A Bound on Kappa-Error Diagrams for Analysis of Classifier Ensembles

Ludmila I. Kuncheva, *Member*, *IEEE*

Abstract—Kappa-error diagrams are used to gain insights about why an ensemble method is better than another on a given data set. A point on the diagram corresponds to a pair of classifiers. The x-axis is the pairwise diversity (kappa), and the y-axis is the averaged individual error. In this study, kappa is calculated from the 2×2 correct/wrong contingency matrix. We derive a lower bound on kappa which determines the feasible part of the kappa-error diagram. Simulations and experiments with real data show that there is unoccupied feasible space on the diagram corresponding to (hypothetical) better ensembles, and that individual accuracy is the leading factor in improving the ensemble accuracy.

Index Terms—Classifier ensembles, kappa-error diagrams, ensemble diversity, limits

1 Introduction

Classifier ensembles (multiple classifier systems) have now shown their potential for solving challenging real-life problems, an example of which is predicting users' film preferences in the high-profile Netflix competition.[1] Multiple classifier systems have been studied extensively in the past 10-15 years [24], and a wide collection of approaches, methods, and algorithms has been created [6], [7], [13], [19], [26], [23].

The success of classifier ensembles is often attributed to the concept of diversity. Many attempts have been made to define, explain, and measure diversity in classifier ensembles, and to relate it with the ensemble accuracy [5], [14], [16], [25], and yet diversity is often listed among the open questions about ensemble classification [21]. The relationship between certain measures of diversity and the classification margin theory has been demonstrated, explaining why large diversity maybe preferable [25]. The common wisdom now is that while diversity is an important factor for the ensemble accuracy, their interdependence is not straightforward. This discourages constructing ensemble methods by handling diversity explicitly [25]. Regardless of the theoretical doubts and the limited success thus far, explicit handling of diversity seems to be still an appealing research perspective [1], [11], [18].

A popular tool for analyzing ensemble methods is the so-called kappa-error diagrams proposed by Margineantu and Dietterich [17]. Kappa error diagrams visualize individual accuracy and diversity in a 2D plot, and have been used to decide which ensemble members can be pruned without much harm to the overall performance [17], [22]. An ensemble of L classifiers is shown on the diagram as a scatterplot (a "cloud") of $L(L-1)/2$ points, each corresponding to a pair of classifiers. The x-coordinate is a diversity measure of the pair, κ, which is also known as the interrater agreement [8]. Smaller values of kappa indicate high diversity, $\kappa = 0$ indicates independent classifiers, and $\kappa = 1$, identical classifiers. The y-coordinate is the averaged individual error rate of the classifier pair. Thus, points that are closer to the bottom left corner of the diagram are preferable (high diversity and low error).

Note that the exact left bottom corner at $(-1, 0)$ is not achievable. Classifiers that are ideally accurate ($e = 0$) will be identical; therefore, $\kappa = 1$. For each ensemble, a compromise between diversity and individual accuracy must be negotiated. The clouds corresponding to different ensembles, plotted on the same diagram, usually form a "belly" whereby ensembles with higher diversity have higher errors and vice versa.

It is curious to find out why this belly-shaped pattern exists, and how close a pair of classifiers can be to the bottom left corner of the diagram. The objective of this paper is to derive and illustrate with examples a *tight lower bound* on the kappa-error diagram, which delineates its feasible region.

The rest of the paper is organized as follows. Section 2 gives the derivation of the bound, Section 3 contains a simulation study, and Section 4 contains an illustration with 31 real data sets and five classifier ensemble methods.

2 Derivation of the Bound

The interrater agreement measure κ can be calculated from two different perspectives. First, disregarding the issue of correct or wrong classification, κ can be used to measure to what extent the classifiers agree in assigning the class labels. Second, kappa can be used to measure to what extent the classifiers agree in assigning the correct label, regardless of the class assignment. Diagrams with both choices of kappa exhibit the belly-shaped pattern. In this study, we chose the second perspective because it lends itself to the algebraic

1. http://www.netflixprize.com/.

• *The author is with the School of Computer Science, Bangor University, Dean Street, Bangor, Gwynedd LL57 1UT, United Kingdom. E-mail: l.i.kuncheva@bangor.ac.uk.*

Manuscript received 11 May 2011; revised 5 Oct. 2011; accepted 7 Oct. 2011; published online 10 Nov. 2011.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2011-05-0260. Digital Object Identifier no. 10.1109/TKDE.2011.234.

manipulations necessary to derive the bound, and is equally useful for analyzing classifier ensembles. The ensemble analyzes and comparisons are usually carried out offline, when training and testing have been completed. Typically, no decision about the training of a particular ensemble is made through the kappa-error analysis. Then, the individual errors of the classifiers and the ensembles (*on the testing data*) are readily available, and can be fed into the kappa-error calculations.

Consider N data points and the contingency table of two classifiers, C_1 and C_2

	C_2 correct	C_2 wrong
C_1 correct	a	b
C_1 wrong	c	d

where the table entries are the number of points jointly classified as indicated, and $a+b+c+d=N$.

