Bayesian Linear Combination of Neural Networks

Battista Biggio, Giorgio Fumera, and Fabio Roli

University of Cagliari,
Department of Electrical and Electronic Engineering,
Piazza d'Armi, I-09123 Cagliari, Italy

1 Introduction

Classifier ensembles have been one of the main topics of interest in the neural networks, machine learning and pattern recognition communities during the past fifteen years [21,28,16,17,26,36,27,23,11]. They are currently one of the state of the art techniques available for the design of classification systems and an effective option to the traditional approach based on the design of a single, monolithic classifier in many applications. Broadly speaking, two main choices have to be made in the design of a classifier ensemble: how to generate individual classifiers and how to combine them. Two main approaches have emerged to deal with these design steps: coverage optimisation, focused on generating an ensemble of classifiers as much complementary as possible, which are then fused with simple combining rules, and decision optimisation, focused on finding the most effective combining rule to exploit at best a given classifier ensemble [21]. One of the most studied and widely used combining rules, especially in the former approach, is the linear combination of classifier outputs. Linear combiners are often used for neural network ensembles, given that neural networks provide continuous outputs. The simplicity of linear combiners and their continuous nature favoured the development of analytical models for the analysis of the performance of ensembles of predictors, both for the case of regression problems and for the relatively more complex case of classification problems.

In this chapter, we give an overview on ensembles of linearly combined neural networks. Our survey is focused on a Bayesian analytical model introduced about ten years ago in works by K. Tumer and J. Ghosh [31,32] and recently extended by the authors [8,4]. Basically, this model allows to quantify the advantage attainable by linearly combining an ensemble of classifiers, in terms of the reduction in misclassification probability. Although based on strict assumptions to make it analytically tractable, this model allows to point out the main factors which affect the performance of linearly combined classifier ensembles and suggests simple guidelines for their design. It was also recently exploited to develop a novel method for training ensembles of linearly combined neural networks [37] and to analyse the behaviour of bagging (a well known technique for constructing classifier ensembles) as a function of the ensemble size [9].

This chapter starts with an overview of past works on ensembles of linearly combined neural networks, both for regression and classification problems (section 2). The analytical model by Tumer and Ghosh is then presented, followed by the extension given by the authors, and its main results and implications are discussed in section 3. Finally, some experimental results are reported in section 4 to illustrate the main results of this model.

M. Bianchini et al. (Eds.): Innovations in Neural Infor. Paradigms & Appli., SCI 247, pp. 201–230.
springerlink.com

1.1 Notation

In the rest of this work we will use the following notation for network outputs. Given a feature vector $\mathbf{x}$, the output of the i-th output unit of a neural network (corresponding to the i-class of the problem) will be denoted as $f_i(\mathbf{x})$. In the case of two-class problems, in which a network with only one output unit is usually used, the output will be simply denoted as $f(\mathbf{x})$. When an ensemble of neural networks is considered, the outputs of the m-th individual networks will be denoted with the superscript m (for instance, $f_i^m(\mathbf{x})$ denotes the i-th output of the m-th network). The outputs obtained by the linear combination will instead be denoted with the superscript 'sa' (standing for 'simple average'), if the weights are identical, and with 'wa' (standing for 'weighted average'), if the weights can be different (for instance, $f_i^{\mathrm{sa}}(\mathbf{x})$ denotes the i-th output resulting from the simple average rule).

2 Overview of Past Works on Ensembles of Neural Networks

The aim of this section is to give an overview of the literature on ensembles of neural networks, focusing on fusion strategies based on the linear combination of network outputs. We point out that, although in this chapter we focus on classification problems, we will also review some relevant works on linearly combined neural networks for regression problems, since results obtained for the latter kind of problems often apply to the former as well.

