Influence of Resampling and Weighting on Diversity and Accuracy of Classifier Ensembles

R.M. Valdovinos[1], J.S. Sánchez[2], and E. Gasca[1]

[1] Lab. Reconocimiento de Patrones, Instituto Tecnológico de Toluca
Av. Tecnológico s/n, 52140 Metepec. México
[2] Dept. Llenguatges i Sistemes Informàtics, Universitat Jaume I
Av. Sos Baynat s/n, E-12071 Castelló de la Plana. Spain

Abstract. Diversity in the decisions of a classifier ensemble appears as one of the main issues to take into account for its construction and operation. However, the potential relationship between diversity and accuracy, with respect to the resampling method and/or the classifier fusion technique has not been clearly proved. The present paper analyzes the influence of different resampling methods and dynamic weighting schemes on diversity and how this can affect to the accuracy of the classifier ensemble. This is specifically studied in the framework of the Nearest Neighbor classification algorithm.

1 Introduction

Many researchers have investigated the technique of combining the predictions of multiple classifiers to produce a single decision. The basic idea considers a number of advantages when compared to the use of an individual classifier [10], the most important argues that the resulting classifier (typically called *ensemble*) is generally more accurate than any of the single classifiers making up the ensemble.

It is widely accepted that the major factor for a better accuracy is the diversity among the classifiers to be combined, that is, they must differ in their decisions to complement each other [2, 13, 15]. To obtain diversity, there exist many distinct techniques for constructing classifier ensembles. One consists of using different classifiers over a unique training set; in this case, the classifiers themselves must be different enough to produce diverse decisions. Another consists of manipulating (or resampling) the data set on which the classifiers are trained. Under this scenario, classifiers may be all based upon the same technique, e.g., a k-Nearest Neighbor (k-NN) classifier.

Another issue of interest in the framework of combining classifiers refers to the different methods to obtain the output of an ensemble. Two main strategies are discussed in the literature: classifier selection and classifier fusion. The idea in classifier selection is that each individual classifier has expertise in some local area of the feature space and therefore, only one expert is responsible to label a new input pattern. Conversely, classifier fusion assumes that all classifiers have equal knowledge of the whole feature space and the decisions of all of them are taken into account for any input pattern.

Within the fusion context, the most popular method for combining the decisions corresponds to the majority voting [12]; however when the performance of the ensemble

J. Martí et al. (Eds.): IbPRIA 2007, Part II, LNCS 4478, pp. 250–257, 2007.

components is not uniform, the efficiency of this type of voting results affected negatively. More elaborated schemes employ weighted voting rules, in which each individual classifier is associated with a different weight [14, 18]. Importance of this comes from the fact that a choice of an appropriate fusion strategy can improve further on the performance of the ensemble.

This study mainly concentrates on establishing the relationship, if any, between diversity and the techniques used for constructing the classifier ensemble, that is, how is diversity affected by the resampling method and/or the fusion strategy? Also, we are interested in knowing the empirical effects of diversity on accuracy, that is, does lower (higher) diversity really imply lower (higher) overall accuracy performance?

In order to address these questions, we here use different resampling methods existing in the literature: selection without replacement [3], Bagging [4], Boosting [8], and Arc-x4 [5]. With respect to the fusion strategies, we introduce a number of techniques to weight the individual decisions dynamically. Finally, we use four measures of diversity: Q-statistics, correlation coefficient, disagreement measure, and variability measure. While the first three correspond to well-known measures properly adopted from the Statistics literature, the latter is a new measure proposed in this paper.

2 Measures of Diversity

Let $\mathcal{D} = \{D_1, \ldots, D_L\}$ be a set of L classifiers, and $\Omega = \{\omega_1, \ldots, \omega_c\}$ be a set of c classes. Each classifier D_i $(i = 1, \ldots, L)$ gets as input a feature vector $\mathbf{x} \in \Re^d$, and assigns it to one of the c problem classes. The output of an ensemble of classifiers is an L-dimensional vector $r = [D_1(\mathbf{x}), \ldots, D_L(\mathbf{x})]^{\mathrm{T}}$ containing the decisions of each classifier.