The averaged individual error for the pair of classifiers is

$$e = \frac{1}{2}\left(\frac{c+d}{N} + \frac{b+d}{N}\right) = \frac{b+c+2d}{2N}. \tag{1}$$

Diversity between the two classifiers is measured by κ [8], which is constructed as

$$\kappa = \frac{\mathrm{OA} - \mathrm{AC}}{1 - \mathrm{AC}}, \tag{2}$$

where OA is the observed agreement, i.e., the probability that the two classifiers will be both correct or both incorrect when classifying a randomly chosen data point. AC is the agreement by chance, i.e., the probability that the two classifiers will agree by chance on a randomly chosen data point. The two quantities are calculated as

$$\mathrm{OA} = \frac{a+d}{N} \tag{3}$$

$$\mathrm{AC} = \frac{(a+b)(a+c)+(b+d)(c+d)}{N^2}. \tag{4}$$

Substituting in (2) and rearranging the terms, we obtain

$$\kappa = \frac{2(ad-bc)}{(a+b)(b+d)+(a+c)(c+d)}. \tag{5}$$

To facilitate further analyses, it will be convenient to express κ in terms of e and N. We can express a and d as functions of b, c, e, and N and substitute in (5), which leads to

$$\kappa = 1 - \frac{2N(b+c)}{4N^2 e(1-e) + (b-c)^2}. \tag{6}$$

The only restrictions on the values of a, b, c, and d so far are that each is nonnegative and they sum up to N. For a fixed e and $(b+c)$, if $b \neq c$, there will be a positive term $(b-c)^2$ in the denominator, which will decrease the fraction, and therefore increase κ. By requiring that $b=c$, and hence dropping the respective term from the denominator, a smaller κ is obtained

$$\kappa' = 1 - \frac{2b}{2Ne(1-e)} = 1 - \frac{b}{Ne(1-e)} \leq \kappa. \tag{7}$$

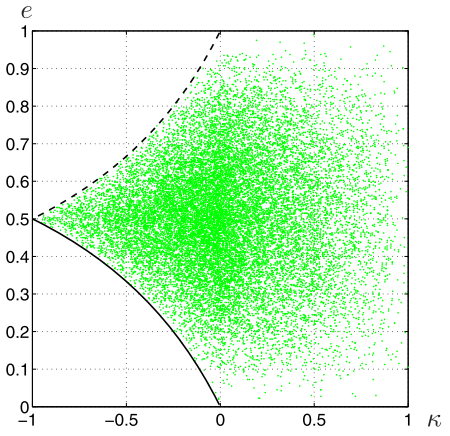


Fig. 1. Kappa-error diagram of 20,000 simulated classifier pairs and the bound.

The minimum value of kappa will be obtained for the largest possible b for the fixed e. To find this value, consider the following system of equations and inequalities:

$$e = \frac{b+c+2d}{2N} = \frac{b+d}{N} \quad \text{error} \tag{8}$$

$$2b + d \leq N \quad \text{total count} \tag{9}$$

$$d \geq 0 \quad \text{nonnegativity}. \tag{10}$$

Expressing d from (8), $d = Ne - b$, and substituting in (9), we obtain

$$b \leq N(1-e).$$

On the other hand, substituting in (10),

$$b \leq Ne.$$

Since both must be satisfied,

$$b_{\max} = \min\{N(1-e), Ne\}.$$

If $e \leq 0.5$, $b_{\max} = Ne$ and for $e > 0.5$, $b_{\max} = N(1-e)$. Then, the minimum κ is given by

$$\kappa_{\min} = \begin{cases} 1 - \dfrac{1}{1-e}, & \text{if } 0 < e \leq 0.5 \\ 1 - \dfrac{1}{e}, & \text{if } 0.5 < e < 1. \end{cases} \tag{11}$$

Note that the bound is tight. It is achievable for $b=c$ and $d = \max\{0, (e-0.5)N\}$. The bound is plotted in Fig. 1. The upper branch, plotted with a dashed line, is of less interest because it corresponds to individual error for the pair of classifiers $e > 0.5$. The lower branch is the "target" part of the bound, where better ensembles are expected to be found.

The bound itself is not directly related to the ensemble performance. It is expected that ensembles that have classifier pairs closer to the bound will fare better than ensembles that are far away. The bound helps by giving additional insight about the extent of theoretically possible improvement of the ensemble members. It does not however prescribe the way of creating these classifiers.

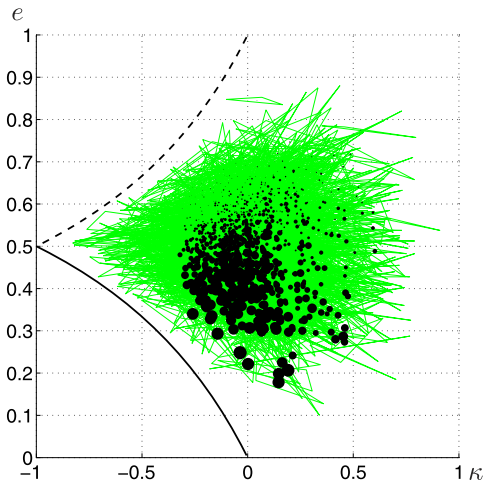


Fig. 2. Kappa-error diagram of 1,000 simulated ensembles of three classifiers. The size of the dots represent the majority vote accuracy of the ensemble—the larger the dot, the higher the accuracy.

3 A Simulation Study

Fig. 1 shows the results from 20,000 simulated classifier pairs. The number of data points was fixed at $N = 200$ for each contingency table. The N points were randomly split to fill in the a, b, c, and d values in the contingency table. Each classifier pair is a point on the plot, where the coordinates κ and e are calculated as in (1) and (5). The bound (11) is drawn with a solid black line for $e \leq 0.5$ and with a dashed line for $e > 0.5$.