Let us start by listing the different schemes for the linear combination of network outputs proposed in the literature. Given a C-class problem (or a regression problem involving a vector function with C values) and an ensemble of N neural networks, each one with C output units, the simplest and most used kind of linear combiner consists in separately averaging each of the C outputs over the N networks, for a given input sample $\mathbf{x}$, using one constant weight for each network, identical for all outputs:

$$f_i^{\mathrm{wa}}(\mathbf{x}) = \sum_{m=1}^{N} w^m f_i^m(\mathbf{x}), \quad i = 1, \ldots, C. \tag{1}$$

A more complex scheme consists in using weights that depend on the class [33]:

$$f_i^{\mathrm{wa}}(\mathbf{x}) = \sum_{m=1}^{N} w_i^m f_i^m(\mathbf{x}), \quad i = 1, \ldots, C. \tag{2}$$

The most general scheme involving constant weights consists in linearly combining *all* the outputs of *all* the networks to compute each of the $y_i^{\mathrm{sa}}(\mathbf{x})$ [3,33]:

$$f_i^{\mathrm{wa}}(\mathbf{x}) = \sum_{m=1}^{N} \sum_{j=1}^{C} w_{ij}^m f_j^m(\mathbf{x}), \quad i = 1, \ldots, C. \tag{3}$$

Finally, the case of weights dependent on the input sample has been considered by Tresp and Taniguchi [29]:

$$f_i^{\mathrm{wa}}(\mathbf{x}) = \sum_{m=1}^{N} w^m(\mathbf{x}) f_i^m(\mathbf{x}), \quad i = 1, \ldots, C. \tag{4}$$

The only comment we make here, on the different weighting schemes, is that a higher complexity leads to a higher flexibility and, thus, to a better capability to fit the unknown function to approximate (either the discriminant function in a classification problem, or a continuous-valued function in a regression problem). However, reliably estimating a larger number of weights usually requires a larger training set. Therefore, in practice, the theoretical superiority of a more complex weighting scheme over a simpler can be cancelled out by a too small training set. We point out that this is just a specific case of a more general and well known issue in the field of multiple classifier systems, namely the trade-off between the complexity of a combining rule and the amount of training data required to exploit its potential effectiveness.

Works on linearly combined neural networks can be broadly subdivided into two groups. One of the groups includes works whose aim is to devise methods to estimate the optimal weights for one of the linear combination schemes mentioned above (namely the weights that minimise either the minimum squared error, MSE, or the misclassification probability of the ensemble) and, for a given ensemble of individual networks, sometimes exploiting analytical results derived under some assumptions about the output distribution of individual networks. The other group includes works that provide some theoretical investigation on the performance of linearly combined neural network ensembles, resulting in guidelines on their design and sometimes in suggesting a weight estimation method.

Among works in the first group, the ones by Perrone and Cooper [24], Hashem and Schmeiser [14] and Hashem [13] provide similar results. They focused on the simplest linear combination scheme (1) and derived the analytical expressions of the optimal weights which minimise the expected value (over the input variables) of the MSE of a neural network ensemble as a function of the covariances between the outputs of the individual networks. Perrone and Cooper [24] considered only the case of positive weights that sum up to 1, while Hashem and Schmeiser [14] and Hashem [13] considered also the more general cases of unconstrained weights (as well as the case of an additional term in the linear combination, w_0, which makes sense only for regression problems). In particular, all these works show that when the estimation errors (with respect to the target function) of the individual networks are unbiased and uncorrelated, then the optimal weights are inversely proportional to the variance of the output of the corresponding network. In this case, Perrone and Cooper [24] showed that simply averaging the individual networks (using identical weights) leads to an MSE lower or at worst identical to the average MSE of the individual networks. The expressions of the optimal weights derived in these works require the inversion of the matrix containing the covariances between the outputs of individual networks. The authors pointed out that collinearity in these matrices (due for instance to different networks with highly correlated outputs) makes the weights computation unreliable. Some robust weights estimation methods were discussed by Hashem [13], including the selection of a subset of the available individual networks.

Benediktsson et al. [3] used the more complex weighting scheme 3 and derived the weights that minimise the squared error with respect to target values, computed on a given data set. As in the works mentioned above, even in this case computing the

weights requires matrix inversion: robust estimation methods are discussed to avoid computational problems.

