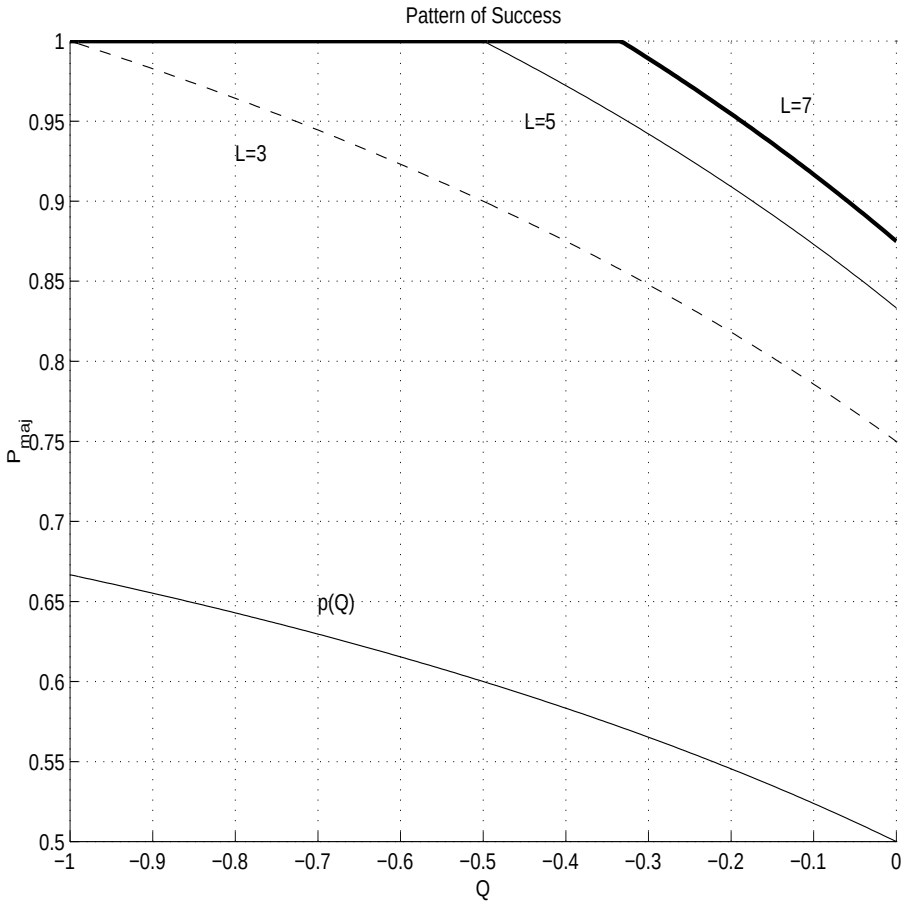

Figure 3: Upper limits: The majority vote accuracy P_{maj} and the individual accuracy p for the *pattern of success* as functions of Q

$$P_{maj} = \frac{L}{(l+1)} \frac{(1-Q)}{(2-Q)}. \tag{19}$$

The derivative of P_{maj} with respect to Q is $\partial P_{maj}/\partial Q = -\frac{1}{(l+1)(2-Q)^2}$, i.e., for the pattern of success, P_{maj} is a monotone decreasing function of the pairwise dependence Q. This result supports the intuition that *negatively* correlated classifiers are a better team than unrelated or positively related ones.

A family of curves for $L = 3$, 5 and 7, showing P_{maj} as functions of Q are plotted in Figure 3. Also plotted is p as a function of Q to illustrate the improvement over the single classifier.

4.2 Worst case (the pattern of failure)

When combining classifiers we always hope that the resultant P_{maj} will exceed the individual accuracy p. This is not guaranteed, however, as we showed in the synthetic example in Section

II. An analysis of (17) shows that for the pattern of failure, P_{maj} is a monotone decreasing function of L. Therefore, to find the smallest possible value of P_{maj} for a given p we take the limit in (17)

$$\lim_{L\to\infty} P_{maj} = 2p - 1. \tag{20}$$

Table 11 is a counterpart of Table 1, showing the smallest theoretically possible values of P_{maj}.