Next, we generated randomly 1,000 ensembles of $L = 3$ classifiers. Each ensemble was a three-way contingency table with eight entries: $N_{000}, N_{001}, \ldots, N_{111}$. The value N_{xxx} is the number of data points that have been classified correctly ($x = 1$) or wrongly ($x = 0$) by classifiers 1, 2, and 3, respectively. For example, N_{011} is the number of points classified correctly by classifiers 2 and 3 and misclassified by classifier 1. The integers N_{xxx} were generated randomly so that $\sum_{xxx} N_{xxx} = N$. The majority vote accuracy can be calculated from the three-way contingency table as

$$P_{\text{maj}} = \frac{1}{N}(N_{110} + N_{101} + N_{011} + N_{111}). \tag{12}$$

Fig. 2 illustrates the random ensembles. Each ensemble is depicted as a triangle where the three classifier pairs in the ensemble (points) are linked with green lines. In the geometric centre of each triangle, a black dot is plotted to indicate the centre of the ensemble "cloud." The size of the dot is a gauge of the ensemble accuracy. Ensembles with higher majority vote accuracy are shown with larger dots. A tendency can be observed: ensembles that have more accurate individual classifiers (the triangle is lower down on the y-axis) are better. This tendency is mirrored in the experiments with real data and with ensembles of size $L = 1{,}000$, shown later. Interestingly, diversity does not play such big a role as might be expected. The size of the centre points increases slightly to the left (toward smaller κ, hence large diversity) but the error-related tendency is much more pronounced. This suggests that in order to create small ensembles with high majority vote accuracy, we should strive to obtain accurate individual classifiers and be less concerned about their diversity. We note that, while the bound on the diagram is valid for any ensemble method, Fig. 2 gives insights only about the majority vote of ensembles of three classifiers.

Shown in Fig. 3 is the surface of the ensemble accuracy over the kappa-error diagram. The contours of the surface are projected on the diagram. The data for this plot were approximated from the 1,000 random ensembles used to produce Fig. 2. The peak toward the lower branch of the bound suggests that low individual error and high diversity, *in combination*, lead to better ensembles. It is also

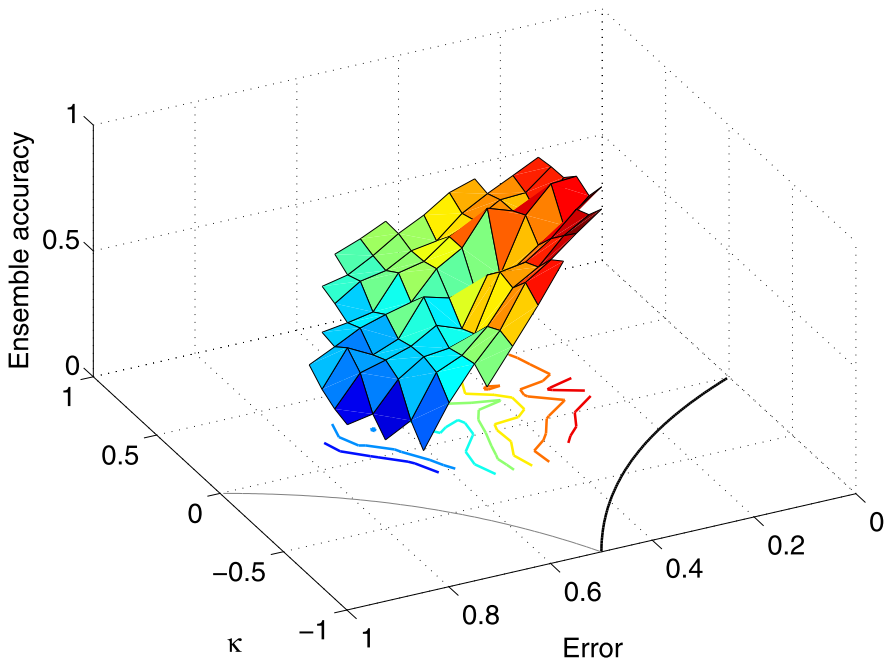


Fig. 3. Surface of the ensemble accuracy of the simulated classifier ensembles.

TABLE 1
Accuracy (10-Fold Cross Validation) [in Percent] for the Five Ensemble Methods and the Real Data Sets