Tresp and Taniguchi [29] investigated scenarios in which it can be useful to use combination weights that depend on the feature vector of the input sample, as in 4. This can happen when individual networks are trained to solve the same problem but exhibit different statistical characteristics (like a different output variance) in different areas of the feature space, or when they are trained to solve subproblems of the original problem and thus exhibit different strengths or "expertise" in different subsets of the feature space. Methods for computing the optimal weights in both cases are discussed. In particular, if in the former case the goal is to minimise the variance of the combined estimate of the target function, it is shown that the solution is analogous to the one derived by Perrone and Cooper [24], namely the weights must be inversely proportional to the variance of the output of the corresponding networks (on a given point of the feature space).

The works mentioned so far focus on minimising the MSE of a neural network ensemble. Although this approach can be used also in classification problems, it can lead to suboptimal solutions: it is indeed known that minimising the MSE is not equivalent to minimising misclassification probability. Ueda [33] deals with this issue and proposes a weight estimation method tailored to classification problems, based on an optimisation algorithm aimed at minimising the misclassification probability of a neural network ensemble.

Among works in the second group we mention first the one by Krogh and Vedelsby [19], in which a particular expression for the expected MSE of an ensemble of linearly combined neural networks is derived, given by the sum of the weighted average of individual networks MSE and of a term (named "ambiguity") depending only on the correlation between their outputs. This is a kind of bias-variance decomposition, but is different than the one commonly used (derived originally by [10]). The decomposition derived by Krogh and Vedelsby shows that, to obtain an ensemble with a small MSE, it is necessary that the individual networks exhibit a small average MSE and that their outputs are as low correlated as possible. However, these are known to be almost opposite goals (as for the bias and variance components of the MSE), so a trade-off has to be achieved between them. Differently from works in the other group mentioned above, the results by Krogh and Vedelsby do not provide an analytical expression for the optimal weights. However, they clearly show that, to obtain an effective neural network ensemble, the correlation between individual networks has to be taken into account, as well as their individual performances, and this can be attained during ensemble construction. Furthermore, these results also suggest that also unlabelled samples may be useful for estimating the optimal weights by minimising an estimate of the MSE, since the ambiguity term does not depend on target values.

The results by Krogh and Vedelsby were exploited by Brown et al. [6] to provide a theoretical support to *negative correlation learning*, a method originally proposed by Liu [22]. Such method consists in training *in parallel* the individual members of a linearly combined neural network ensemble by a backpropagation-like learning algorithm, in which the error function of each network is given by its individual error measure (as in standard backpropagation) minus a term depending on the correlation between the

outputs of the individual network. As shown by Brown et al. [6], it turns out that (for a proper choice of its parameters), the negative correlation learning algorithm seeks to minimise the overall ensemble error, as given by the ambiguity decomposition.

One of the main theoretical contributions to the field of classifier ensembles was given by Kittler et al. [15], who developed a common theoretical framework for several classifier combination strategies for the case when individual classifiers use distinct feature subsets that are conditionally independent given the class. In particular, this work showed that several combining strategies, including the sum rule (a variant of the simple average rule), can be derived under some approximations from the product rule. A further analysis of the simple average rule and a comparison with other combining strategies was also reported by Kittler and Alkoot [18] and by Kuncheva [20].

Another relevant contribution, related to linear combiners, was given in works by Tumer and Ghosh [31,32], that were further extended by the authors in [8]. Basically, these works provided an analytical framework for quantifying the reduction of the misclassification probability which can be attained by linearly combining an ensemble of classifiers which provide estimates of the a posteriori probabilities, as a function of mean, variance and correlation of estimation errors. Although the framework by Tumer and Ghosh is based on rather strict assumptions, it was shown, by Fumera and Roli [8], that it allows to accurately predict some qualitative aspects of the behaviour of linear combiners and, thus, to provide some useful practical guidelines for their design. These works will be described in more detail in the next section, where we also present an extension of the framework by Tumer and Ghosh based on less strict assumptions, proposed in [4].