Table 11: Tabulated values of the minimal possible majority vote accuracy of L classifiers with individual accuracy p

	$L = 3$	$L = 5$	$L = 7$	$L = 9$
$p = 0.6$	0.4000	0.3333	0.3000	0.2800
$p = 0.7$	0.5500	0.5000	0.4750	0.4600
$p = 0.8$	0.7000	0.6667	0.6500	0.6400
$p = 0.9$	0.8500	0.8333	0.8250	0.8200

The dependence Q for the pattern of failure (18), for any $p > 0.5$, is positive. Eliminating p from (17) and (18), and solving for P_{maj}, we obtain

$$P_{maj} = \frac{L}{(l-1)}\frac{1}{(2-Q)} - \frac{l}{(L-l)}. \tag{21}$$

P_{maj} is a monotone increasing function of Q for the pattern of failure. A family of curves for $L = 3$, 5 and 7, showing P_{maj} and p as functions of Q for the pattern of failure are plotted in Figure 4. Again, $p(Q)$ is plotted for illustration of the decline in the performance in the pattern of failure.

The monotone increasing behavior of P_{maj} with respect to Q sounds counter-intuitive. The claim that diverse classifiers fare better than identical ones is not true for this case. Apparently there is some "bad" diversity which leads to deterioration of the performance. When this diversity is gradually "removed", the team accuracy reaches the individual p. For any $Q \in [0, 1)$ we can find a pattern of failure where $P_{maj} < p$ by enforcing this "bad" diversity. Consider for example three classifiers and three objects from $\mathbf{Z}$. Assume that each of the classifiers correctly recognizes one of the three objects and fails on the other two. If they all recognize the same object, there will be no diversity, and one of the three objects will be recognizes correctly. If the classifiers are diverse in a way that each classifier recognizes a different object, then the majority vote will be 0, worse than the single classifier. This example explains why increasing Q (reducing diversity) will increase P_{maj}. Most of the cases in real-life problems will fall between the two extreme patterns. Then the question is: Is it good to combine the classifiers or is it better to take the single best? And which diversity do we increase when we minimize Q or ρ for that matter, the "good" one or the "bad" one?