	Decision Trees, $L = 11$					Linear Classifiers, $L = 11$					Linear Classifiers, $L = 1000$				
Dataset	BA	AD	RS	RF	RO	BA	AD	RS	RF	RO	BA	AD	RS	RF	RO
breast	72.5	71.0	72.6	71.1	72.1	74.0	72.2	73.6	73.2	73.6	73.6	72.5	72.9	73.6	72.1
conetorus	86.5	85.4	72.4	85.4	87.0	74.7	75.0	65.7	72.1	79.2	74.9	74.9	68.5	72.6	79.8
contractions	85.7	84.7	84.7	87.7	82.7	84.9	78.8	84.8	84.9	83.9	82.9	83.9	81.8	81.9	85.9
crabs	86.5	92.5	85.0	99.0	86.0	100.0	100.0	97.0	100.0	100.0	100.0	98.0	100.0	100.0	100.0
4_gauss	98.0	97.0	91.0	99.0	98.0	97.0	97.0	86.0	98.0	98.0	98.0	89.0	97.0	98.0	98.0
german	75.6	73.3	73.9	77.0	74.4	77.7	76.7	74.1	76.9	76.4	76.8	77.5	74.8	77.3	77.4
glass	70.2	65.8	72.5	66.4	72.9	59.8	58.9	55.8	61.2	64.5	59.8	58.9	59.4	59.8	63.1
heart	78.7	78.1	80.5	79.4	78.1	65.3	69.4	80.1	63.6	67.7	63.3	70.1	79.5	63.6	68.7
image	96.6	98.2	97.0	98.1	97.4	72.0	77.0	75.2	72.2	76.5	72.1	80.7	79.9	72.4	78.3
intubation	86.4	86.8	87.1	87.8	85.4	83.1	82.1	80.8	65.3	84.1	82.5	84.8	81.8	63.9	84.1
ionosphere	89.2	91.5	92.3	94.3	90.0	80.6	86.0	83.5	79.5	84.9	79.2	87.8	84.0	78.9	86.3
iris	95.3	94.7	94.0	98.7	96.0	98.0	97.3	96.7	92.0	98.0	98.0	96.0	96.0	92.0	98.0
laryngeal1	85.1	81.3	82.6	83.1	82.2	83.1	85.4	84.0	63.9	83.1	85.0	85.0	83.5	63.4	83.6
laryngeal2	95.8	96.5	95.5	96.4	96.0	95.2	95.9	95.5	69.5	95.5	95.7	96.5	95.5	69.5	95.4
laryngeal3	69.7	70.3	69.7	69.7	70.3	70.5	75.9	73.7	53.6	72.8	71.4	72.5	71.7	53.6	71.9
liver	68.7	65.8	65.8	69.5	69.3	69.2	68.1	60.8	66.3	72.2	67.8	68.6	59.7	67.8	73.0
lymph	41.0	42.3	43.6	41.5	41.5	37.3	37.8	42.9	39.8	37.3	38.0	41.1	41.1	38.0	39.8
pima	75.4	69.8	73.6	75.4	77.2	76.6	76.7	73.7	63.0	76.0	76.8	77.3	75.9	63.4	76.0
rds	88.3	90.6	88.5	88.3	91.9	91.8	89.6	94.0	70.4	94.2	92.9	81.0	94.0	70.4	92.8
sonar	79.3	79.8	80.3	81.8	78.4	70.2	74.1	76.0	73.5	75.5	74.5	73.6	78.0	73.6	78.9
spect_bin	82.4	78.6	81.3	82.8	81.3	82.4	81.3	82.4	83.5	81.3	82.4	83.2	83.5	84.3	80.5
spect_cont	87.1	91.7	89.7	93.1	89.4	77.7	85.4	77.4	67.9	80.2	79.7	89.4	78.8	67.4	83.7
spirals	57.7	64.9	53.6	58.2	57.7	47.9	49.5	47.9	49.6	50.0	46.9	48.4	50.6	47.0	48.5
thyroid	93.5	95.8	94.9	95.8	94.8	91.2	96.3	89.8	87.0	92.6	91.2	96.3	91.2	86.6	94.0
vehicle	74.8	77.0	73.4	78.1	75.2	76.9	76.8	76.9	77.4	82.6	77.6	78.5	74.7	77.5	83.3
voice_3	77.4	74.5	78.2	77.8	79.5	76.4	76.9	78.6	58.4	76.0	73.9	76.9	76.5	60.9	76.1
votes	96.1	96.6	93.9	96.1	95.2	97.0	94.3	94.8	89.7	97.0	97.0	93.1	96.1	90.1	97.0
vowel	86.8	92.8	87.0	90.8	88.2	64.0	63.9	58.3	58.4	77.8	62.8	63.3	63.7	60.9	80.0
wbc	94.7	96.5	95.3	96.3	94.9	95.4	95.2	96.3	95.3	96.5	95.6	95.2	95.9	95.6	96.7
wine	94.9	98.3	96.1	97.2	96.6	98.3	97.1	95.4	73.1	96.0	98.3	96.5	98.2	73.1	97.1
zoo	71.8	72.7	73.6	71.8	70.1	67.5	69.3	74.5	65.8	67.4	66.7	66.7	73.6	66.7	70.1

clear from the steep descent of the surface toward the higher branch of the bound, that high diversity is harmful if the individual error is high.

4 Illustration with Real Data

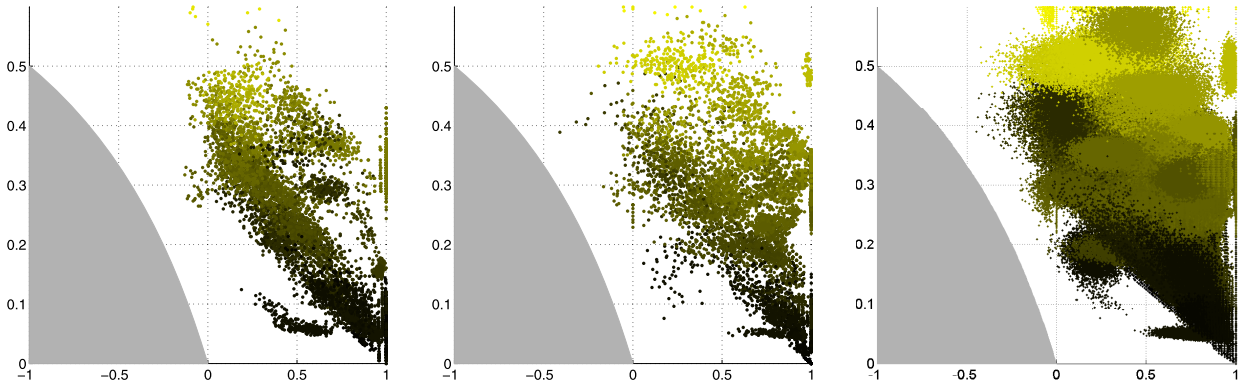
We used 31 data sets from the UCI repository [9] and a private collection[2] as indicated in Table 1. Notice that the purpose of the experiments was not to compare classifier ensemble methods across data sets but to illustrate how the new bound can be used as a reference point in such analyses. More specifically, we seek to answer the following questions:

- Question 1. Which part of the kappa-error diagram is occupied by the clouds representing the classical ensemble methods (Bagging (BA), AdaBoost (AD), and Random Subspace (RS)) and some more recent methods (Rotation Forest (RF) and Random Oracle (RO))? How close are these clouds to the bound proposed here?
- Question 2. What leverage do we have to move the ensemble clouds on the diagram toward more desirable feasible spaces? In other words, what effect do ensemble parameters have on the positioning of the cloud of points? For example, does a larger ensemble size expand the cloud in the direction of the bound? Does a change of the base classifier model affect the position of the cloud?

The methods for creating classifier ensembles use different heuristics and theories to achieve simultaneous individual accuracy and diversity. The following classifier ensemble methods were considered here:

1. *Bagging* [3]. L bootstrap samples of the data are taken and a classifier is trained on each sample. The joint decision is made by aggregating the individual classifier votes. The most popular aggregation method is the majority vote. Bagging has been found to be robust to noise, and is known to reduce the variance of the individual classifiers [2], making it a necessary benchmark contestant in any classifier ensemble experimental study.
2. *AdaBoost* [10]. The ensemble is constructed sequentially. The new classifier is trained on a data set that is sampled from the initial data set, so that the instances that were difficult for the previous ensemble members get more exposure. The decision is made by weighted majority voting. AdaBoost has

2. Available at http://pages.bangor.ac.uk/~mas00a/activities/artificial_data.htm and http://pages.bangor.ac.uk/~mas00a/activities/real_data.htm.



(a) $L = 11$, decision trees (b) $L = 11$, linear (c) $L = 1000$, linear

Fig. 4. Kappa-error diagram of the five ensemble methods on the 31 real data sets, $L = 1{,}000$, linear base classifiers.

been declared in the past to be the "best off-the-shelf classifier" [4]. A host of advanced AdaBoost variants have been proposed including LogitBoost, AveBoost, Gentle AdaBoost, Modest AdaBoost, and more. AdaBoost is sensitive to noise but increases the ensemble diversity in a clever implicit way. It was found to reduce both variance and bias of the error of the individual classifiers. AdaBoost sacrifices individual accuracy for diversity; hence, the respective cloud on the kappa-error diagram is usually oblong, tilted to the left, and situated slightly North-West from the cloud of the Bagging ensemble for the same data.

3. *Random Subspace* [12]. L subsets of features are sampled randomly from the feature set, and a classifier is trained on each subspace. The results depend heavily on the data sets; most often this method increases diversity, but not as much as AdaBoost.
4. *Rotation Forest* [20]. By design, this ensemble method is aimed at increasing diversity between the classifiers without, if possible, sacrificing individual accuracy. Bagging and AdaBoost take samples from the data; therefore, none of the individual classifiers gets to "see" the whole training data sets. Random Subspace, on the other hand, may use all of the instances to train each individual classifier but sheds features to create diversity. Rotation Forest uses decision tree classifiers, as suggested by its name. Each classifier is trained on the whole data set using *extracted* features. These are formed through a combination of partial rotations of the feature space through Principal Component Analysis (PCA). Thus, the cloud of the Rotation Forest ensemble is expected to appear left to that of Bagging and underneath these of AdaBoost and Random Subspace. The diversity introduced through RF is not as high as that due to AdaBoost or Random Subspace but, as argued in Section 3, diversity seems to be less important than the individual accuracy.
5. *Random Oracle* [15]. Each member of the ensemble is a miniensemble itself. To build an ensemble member, a random oracle is constructed, for example by splitting the space with a random hyperplane (Random Linear Oracle). The data are split according to the oracle, and a separate classifier is trained for each half-space. When a new instance comes for classification, the ensemble member applies the oracle, and the classifier responsible for the instance offers a class label, taken to be the label proposed by the ensemble member. This method will work if the data set is large enough to ensure that the classifiers in the two half-spaces are adequately trained.

Table 1 shows the classification accuracy with the five classifier ensemble methods and the 31 data sets for ensemble sizes $L = 11$ and $L = 1{,}000$. The accuracy is calculated through a 10-fold cross validation, where the folds were kept the same for all ensemble methods. The left subtable shows the accuracies with the decision tree classifier as implemented within the Statistics Toolbox of Matlab, while the middle and the right subtables show the ensemble accuracies with a linear base classifier. No parameters were optimized within the experiment. All code was written in Matlab. The highest accuracy for each data set and each subtable is underlined.

Even though the ensemble accuracy depends primarily on the data set, with the reasonably sized collection used here, it is interesting to look for a general pattern. Fig. 4 shows the kappa-error plots for the 31 data sets and the five ensemble methods. The plots of all ensemble methods and data sets for a specific L and base classifier are overlaid. The ensemble accuracy is indicated by color. Lighter color signifies lower accuracy.