In the last years, numerous measures of diversity have been proposed in the literature, most of them being adapted from existing statistical measures. In practice, these measures can be categorized into two groups [13]: pairwise measures and non-pairwise measures. The pairwise measures are computed for each pair of classifiers in $\mathcal{D}$ and then averaged. The non-pairwise measures ether use the concept of entropy or correlation of individual outputs with the averaged output of $\mathcal{D}$ or are based on the distribution of "difficulty" of the data points. In this work, we concentrate on three pairwise measures taken from the literature, Q-statistics, correlation coefficient and disagreement measure. Moreover, a new non-pairwise measure, here called *variability measure*, will be proposed in the present section.

2.1 The *Q*-Statistics

Let $Y = \{y_1, y_2, \ldots, y_N\}$ be a set of labelled data. For two classifiers D_i and D_j, the Q-statistics is defined as

$$Q_{i,j} = \frac{ad - bc}{ad + bc} \tag{1}$$

where a is the number of elements in Y correctly classified by D_i and D_j, b is the number of elements correctly classified by D_i but not by D_j, c is the number of

elements wrongly classified by D_i and correctly classified by D_j, and d is the number of elements wrongly classified by D_i and D_j. Then, $N = a + b + c + d$.

For a set of L classifiers, the averaged Q-statistics over all pairs of classifiers can be expressed as [13]

$$Q_{ave} = \frac{2}{L(L-1)} \sum_{i=1}^{L-1} \sum_{j=i+1}^{L} Q_{i,j} \tag{2}$$

For statistically independent classifiers (and $N \to \infty$), $Q_{i,j} = 0$. Q varies between -1 and $+1$. Classifiers that tend to classify the same objects correctly will have positive values of Q, while those which err on different objects will obtain negative values.

2.2 The Correlation Coefficient

This measure allows to quantify the relation between a pair of classifiers D_i and D_j.

$$\rho_{i,j} = \frac{ad - bc}{\sqrt{(a+b)(c+d)(a+c)(b+d)}} \tag{3}$$

For any two classifiers, ρ is between -1 and $+1$. Both -1 and $+1$ represent a total correlation between D_i and D_j, while $\rho_{i,j} = 0$ means that the pair of classifiers are not correlated at all. On the other hand, Q and ρ have the same sign, and it can be proved that $|\rho| \leq |Q|$.

2.3 The Disagreement Measure

This measure is the ratio between the number of elements correctly classified by one classifier and wrongly by the other to the total number of elements [17, 13]. For two classifiers D_i and D_j, the disagreement measure varies between 0 and $+1$.

$$Dis_{i,j} = \frac{b + c}{a + b + c + d} \tag{4}$$

2.4 The Variability Measure

This corresponds to a new diversity measure proposed in this paper. Unlike the previous measures, this makes use of a decision matrix to store the class labels given by the L classifiers to each object. The measure can be defined as the proportion of the cases that have received different decisions, that is, at least one classifier disagrees with the rest of classifiers.

$$\upsilon = \frac{\sum_{i=1}^{N} \lambda}{N} \tag{5}$$

where $\lambda = 0$ if $D_1(\mathbf{y}) = D_2(\mathbf{y}) = \cdots = D_L(\mathbf{y})$, and $\lambda = 1$ otherwise.

The variability measure varies between 0 and $+1$. For $L = 2$ and $c = 2$, the variability measure matches the disagreement measure, that is, $\upsilon = Dis_{i,j}$.

3 Some Classifier Fusion Techniques

The simplest way to combine multiple classifiers is by *voting*, which corresponds to take a linear combination of the classifiers. Let w_j be the weight of the j classifier D_j, then the final output of the ensemble is computed as

$$r = \sum_{j=1}^{L} w_j D_j(\mathbf{x}) \tag{6}$$

where $\forall j, w_j \geq 0$ and $\sum_{j=1}^{L} w_j = 1$.

If each classifier just provides the class of the input pattern **x**, then one can only have the *simple majority voting* where all classifiers have equal weight $w_j = 1/L$. If the classifiers can also supply additional information, then their votes can be weighted [14, 18], for example by a function of their distance to the input pattern. In this section, we introduce several weighting functions for classifier ensembles; some of them are taken from the Pattern Recognition literature and conveniently adapted to combine multiple classifiers, while others are now proposed for the first time.