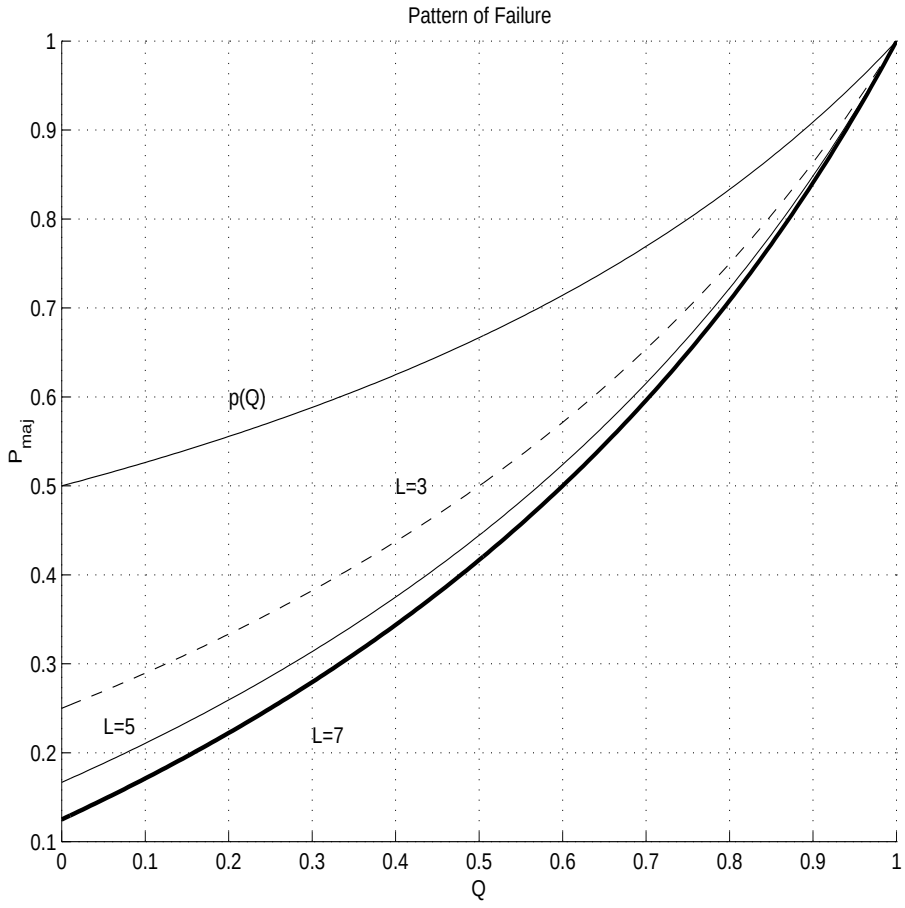

Figure 4: Lower limits: The majority vote accuracy P_{maj} and the individual accuracy p for the *pattern of failure* as functions of Q

4.3 General remarks

We can draw a parallel between the limits derived here and the abstraction called *Oracle* in classifier combination. Oracle assigns the correct class label to $\mathbf{x}$ if at least one classifier in the pool outputs the correct label. Therefore the probability of correct classification by the Oracle is

$$P_{Oracle} = 1 - P(\text{all wrong}). \tag{22}$$

In the pattern of success, $P(\text{all wrong})$ is γ, hence $P_{Oracle} = 1 - \gamma = P_{maj}$, i.e., the Oracle cannot offer any further improvement. In the pattern of failure, $P_{Oracle} = 1$, i.e., the Oracle is guaranteed to outperform any pattern of failure distribution.

It is impossible to tell which case we will face in practice. The most likely situation is to have reasonably accurate and positively dependent classifiers. This leads to a small improvement over the individual rate, usually not surpassing the majority vote over independent classifiers. As we show here, the limits for improvement (and also for deteriorating!) are substantially different from the individual rate. This could be used for designing a new strategy for *generating* the pool of classifiers $\mathcal{D}$. As shown in Table 10, a small number of not very accurate classifiers can achieve (in theory) $P_{maj} = 1$. Laying out such a strategy is not straightforward.

5 Conclusions

We derive an upper and a lower limit of the majority vote accuracy for individual classifiers, each one of accuracy p. The problem is explained using a synthetic example of three classifiers and finding all possible combinations of correct/incorrect votes on hypothetical data sets of 10 and 50 samples for $p = 0.6$. The results showed that the pairwise dependence plays an important although not clear-cut role for the final P_{maj}. We explored the problem by defining two extreme cases: the pattern of success and the pattern of failure. Each of these is a specific probability distribution over all possible combinations of correct/incorrect votes of the L classifier outputs from the pool $\mathcal{D}$. The pattern of success is when the correct votes are used in the most efficient way, whereas the pattern of failure is when most correct votes are "wasted". The equations connecting P_{maj}, p, L and Q (the pairwise dependence) have been derived for both cases and analyzed. We found that P_{maj} is a decreasing function of Q for the pattern of success and an increasing function for the pattern of failure, supporting the intuition that negatively related classifiers should be used.

The practical messages from this study can be summarized as follows

1. Independence of the classifiers in the team is not the best possible situation: the pattern of success is better. Both are unlikely to happen. If we should strive for independence, then it would be even better to look for dependent classifiers with a specific pattern of dependency.

2. Diversity is not always beneficial. As the pattern of failure shows, sometimes diversity may work toward deterioration of the performance.

3. There is no realistic framework for benchmarking classifier ensembles with either synthetic or real data. Some experimental setups might produce highly related classifiers while

others might be different, depending heavily on the design choices. Hence, enumerative examples such as the ones presented in this paper, though artificial, seem to be useful at this stage. It has been recognized that a benchmark framework for classifier combination is urgently needed.

References

[1] A.A. Afifi and S.P. Azen. *Statistical Analysis. A Computer Oriented Approach.* Academic Press, N.Y., 1979.

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

[3] L. Breiman. Combining predictors. In A.J.C. Sharkey, editor, *Combining Artificial Neural Nets*, pages 31–50. Springer-Verlag, London, 1999.

[4] G. Giacinto and F. Roli. Design of effective neural network ensembles for image classification processes. *Image Vision and Computing Journal*, 2000. (to appear).