Indeed, the plots demonstrate several general tendencies:

- Ensemble accuracy is higher (darker color) for clouds closer to the bound.
- The darker color toward the bottom right corner confirms the result observed in the simulations: the individual accuracy is the dominant factor for better ensemble accuracy.
- There is feasible unoccupied space in the diagram, where ensembles of higher accuracy may be engineered.
- Increasing the ensemble size leads to more dense clouds, creating outliers which lie closer to the bound, paving the ground for ensemble pruning methods.
- Interestingly, for the same ensemble construction methods and the same ensemble size $L = 11$, the

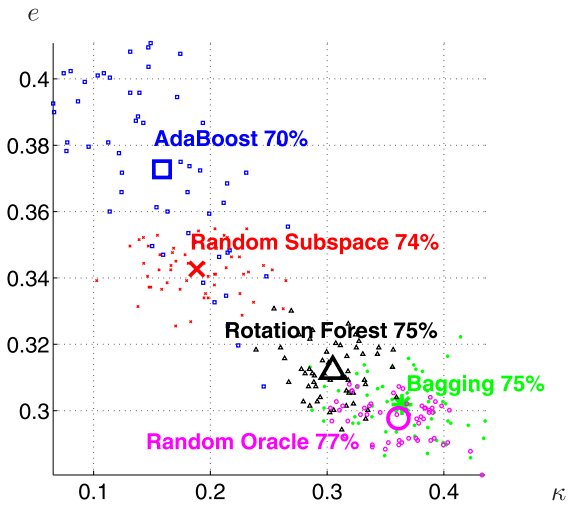


Fig. 5. Kappa-error diagrams for the pima data with ensemble size $L = 11$ and *tree* classifiers.

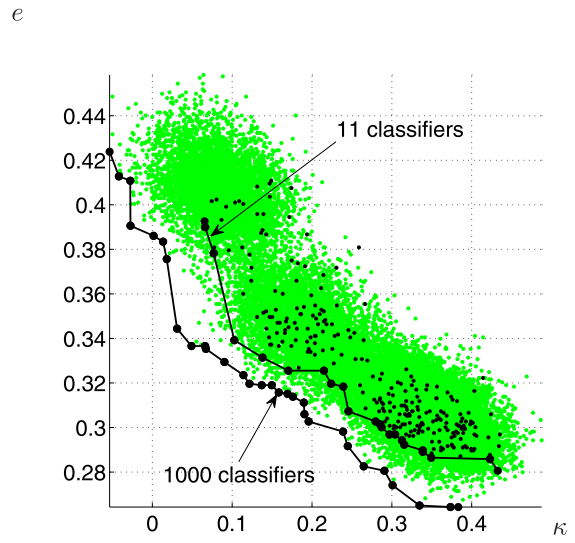


Fig. 7. Position of the two pima clouds and their front edges.

linear classifiers outreach the decision tree classifiers, having more "accidental" classifier pairs closer to the bound (compare subplots (a) and (b)). However, they also generate more clouds toward the right edge of the diagram ($\kappa = 1$) showing low diversity.

The following examples illustrate further some of the observed points. Fig. 5 is a "zoom" of the ensemble clouds for the pima data. The individual ensembles are shown with different marker and color.

The relative positioning of the clouds matches the expectation to a large extent, placing Bagging as the one of the most accurate and least diverse methods and Adaboost as the most diverse and least accurate one. As observed in the simulations, when relatively small ensemble sizes are involved, the accuracy of the individual classifiers is the dominant factor for the success of the ensemble. The belly shape of the overall cloud is clearly visible. Fig. 6 relates this shape to the proposed bound. We join the points that form a Pareto-optimal set to mark the front edge of the cloud. These points correspond to the *nondominated pairs* of classifiers. A pair is called nondominated if there is no other pair that has

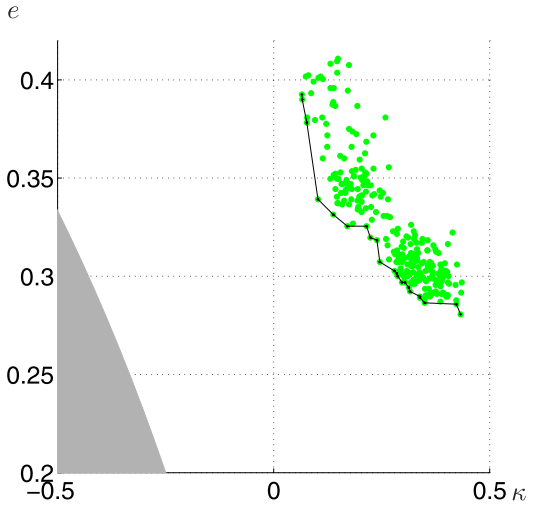


Fig. 6. Position of the pima ($L = 11$) cloud in relation to the bound.

both lower error and higher diversity. The large gap between the bound and the cloud edge suggests that, in theory, there could be classifier ensemble pairs with better diversity-error values, hopefully leading to better overall ensembles. For a point to reside on the bound, the two classifiers must have exactly the same error and be highly dependent in that there should be no points misclassified simultaneously. It is unlikely that this case will appear in real data, and it is difficult to engineer such classifier pairs artificially.

For large ensembles, the clouds on the kappa-error diagram are expected to be denser and larger, but not shifted. Fig. 7 shows a comparison between the front edge of the cloud of points containing all ensemble methods for ensemble sizes $L = 11$ and $L = 1{,}000$. The points for $L = 11$ are inset with black dots within the larger cloud for $L = 1{,}000$. Expectedly, the edge is shifted in the direction of the bound which prompts the idea of constructing an ensemble by selecting a subset of ensemble members; the same idea that underpinned the study of Margineantu and Dietterich [17] where the kappa-error diagrams were first introduced. It turned out that manual or heuristic selection of the ensemble members has not led to dramatically better ensembles. The kappa-error bound may give a new perspective on the evaluation of the merit of classifier pairs and open new possibilities for creating ensembles by selection.