A relevant application of the results derived in the mentioned works by Tumer and Ghosh and by Fumera and Roli was recently proposed by Zanda et al. [37]. This work generalised the concept of the ambiguity decomposition, previously defined only for regression problems, to classification problems, and proposed an algorithm based on the negative correlation learning framework, which applies to ensembles of linearly combined classifiers. The results derived by Tumer and Ghosh and by Fumera and Roli were also exploited by the authors in [9] to analyse the behaviour of the misclassification probability of linearly combined classifiers constructed with the bagging method, as a function of the ensemble size.

For the sake of completeness, we conclude this section mentioning some of the most relevant works on neural network ensembles based on fusion strategies different from the linear combination. The most interesting strategies are perhaps voting-based ones like majority and plurality. Two relevant works on this kind of fusion strategies are the ones by Hansen and Salamon [12] and by Battiti and Colla [2]. Hansen and Salamon considered both the case of neural networks that make independent classification errors on a given sample and the case of dependent errors. In the latter case, to model the effect of correlated errors, they extended the model proposed by Eckhardt and Lee[7] to classification problems, to study software reliability through the use of multiple versions of the same program. Battiti and Colla [2] considered instead a more general kind of classification problems, namely classification problems with the reject option, in which a classifier may decide to withhold assigning an input pattern to one of the predefined classes, if it is not sufficiently confident in the correctness of its classification.

Besides experimental results, both works provided some analytical result to quantify the reduction of the classification error attainable by classifier ensembles, mainly under the assumption of independence among the errors of individual classifiers. It is also worth mentioning the work by Rogova [25], who proposed a combination method based on the Dempster-Shafer theory of evidence, tailored to the continuous-valued outputs provided by neural network classifiers.

3 An Analytical Model for Linear Combiners

In this section, we focus on a theoretical analysis of linearly combined classifiers and of neural networks in particular, based on an analytical model derived in works by Tumer and Ghosh and further extended by the authors. The relevance of this model is due to the fact that it is one of the few theoretical models developed so far in the field of classifier ensembles and that it proved to be useful both to improve the understanding of a widely used combining strategy as the linear combination, and to derive practical guidelines for the design of linearly combined classifier ensembles.

In regression problems, it is relatively easy to analytically derive the optimal weights of the linear combination of an ensemble of regressors and compare the performance of individual regressors and of their combination. This is due to the fact that the loss function (typically, the MSE) is usually continuous. Obtaining analogous analytical results for classification problems is known to be much more difficult, since the loss function is discrete. For instance, if the misclassification probability is used as performance measure, the loss function is 0 for correct classifications and 1 for misclassifications. Tumer and Ghosh [30,31,32] developed a model that partially overcomes this problem, allowing to analytically compute and compare the misclassification probability of individual classifiers and of a linear combination of classifiers, under some rather strict assumptions which make analytical derivations possible (it is worth noting that the model was applied also to order statistics combiner in [32]). Their model applies to classifiers which provide estimates of the class posterior probabilities and, thus, also to neural networks. Tumer and Ghosh limited their analysis to the simple average combining rule, providing interesting insights about the factors that affect the performance of this rule. Their model and the main results of their analysis are described in section 3.1. The authors [8] exploited this model to analyse the more general case of the weighted average rule, deriving general guidelines for the design of linear combiners. This work is summarised in section 3.2. As mentioned above, the model by Tumer and Ghosh is based on rather strict assumptions that may not hold in real classification problems. This motivated the authors to investigate whether a model for linear combiners could be derived under less strict assumptions. Another motivation was given by the observation that (qualitative) predictions derived from the model by Tumer and Ghosh were shown to hold with good accuracy on some real data sets [8]: this raised the issue of better understanding the conditions under which such predictions can be expected to hold. A partial answer to these questions was given in [4], where the authors derived an extension of the model by Tumer and Ghosh by relaxing one of its assumptions. This extension is described in section 3.4.