A voting rule for the k-NN classifier, in which the votes of different neighbors are weighted by a function of their distance to the input pattern **x**, was first proposed by Dudani [7]. A neighbor with smaller distance is weighted more heavily than one with a greater distance: the nearest neighbor gets a weight of 1, the furthest neighbor a weight of 0, and the other weights are scaled linearly to the interval in between.

$$w_j = \begin{cases} \frac{d_k - d_j}{d_k - d_1} & \text{if} \quad d_k \neq d_1 \\ 1 & \text{if} \quad d_k = d_1 \end{cases} \tag{7}$$

where d_j denotes the distance of the j'th nearest neighbor, d_1 is the distance of the nearest neighbor, and d_k indicates the distance of the furthest (k'th) neighbor.

In order to employ this weighting function in the context of classifier fusion, the value of k (i.e., the number of neighbors in Dudani's rule) can be here replaced by the number of classifiers L that constitute the ensemble. Moreover, the L distances of **x** to its nearest neighbor in each individual classifier have to be sorted in increasing order $(d_1, d_2, \ldots, d_L)$. Thus, the original Dudani's weight (Eq. 7) can be now rewritten as:

$$w(D_j) = \begin{cases} \frac{d_L - d_j}{d_L - d_1} & \text{if} \quad d_L \neq d_1 \\ 1 & \text{if} \quad d_L = d_1 \end{cases} \tag{8}$$

where d_1 denotes the shortest of the L distances of **x** to the nearest neighbor, and correspondingly d_L is the longest of those distances.

Dudani further proposed the *inverse distance weight* [7], which can be expressed as follows:

$$w(D_j) = \frac{1}{d_j} \quad \text{if} \quad d_j \neq 0 \tag{9}$$

Another weighting function proposed here is based on the work of Shepard [16], who argues for a universal perceptual law which states that the relevance of a previous

stimulus for the generalization to a new stimulus is an exponentially decreasing function of its distance in psychological space. This gives the weighted voting function of Eq. 10, where α and β are constants and determine the slope and the power of the exponential decay function.

$$w(D_j) = \exp^{-\alpha d_j^{\beta}} \tag{10}$$

A modification to Shepard's weight function consists of using a different value of α for each input pattern. Firstly, the L distances of $\mathbf{x}$ to its nearest neighbor in each individual classifier have to be sorted in decreasing order. Then, the value of α for each input pattern is computed according to $\alpha = L - j + 1$. By this, the higher the distance given by a classifier, the higher the value of α and thereby, the lower the weight assigned to such a classifier.

Finally, we propose another weighting function, which corresponds to the *average distance weight*. In summary, the aim is to reward (by assigning the highest weight) the individual classifier with the nearest neighbor to the input pattern. The rationale behind this function is that the classifier with the nearest neighbor to $\mathbf{x}$ will probably correspond to that with the highest accuracy in its classification.

$$w(D_j) = \frac{\sum_{i=1}^{L} d_i}{d_j} \tag{11}$$

4 Experimental Results

In this section, we present the results corresponding to the experiments carried out over seven data sets taken from the UCI Machine Learning Database Repository (`http://www.ics.uci.edu/~mlearn`). We adopted a 5-fold cross-validation process: each data set was divided into five equal parts, using four folds as the training set and the remaining block as an independent test set.

All classifier ensembles consist of nine individual classifiers ($L = 9$). The ensembles have been constructed through a class-dependent (stratified) resampling method by using four different techniques: selection without replacement (SWR), Bagging, Boosting, and Arc-x4. The unique classifier used for training all subsets corresponds to a 1-NN decision rule. Table 1 reports the averaged diversity computed over the different ensembles of classifiers, thus making possible to determine which resampling method produces the highest diversity, according to the measures introduced in Sect. 2.