To demonstrate in more detail the effect of using a different base classifier on the positioning of the ensemble clouds (part of Question 2), we chose pima data with $L = 11$ and the linear classifier. The results are plotted in Fig. 8, and the classification accuracies are indicated as in Fig. 5.

The front edge of the cloud has the same shape; however, the whole cloud is shifted a little down and to the right, indicating higher accuracy and lower diversity. Fig. 9 shows the two Pareto-optimal edges. Interestingly, the Rotation Forest is no longer a competitive ensemble method contributing to the Pareto-optimal set of classifier pairs. Its accuracy drops substantially, most likely because the linear classifier is not sensitive to the heuristics which ensure diversity for this ensemble method, while the decision tree classifier is.

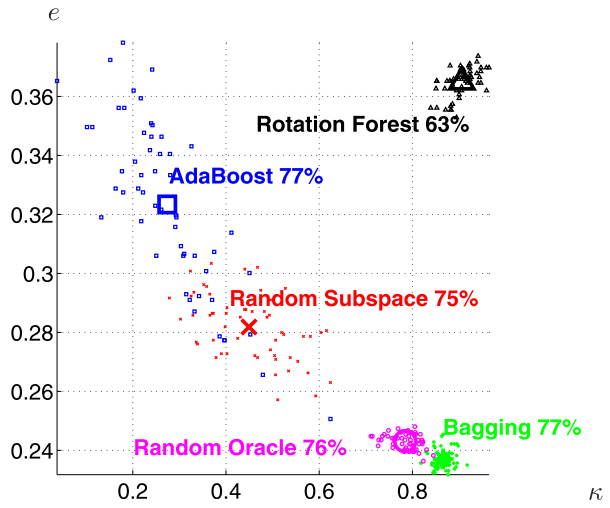


Fig. 8. Kappa-error diagrams for the pima data with ensemble size $L = 11$ and *linear* classifiers.

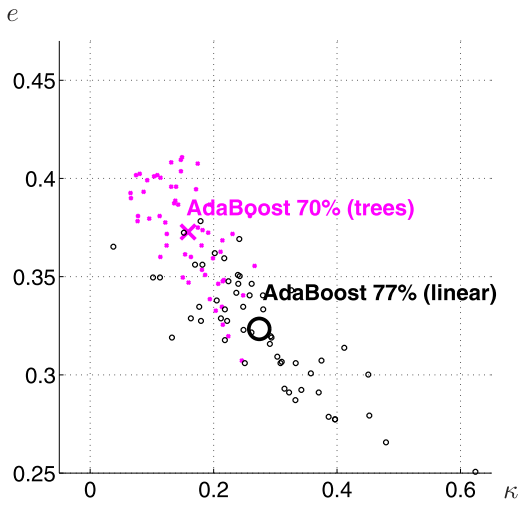


Fig. 10. Relative position of the Adaboost clouds for two base classifier models and the pima data $L = 11$.

The most noticeable accuracy gain resulting from switching the classifier models is achieved by Adaboost. To examine the cause for this, Fig. 10 shows the two clouds together. The results indicate that for small ensemble sizes, sacrificing diversity for individual accuracy is justifiable.

To summarize, the ensemble clouds usually lie far from the feasible lower bound. One way to use up the space is to increase the ensemble size, possibly followed by a classifier selection procedure. Changing the base classifier type can alter the position of the ensemble cloud.

5 Conclusions

Kappa-error diagrams have been used for getting insights in comparing classifier ensembles. In this study, we chose pairwise diversity kappa calculated from the 2×2 contingency matrix of correct/wrong classification. We derive a bound which determines the best achievable tradeoff between individual accuracy and diversity. The experimental results demonstrate that there is unoccupied *feasible* space on the diagram for new ensemble methods. We also found through simulations and real-data experiments that individual accuracy is the leading factor for the ensemble success compared to ensemble diversity.

It is interesting to investigate how the shape of the ensemble cloud is related to the ensemble accuracy. Such an analysis may spawn new ensemble creation methods which, like AdaBoost, construct the ensemble sequentially so that a certain cloud shape is achieved.

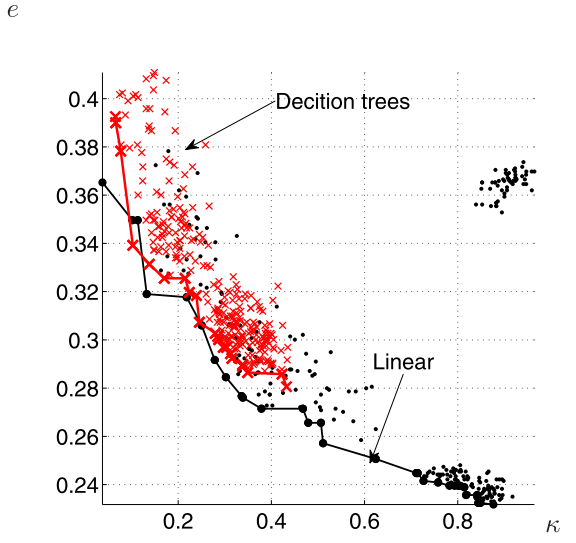


Fig. 9. Comparison of the cloud edges for pima data ($L = 11$) for decision tree and linear base classifiers.