3.1 The Analytical Model by Tumer and Ghosh

The goal of the model by Tumer and Ghosh is to provide an analytical expression of the misclassification probability of individual and linearly combined classifiers, allowing to compare them. The model is focused on classifiers which provide estimates of the class posterior probabilities, like neural networks. The above goal is difficult due to the non-continuous loss function used in classification problems, as mentioned before. Indeed, many works limited their analysis on a single point in the feature space [15,20,18]. Since analytically computing the overall misclassification probability (over the whole feature space) seems not possible in general, the idea of Tumer and Ghosh was to consider at least a relevant subset of the feature space, namely the neighbourhood of a given class boundary, and to investigate the case in which the individual classifiers produce a boundary between the same two classes in that neighbourhood. This is a reasonable assumption, if a classifier provides accurate estimates of the a posteriori probabilities. To further simplify the computations, the analysis was limited to a one-dimensional feature space. In this case, the difference between the estimated and the ideal boundary (denoted in the following respectively as x_b and x^*) can be simply represented as a shift of the former from the latter, by an amount b given by $b = x_b - x^*$. An example is depicted in Fig. 1, involving a boundary between any two classes ω_i and ω_j. When the estimated class boundaries are used, the contribution of the considered subset of the feature space to the overall misclassification probability can be subdivided into a Bayes error, due to the overlap of the class posterior probabilities (corresponding to the light grey area of Fig. 1), and an *added error*, due to the mismatch between the ideal and estimated boundaries (corresponding to the dark grey area of Fig. 1). The added error is due to the fact that, if $b > 0$ as in Fig. 1, samples in the interval $[x^*, x_b]$ are assigned by the estimated boundaries to class ω_i instead of class ω_j. The added error with respect to Bayes error is given by the following expression (note that it holds both when $b > 0$ and when $b < 0$):

$$E_{\text{add}} = \int_{x^*}^{x^*+b} \Big[P(\omega_j|x) - P(\omega_i|x) \Big] \cdot p(x) dx.$$

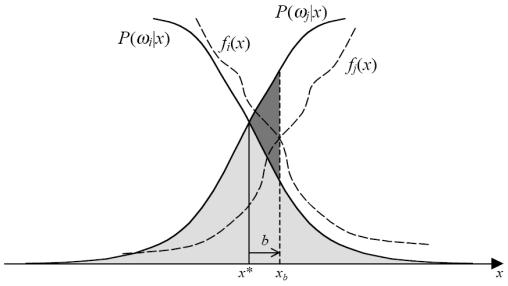

Fig. 1. True posteriors (*solid lines*) between classes ω_i and ω_j around the optimal boundary x^*. Estimated posteriors (*dotted lines*) lead to a different boundary, shifted from x^* to x_b by an amount b, and to an added error (*dark grey area*) over the Bayes error (*light grey area*)

Now the idea of Tumer and Ghosh is to compute the above added error as a function of the estimation errors made by a classifier on the a posteriori probabilities. To this aim, without loosing generality, they write the classifier's output (namely, the estimated posterior probability) for a generic class ω_k and for a point x in the feature space as:

$$f_k(x) = P(\omega_k|x) + \varepsilon_k(x), \tag{5}$$

where $\varepsilon_k(x)$ denotes the estimation error. To compute the added error, Tumer and Ghosh make a first-order approximation of the true posteriors and a zero-order approximation of $p(x)$ around x^* (note that this approximation is reasonable, if the classifier provides accurate estimates of the a posteriori probabilities as assumed above and, thus, the estimated boundary is close to ideal one):

$$P(\omega_k|x) \simeq P(\omega_k|x^*) + (x - x^*) \cdot P'(\omega_k|x), \tag{6}$$

where $P'(\omega_k|x)$ is the first derivative of $P(\omega_k|x)$ with respect to x. Note that under this approximation, the corresponding added error region (the dark grey area in fig. 1) becomes a triangle. The added error can thus be approximated as