[5] N. Griffith and D. Partridge. Self-organizing decomposition of functions in the context of a unified framework for multiple classifier systems. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems*, volume 1857 of *Lecture Notes in Computer Science*, pages 250–259, Cagliari, Italy, 2000. Springer.

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

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

[8] S. Impedovo and A. Salzo. A new evaluation method for expert combination in multiexpert system designing. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems*, volume 1857 of *Lecture Notes in Computer Science*, pages 230–239, Cagliari, Italy, 2000. Springer.

[9] R. Kohavi and D.H. Wolpert. Bias plus variance decomposition for zero-one loss functions. In L. Saitta, editor, *Machine Learning: Proc. 13th International Conference*, pages 275–283. Morgan Kaufmann, 1996.

[10] A. Krogh and J. Vedelsby. Neural network ensembles, cross validation and active learning. In G. Tesauro, D.S. Touretzky, and T.K. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 231–238. MIT Press, Cambridge, MA, 1995.

[11] L. Lam. Classifier combinations: implementations and theoretical issues. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems*, volume 1857 of *Lecture Notes in Computer Science*, pages 78–86, Cagliari, Italy, 2000. Springer.

[12] L. Lam and C.Y. Suen. Optimal combination of pattern classifiers. *Pattern Recognition Letters*, 16:945–954, 1995.

[13] L. Lam and C.Y. Suen. Application of majority voting to pattern recognition: An analysis of its behavior and performance. *IEEE Transactions on Systems, Man, and Cybernetics*, 27(5):553–568, 1997.

[14] B. Littlewood and D.R. Miller. Conceptual modeling of coincident failures in multiversion software. *IEEE Transactions on Software Engineering*, 15(12):1596–1614, 1989.

[15] Y. Liu and X. Yao. Negatively correlated neural networks for classification. In *Proc. 3d International Symposium on Artificial Life and Robotics (AROBIII'98)*, pages 736–739, Japan, 1998.

[16] Y. Liu and X. Yao. Simultaneous learning of negatively correlated neural network. In *Proc 9th Australian Conference on Neural Networks (ACNN'98)*, pages 183–187, Brisbane, Australia, 1998.

[17] Y. Liu and X. Yao. Ensemble learning via negative correlation. *Neural Networks*, 12:1399–1404, 1999.

[18] D. Opitz and J. Shavlik. A genetic algorithm approach for creating neural network ensembles. In A.J.C. Sharkey, editor, *Combining Artificial Neural Nets*, pages 79–99. Springer-Verlag, London, 1999.

[19] D. Partridge and W. Krzanowski. Distinct failure diversity in multiversion software. (personal communication).

[20] D. Partridge and W. J. Krzanowski. Software diversity: practical statistics for its measurement and exploitation. *Information & Software Technology*, 39:707–717, 1997.

[21] B.E. Rosen. Ensemble learning using decorrelated neural networks. *Connection Science*, 8(3/4):373–383, 1996.

[22] A.J.C. Sharkey and N.E. Sharkey. Combining diverse neural nets. *The Knowledge Engineering Review*, 12(3):231–247, 1997.

[23] 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.

[24] C.Y. Suen and L. Lam. Multiple classifier combination methodologies for different output levels. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems*, volume 1857 of *Lecture Notes in Computer Science*, pages 52–66, Cagliari, Italy, 2000. Springer.

[25] K. Tumer and J. Ghosh. Error correlation and error reduction in ensemble classifiers. *Connection Science*, 8(3/4):385–404, 1996.

[26] K. Tumer and J. Ghosh. Linear and order statistics combiners for pattern classification. In A.J.C. Sharkey, editor, *Combining Artificial Neural Nets*, pages 127–161. Springer-Verlag, London, 1999.

[27] W. Wang, P. Jones, and D. Partridge. Diversity between neural networks and decision trees for building multiple classifier systems. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems*, volume 1857 of *Lecture Notes in Computer Science*, pages 240–249, Cagliari, Italy, 2000. Springer.

[28] G.U. Yule. On the association of attributes in statistics. *Phil. Trans., A*, 194:257–319, 1900.