References

[1] B.H.G. Barbosa, L.T. Bui, H. Abbass, L. Aguirre, and A.P. Braga, "The Use of Coevolution and the Artificial Immune System for Ensemble Learning," *Soft Computing*, vol. 15, no. 9, pp. 1735-1747, June 2011.

[2] E. Bauer and R. Kohavi, "An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants," *Machine Learning*, vol. 36, pp. 105-142, 1999.

[3] L. Breiman, "Bagging Predictors," *Machine Learning*, vol. 26, no. 2, pp. 123-140, 1996.

[4] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, pp. 5-32, 2001.

[5] G. Brown, J. Wyatt, R. Harris, and X. Yao, "Diversity Creation Methods: A Survey and Categorisation," *Information Fusion*, vol. 6, no. 1, pp. 5-20, 2005.

[6] G. Brown, "Ensemble Learning," *Encyclopedia of Machine Learning*, C. Sammut and G. Webb, ed., Springer Verlag, 2010.

[7] T.G. Dietterich, "Ensemble Methods in Machine Learning," *Proc. Int'l Workshop Multiple Classifier Systems*, J. Kittler and F. Roli, ed., pp. 1-15, 2000.

[8] J.L. Fleiss, *Statistical Methods for Rates and Proportions*. John Wiley & Sons, 1981.

[9] A. Frank and A. Asuncion, "UCI Machine Learning Repository," 2010.

[10] Y. Freund and R.E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *J. Computer and System Sciences*, vol. 55, no. 1, pp. 119-139, 1997.

[11] L. Gupta, S. Kota, D.L. Molfese, and R. Vaidyanathan, "Diversity-Based Selection of Components for Fusion Classifiers," *Proc. Ann. Int'l Conf. IEEE Eng. in Medicine and Biology Soc.*, vol. 1, pp. 6304-6307, 2010.

[12] T.K. Ho, "The Random Subspace Method for Constructing Decision Forests," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832-844, Aug. 1998.

[13] L.I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*. John Wiley and Sons, 2004.

[14] L.I. Kuncheva and C.J. Whitaker, "Measures of Diversity in Classifier Ensembles," *Machine Learning*, vol. 51, pp. 181-207, 2003.

[15] L.I. Kuncheva and J.J. Rodríguez, "Classifier Ensembles with a Random Linear Oracle," *IEEE Trans. Knowledge and Data Eng.*, vol. 19, no. 4, pp. 500-508, 2007.
[16] L.I. Kuncheva and C.J. Whitaker, "Ten Measures of Diversity in Classifier Ensembles: Limits for Two Classifiers," *Proc. IEE Workshop Intelligent Sensor Processing*, pp. 10/1-10/6, 2001.
[17] D.D. Margineantu and T.G. Dietterich, "Pruning Adaptive Boosting," *Proc. 14th Int'l Conf. Machine Learning*, pp. 378-387, 1997.
[18] J. Meynet and J.-P. Thiran, "Information Theoretic Combination of Pattern Classifiers," *Pattern Recognition*, vol. 43, no. 10, pp. 3412-3421, 2010.
[19] R. Polikar, "Ensemble Based Systems in Decision Making," *IEEE Circuits and Systems Magazine*, vol. 6, no. 3, pp. 21-45, third quarter 2006.
[20] J.J. Rodríguez, L.I. Kuncheva, and C.J. Alonso, "Rotation Forest: A New Classifier Ensemble Method," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1619-1630, Oct. 2006.
[21] L. Rokach, "Taxonomy for Characterizing Ensemble Methods in Classification Tasks: A Review and Annotated Bibliography," *Computational Statistics & Data Analysis*, vol. 53, no. 12, pp. 4046-4072, Oct. 2009.
[22] L. Rokach, "Collective-Agreement-Based Pruning of Ensembles," *Computational Statistics & Data Analysis*, vol. 53, no. 4, pp. 1015-1026, 2009.
[23] L. Rokach, "Ensemble-Based Classifiers," *Artificial Intelligence Rev.*, vol. 33, pp. 1-39, Feb. 2010.
[24] *Proc. Int'l Workshops Multiple Classifier Systems*, F. Roli, J. Kittler, T. Windeatt, N. Oza, R. Polikar, M. Haindl, J.A. Benediktsson, N. El-Gayar, and C. Sansone, eds., Lecture Notes in Computer Science series, Springer-Verlag, 2000-2012.
[25] E.K. Tang, P.N. Suganthan, and X. Yao, "An Analysis of Diversity Measures," *Machine Learning*, vol. 65, no. 1, pp. 247-271, 2006.
[26] G. Valentini and M. Re, "Ensemble Methods: A Review," *Advances in Machine Learning and Data Mining for Astronomy, Data Mining and Knowledge Discovery*, M.J Way, J.D. Scargle, K.M. Ali, and A.N. Srivastava, eds. Chapman & Hall/CRC Press, 2012.

Ludmila I. Kuncheva received the MSc degree from the Technical University of Sofia, Bulgaria, in 1982, and the PhD degree from the Bulgarian Academy of Sciences in 1987. Until 1997, she worked at the Central Laboratory of Biomedical Engineering at the Bulgarian Academy of Sciences. She is currently a professor in the School of Computer Science, Bangor University, United Kingdom. Her interests include pattern recognition and classification, machine learning, classifier combination and fMRI data analysis. She has published two books and more than 150 scientific papers. She is a member of the IEEE.