$$\begin{aligned} e_{\text{add}} &= \int_{x^*}^{x^*+b} \Big[P(\omega_j|x) - P(\omega_i|x) \Big] \cdot p(x)dx \\ &\simeq \int_{x^*}^{x^*+b} \Big[P(\omega_j|x^*) - P(\omega_i|x^*) \\ &+ (x - x^*) \cdot [P'(\omega_j|x) - P'(\omega_i|x)] \Big] p(x^*)dx \\ &= \frac{p(x^*)t}{2} b^2 \end{aligned} \tag{7}$$

where $t = P'(\omega_j|x_b) - P'(\omega_i|x_b)$. Finally, b can be expressed as a function of the estimation errors, by noting first that $f_i(x_b) = f_j(x_b)$ (since the estimated posteriors of classes ω_i and ω_j are by definition identical on x_b) and then rewriting this equality, using the above approximation for the posteriors (note also that $P(\omega_j|x^*) = P(\omega_i|x^*)$):

$$P(\omega_i|x^*) + b \cdot P'(\omega_i|x_b) + \varepsilon_i(x_b) = P(\omega_j|x^*) + b \cdot P'(\omega_j|x_b) + \varepsilon_j(x_b),$$

$$b = \frac{\varepsilon_j(x_b) - \varepsilon_i(x_b)}{t}. \tag{8}$$

Up to now, classifier outputs $f_k(x)$ have been considered as fixed. In practice, they are random variables, since they depend on a random training set used for classifier training and, possibly, on random parameters of the learning algorithm (for instance, the initial values of the connection weights in a neural network). According to eq. 5, this means that the estimation errors $\varepsilon_k(x)$ are random variables, which implies that the shift b between the ideal and the estimated boundary, and the added error 7, are random variables as well. For this reason, the performance measure usually considered in classification problems is the *expected* misclassification probability over training sets. Assuming that each realisation of the classifier outputs provides an estimated boundary

between classes ω_i and ω_j in a neighbourhood of x^*, it makes sense to compute the expected value of the added error 7, which is given by:

$$E_{\text{add}} = \frac{p(x^*)t}{2}(\beta_b^2 + \sigma_b^2), \tag{9}$$

where β_b and σ_b^2 denote respectively the mean and variance of b. From 8, assuming that estimation errors on different classes ($\varepsilon_i(x)$ and $\varepsilon_j(x)$, $i \neq j$) are uncorrelated, it easily follows that

$$\beta_b = \frac{\beta_i - \beta_j}{t}, \quad \sigma_b^2 = \frac{\sigma_i^2 + \sigma_j^2}{t^2}, \tag{10}$$

where β_k and σ_k^2 are respectively the mean and variance of the estimation error $\varepsilon_k(x_b)$.

Consider now an ensemble of N classifiers which are combined by averaging their outputs. The corresponding estimated posterior for class ω_k is given by

$$f_k^{\text{sa}}(x) = \frac{1}{N}\sum_{m=1}^{N} f_k^m(x) = P(\omega_k|x) + \varepsilon_k^{\text{sa}}(x), \tag{11}$$

where

$$\varepsilon_k^{\text{sa}}(x) = \frac{1}{N}\sum_{m=1}^{N} \varepsilon_k^m(x). \tag{12}$$

Note that eq. 12 simply states that the estimation error of the linear combination is the average of the estimation errors made by the individual classifiers. Assuming that, for each realisation of the N classifiers, also their linear combination provides an estimated boundary $x_{b^{\text{sa}}}$ between classes ω_i and ω_j in a neighbourhood of x^*, repeating the same computations above one easily finds that the added error of the ensemble is

$$e_{\text{add}}^{\text{sa}} = \frac{p(x^*)t}{2}(b^{\text{sa}})^2, \tag{13}$$

and its expected value is given by

$$E_{\text{add}}^{\text{sa}} = \frac{p(x^*)t}{2}(\beta_{b^{\text{sa}}}^2 + \sigma_{b^{\text{sa}}}^2), \tag{14}$$

where b^{sa} denotes the shift $x_{b^{\text{sa}}} - x^*$, which is given by

$$b^{\text{sa}} = \frac{\varepsilon_i^{\text{sa}}(x_{\text{b}^{\text{sa}}}) - \varepsilon_j^{\text{sa}}(x_{\text{b}^{\text{sa}}})}{t}, \tag{15}$$