It has to be noted that small values of the Q-statistics and the correlation coefficient indicate high diversity. Conversely, high values of the disagreement and the variability measures point to high diversity. This has been represented in Table 1 by including in brackets the relative position of each resampling method in a ranking of diversity (1 – highest; 4 – lowest). From this, one can see that while Arc-x4 clearly produces the highest diversity (except in the case of using the variability measure), the other resampling strategies give very similar levels of diversity (despite Boosting could be viewed as the method with the second highest diversity).

Theoretically, from these results, it is expected that the highest overall accuracies will be achieved when using Arc-x4, followed by Boosting. This will be checked in the next section, where we analyze the possible relation between diversity and accuracy, with respect to the resampling technique and/or the fusion scheme applied.

Table 1. Diversity by using different resampling techniques

	Heart	Pima	Vehicle	German	Phoneme	Waveform	Liver
Q-statistics							
SWR	0,27 (2)	0,45 (4)	0,48 (3)	0,40 (3)	0,68 (3)	0,49 (1)	0,17 (4)
Bagging	0,32 (3)	0,41 (3)	0,47 (2)	0,46 (4)	0,70 (4)	0,50 (2)	0,16 (3)
Boosting	0,39 (4)	0,39 (2)	0,48 (3)	0,38 (2)	0,67 (2)	0,57 (4)	0,12(2)
Arc-x4	0,13 (1)	0,22 (1)	0,41 (1)	0,34 (1)	0,61 (1)	0,54 (3)	0,04 (1)
Correlation coefficient							
SWR	0,16 (2)	0,26 (4)	0,30 (3)	0,23 (3)	0,44 (4)	0,29 (3)	0,13 (3)
Bagging	0,20 (3)	0,23 (3)	0,30 (3)	0,23 (3)	0,43 (3)	0,29 (3)	0,12 (2)
Boosting	0,22 (4)	0,22 (2)	0,26 (2)	0,19 (2)	0,36 (2)	0,28 (2)	0,55 (4)
Arc-x4	0,07 (1)	0,10 (1)	0,22 (1)	0,17 (1)	0,32 (1)	0,26 (1)	0,02(1)
Disagreement measure (%)							
SWR	0,41 (2)	0,33 (4)	0,34 (4)	0,36 (4)	0,22 (3)	0,26 (2)	0,42 (4)
Bagging	0,39 (3)	0,34 (3)	0,35 (3)	0,36 (3)	0,22 (3)	0,25 (3)	0,43 (3)
Boosting	0,38 (4)	0,36 (2)	0,37 (2)	0,38 (2)	0,26 (2)	0,28 (1)	0,47 (2)
Arc-x4	0,46 (1)	0,42 (1)	0,39 (1)	0,39 (1)	0,28 (1)	0,28 (1)	0,48 (1)
Variability measure							
SWR	0,92 (1)	0,80 (3)	0,84 (2)	0,88 (2)	0,55 (3)	0,62 (2)	0,95 (3)
Bagging	0,87 (2)	0,81 (2)	0,85 (1)	0,86 (3)	0,55 (3)	0,62 (2)	0,97 (1)
Boosting	0,82 (3)	0,84 (1)	0,83 (3)	0,89 (1)	0,61 (1)	0,67 (1)	0,96 (2)
Arc-x4	0,80 (4)	0,71 (4)	0,58 (4)	0,62 (4)	0,59 (2)	0,55 (4)	0,97 (1)

4.1 On the Relation Between Accuracy and Diversity

Importance of considering the possible relationship between accuracy and diversity of different resampling and/or fusion strategies comes from the fact that by this, it would be feasible to establish an appropriate policy to select the most suitable method for constructing classifier ensembles. The experimental results in Table 2 correspond to the average accuracy (and standard deviations) over the five folds when using the different resampling strategies (SWR, Bagging, Boosting, Arc-x4) together with the simple majority voting and the dynamic weighting methods described in Sect. 3.

From results in Table 2, we can sketch some comments. First, all ensembles provide similar performances, showing a slight improvement over the average accuracy of the single classifier. Second, application of some weighting function outperforms the simple majority voting on 6 out of 7 databases. Comparing the different weighting strategies, the best results correspond to the average distance and the inverse distance. Third, when focusing on the resampling strategies, although Boosting seems to be the method with the highest performance, in general differences are not statistical significant. Taking into account these preliminary results, it is possible to conclude that the fusion technique has a more important influence on accuracy than the resampling scheme.

When relating the diversity levels given in Table 1 with the accuracy rates reported in Table 2, one can observe that in most cases the highest diversity does not produce the highest performance, thus not fulfilling the theoretical expectations. It is worth pointing out that the variability measure is the one reflecting better the behavior of the ensembles, in the sense that those methods with the smallest values correspond to the lowest

Table 2. Average accuracies (and standard deviations) with different resampling and fusion methods. Values in bold type denote the highest accuracy for each database.

	Heart	Pima	Vehicle	German	Phoneme	Waveform	Liver
Single	58,2(6.2)	65,9(5.2)	64,2(1.8)	65,2(2.6)	76,1(8.4)	78,0(2.9)	65.2(4.8)
Simple voting							
SWR	62.2(2.1)	**72.8(5.0)**	61.4(1.9)	68.8(3.4)	75.0(10.0)	82.7(1.8)	63.8(7.2)
Bagging	62.6(5.0)	72.7(1.2)	60.6(2.3)	70.2(3.0)	75.0(9.4)	83.2(1.4)	63.2(5.2)
Boosting	63.0(5.5)	71.0(2.6)	62.3(4.7)	68.5(2.1)	71.9(13.7)	80.0(1.9)	65.2(4.7)
Arc-x4	59.3(3.9)	69.7(2.9)	54.8(4.3)	68.7(2.5)	74.4(11.1)	78.8(2.3)	63.8(6.9)
Average distance weight							
SWR	62.2(4.8)	72.0(4.5)	63.1(3.0)	69.3(3.3)	75.4(9.6)	82.7(1.7)	65.2(7.7)
Bagging	62.6(5.1)	72.7(1.7)	60.7(2.4)	**70.8(3.1)**	75.2(9.2)	83.2(1.4)	64.9(7.2)
Boosting	**63.4(3.5)**	69.7(3.2)	63.9(3.9)	68.8(2.2)	72.9(11.8)	80.0(1.9)	62.9(5.1)
Arc-x4	60.0(5.9)	69.5(3.1)	58.4(2.7)	67.6(2.8)	74.5(11.4)	79.5(1.9)	**66.1(4.8)**
Inverse distance weight							
SWR	62.2(4.8)	72.0(4.5)	63.1(3.0)	69.3(3.3)	75.4(9.6)	**83.5(0.8)**	64.6(7.9)
Bagging	62.6(5.1)	72.7(1.7)	60.7(2.4)	**70.8(3.1)**	75.2(9.2)	83.2(1.4)	64.4(6.7)
Boosting	**63.4(3.5)**	69.7(3.2)	63.9(3.9)	68.8(2.2)	72.9(11.8)	80.0(1.9)	62.6(5.0)
Arc-x4	60.0(5.9)	69.5(3.1)	58.4(2.7)	67.6(2.8)	74.5(11.4)	79.5(1.9)	64.9(5.0)
Shepard's weight							
SWR	58.2(3.4)	66.9(5.6)	63.7(2.2)	67.3(2.1)	75.0(9.9)	82.1(1.9)	65.8(4.1)
Bagging	59.6(10.2)	66.1(4.7)	61.5(2.9)	67.1(2.7)	74.8(9.4)	83.1(1.7)	62.6(5.6)
Boosting	61.1(4.0)	65.0(4.7)	65.6(1.5)	68.1(2.8)	72.9(12.2)	79.5(2.3)	57.4(8.6)
Arc-x4	58.5(8.1)	65.6(5.5)	59.2(3.4)	65.7(1.3)	74.4(11.0)	79.3(1.7)	63.8(3.4)
Modified Shepard's weight							
SWR	58.9(3.8)	66.8(5.4)	63.2(2.5)	64.3(1.9)	**75.7(10.0)**	77.7(2.5)	64.4(4.2)
Bagging	59.6(9.3)	66.0(4.8)	61.1(2.8)	64.4(2.7)	75.5(8.7)	78.1(2.1)	61.5(5.0)
Boosting	60.7(3.2)	65.8(5.0)	**65.7(0.8)**	66.1(1.7)	73.2(10.9)	76.3(3.1)	56.8(5.6)
Arc-x4	58.9(7.9)	65.1(5.4)	59.2(3.4)	65.8(1.1)	73.9(11.2)	76.2(1.6)	62.0(4.3)

accuracies. For example, Arc-x4 appears as the scheme with the lowest variability and also with the lowest accuracy.