while $\beta_{b^{\text{sa}}}$ and $\sigma_{b^{\text{sa}}}^2$ are the mean and variance of b^{sa}. The mean $\beta_{b^{\text{sa}}}$ can be written as

$$\beta_{b^{\text{sa}}} = \frac{\beta_i^{\text{sa}} - \beta_j^{\text{sa}}}{t} = \frac{1}{N}\sum_{m=1}^{N}\frac{\beta_i^m - \beta_j^m}{t} = \frac{1}{N}\sum_{m=1}^{N}\beta_{b^m}, \tag{16}$$

where β_{b^m} is given by 10 (now we use the superscripts to denote the different individual classifiers). The expression of the variance $\sigma_{b^{\text{sa}}}^2$, can be obtained by noting that eq. 12 implies that the variance $(\sigma_k^{\text{sa}})^2$ of the estimation error $\varepsilon_k^{\text{sa}}(x_{\text{b}^{\text{sa}}})$ is given by

$$(\sigma_k^{\text{sa}})^2 = \frac{1}{N^2}\sum_{m=1}^{N}(\sigma_k^m)^2 + \frac{1}{N^2}\sum_{m=1}^{N}\sum_{n\neq m}\rho_k^{mn}\sigma_k^m\sigma_k^n, \tag{17}$$

where ρ_k^{mn} is the correlation coefficient between $\varepsilon_k^m(x_{\mathrm{b}^{\mathrm{sa}}})$ and $\varepsilon_k^n(x_{\mathrm{b}^{\mathrm{sa}}})$, and σ_k^m is the standard deviation of $\varepsilon_k^m(x_{\mathrm{b}^{\mathrm{sa}}})$. Assuming that estimation errors on different classes, $\varepsilon_i^m(x)$ and $\varepsilon_j^n(x)$ $i \neq j$, are uncorrelated also for the classifier ensemble, from eq. 15 it turns out that $\sigma^2_{b^{\mathrm{sa}}} = \frac{1}{t^2}[(\sigma_i^{\mathrm{sa}})^2 + (\sigma_j^{\mathrm{sa}})^2]$. Therefore, using eq. 17 one finally obtains:

$$\sigma^2_{b^{\mathrm{sa}}} = \frac{1}{N^2}\sum_{m=1}^{N}\sigma^2_{b^m} + \frac{1}{t^2}\frac{1}{N^2}\sum_{m=1}^{N}\sum_{n\neq m}(\rho_i^{mn}\sigma_i^m\sigma_i^n + \rho_j^{mn}\sigma_j^m\sigma_j^n), \tag{18}$$

where $\sigma^2_{b^m}$ is given by eq. 10.

Let us now summarise the main results above. First, under the assumption and the approximations mentioned above (which will be further discussed in section 3.3), the added error is proportional to the squared distance (the shift) between the ideal and the estimated boundary (see eqs. 7 and 13). It follows that its expected value is proportional to the sum of two terms: one depending on the bias of the boundary shift, the other on its variance (eqs. 9 and 14). In particular, the bias of the boundary shift depends only on the bias of the estimation errors, according to eqs. 10 and 16, while its variance depends on the variance of the estimation errors, according to eqs. 10 and 18 and, for the linear combiner, also on the correlation between the estimation errors of different classifiers on the same class (18). It is worth noting that this can be considered as a bias-variance decomposition of the misclassification probability, related to a subset of the feature space.

Given that the expected added error both of individual classifiers and of their linear combination is given in terms of the bias and variance of estimation errors of individual classifiers, it becomes possible to compare the two expressions to quantify the reduction attainable by the linear combination. The comparison can be made separately for the bias and variance components. In the following, we summarise the main results of this comparison reported by Tumer and Ghosh [31,32] and by Fumera and Roli [8].