5 Concluding Remarks

The present paper has analyzed the relationship between four diversity measures and the overall accuracy obtained with an ensemble of nine individual classifiers constructed by means of different resampling methods and various weighting functions for classifier fusion.

From the experiments, it seems that in general, diversity has low influence on the overall accuracy. In this sense, we found that not always the best results correspond to those situations with a higher diversity; analogously, small values of diversity do not directly imply low accuracy. With regards to resampling, we found that Arc-x4 appears as the scheme that produces ensembles with the highest diversity levels, although other methods, such as randomization [6], should be tested in a further research.

Acknowledgments. Partially supported by grants 10007-2006-01 (51626) and SEP-2003-C02-44225 from Mexican CONACyT, and DPI2006-15542 from Spanish CICYT.

References

1. Bahler, D., Navarro, L.: Methods for combining heterogeneous sets of classifier. In: Proc. 17th Natl. Conf. on Artificial Intelligence, Workshop on New Research Problems for Machine Learning (2000)
2. Banfield, B.E., Hall, L.O., Bowyer, K.W., Kegelmeyer, Jr., W.P.: A new Ensemble diversity measure applied to thinning ensembles. In: Proc. 4th Intl. Workshop on Multiple Classifier Systems, Guildford, UK, pp. 306–316 (2003)
3. Barandela, R., Valdovinos, R.M., Sánchez, J.S.: New applications of ensembles of classifiers. Pattern Analysis and Applications 6, 245–256 (2003)
4. Breiman, L.: Bagging predictors: Machine Learning. vol. 26, pp. 123–140 (1996)
5. Breiman, L.: Arcing classifiers: Annals of Statistics. vol. 26, pp. 801–823 (1998)
6. Dietterich, G.T.: An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. Machine Learning 40, 139–157 (2000)
7. Dudani, S.A.: The distance weighted k-nearest neighbor rule. IEEE Trans. on Systems, Man and Cybernetics 6, 325–327 (1976)
8. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: Proc. 13th Intl. Conference on Machine Learning, pp. 148–156. Morgan Kaufmann, San Francisco (1996)
9. Ho, T.K.: Complexity of classification problems and comparative advantages of combined classifiers, In: Proc. 1st. Intl. Workshop on Multiple Classifier Systems, Cagliari, Italy, pp. 97–106 (2000)
10. Kuncheva, L.I.: Using measures of similarity and inclusion for multiple classifier fusion by decision templates. Fuzzy Sets and Systems 122, 401–407 (2001)
11. Kuncheva, L.I., Bezdek, J.C., Duin, R.P.W.: Decision templates for multiple classifier fusion. Pattern Recognition 34, 299–314 (2001)
12. Kuncheva, L.I., Kountchev, K.R.: Generating classifier outputs of fixed accuracy and diversity. Pattern Recognition Letters 23, 593–600 (2002)
13. Kuncheva, L.I., Whitaker, C.J.: Measures of diversity in classifier ensembles. Machine Learning 51, 181–207 (2003)
14. Littlestone, N., Warmuth, M.: Weighted majority algorithm. Information and Computation 108, 212–261 (1994)
15. Narasimhamurthy, A.: Evaluation of diversity measures for binary classifier ensembles. In: Proc. 6th Intl. Workshop on Multiple Classifier Systems, Seaside, CA, pp. 13–15 (2005)
16. Shepard, R.N.: Toward a universal law of generalization for psychological science. Science 237, 1317–1323 (1987)
17. Shipp, C.A., Kuncheva, L.I.: Relationships between combination methods and measures of diversity in combining classifier. Information Fusion 3, 135–148 (2002)
18. Wanas, N., Kamel, M.: Weighted combining of neural network ensembles. In: Proc. Intl. Joint Conf. on Neural Networks, vol. 2, pp. 1748–1752 (2002)