First, eq. 16 shows that the bias of the boundary shift of the linear combiner $\beta_{b^{\mathrm{sa}}}$ is the average of the individual boundary shift biases, $\beta_{b^1}, \ldots, \beta_{b^N}$. This means that the bias component of the expected added error of the linear combination 14 is between the minimum and the maximum of the bias terms of individual networks:

$$\min_m \beta_b^m \leq \beta_{b^{\mathrm{sa}}} \leq \max_m \beta_b^m. \tag{19}$$

With regard to the variance components 10 and 18, an analytical comparison is possible only under some simplifying assumptions. If the variances of the estimation errors are all identical (namely, $(\sigma_k^m)^2 = \sigma^2$ for each k and m), as well as the correlations on different classes (namely, $\rho_i^{mn} = \rho_j^{mn}$ for each m, n), then eq. 18 becomes

$$\sigma^2_{b^{\mathrm{sa}}} = \frac{1 + (N-1)\delta_{ij}}{N}\sigma_b^2, \tag{20}$$

where $\sigma_b^2 = 2\frac{\sigma^2}{t^2}$ (from 10), $\delta_{ij} = \frac{\delta_j + \delta_j}{2}$, and $\delta_k = \frac{1}{N(N-1)}\sum_{m=1}^{N}\sum_{n\neq m}\rho_k^{mn}$. Noting that $\delta_k \geq -\frac{1}{N-1}$, one gets $0 \leq \frac{1+(N-1)\delta_{ij}}{N} \leq 1$. In particular, this term equals 1, if the estimation errors exhibit the maximum positive correlation, $\rho_k^{mn} = 1$, for each

m, n. In this case, the variance component of the linear combination is identical to the one of each individual classifier: no reduction is attained by combining classifiers. Instead, if the estimation errors are uncorrelated ($\rho_k^{mn} = 0$), then the term $\frac{1+(N-1)\delta_{ij}}{N}$ becomes zero, and we have $\sigma^2_{b^{\mathrm{sa}}} = \sigma^2_b/N$: this means that combining networks with uncorrelated estimation errors (equivalently, uncorrelated outputs), the variance component of the expected added error is reduced by a factor equal to the ensemble size, N. Finally, if the correlation between estimation errors is negative and attains the minimum possible value, then $\frac{1+(N-1)\delta_{ij}}{N} = 0$, which implies $\sigma^2_{b^{\mathrm{sa}}} = 0$: in other words, combining negatively correlated networks can allow to reduce to zero the variance component of the expected added error. In the most general case of different variances and correlations, it is not possible to analytically compute the reduction of the variance component with respect to individual classifiers, but it is easy to show from 18 that the variance component of the linear combination can not be greater than the maximum variance component among individual classifiers: $0 \leq \sigma^2_{b^{\mathrm{sa}}} \leq \max_m \sigma^2_{b^m}$. In particular, the lower the correlations, the lower $\sigma^2_{b^{\mathrm{sa}}}$. This clearly shows that linearly combining as low correlated networks as possible is always beneficial in classification problems, as well as in regression problems.

Putting together the conclusions drawn above about the bias and variance components, it follows that by simple averaging an ensemble of networks one is guaranteed to obtain bias and variance components of the expected added error not higher than the maximum corresponding components of individual classifiers. However, while there is no direct way to control the bias component of the ensemble, the variance component can be reduced by combining as low correlated networks as possible. Therefore this suggests the following strategy to face the well known bias-variance dilemma: to construct an effective ensemble one should use individual classifiers with as low bias as possible (since it is not necessarily reduced by averaging networks), while the resulting high variance will be reduced by averaging them, provided that they exhibit low correlated outputs [32]. With regard to this issue, it is commonly believed that it is difficult to obtain a large number of classifiers that exhibit uncorrelated or even negatively correlated estimation errors, as claimed also by Tumer and Ghosh [32]. However, we point out that actually it is rather easy to obtain classifiers which make estimation errors $\varepsilon_k^m(x), \varepsilon_k^n(x)$ that are uncorrelated on each given point x in feature space, as shown in [9].

3.2 Application of the Model to the Weighted Average Rule

Simple averaging the outputs of a network ensemble is a widely used and very simple combination strategy, which does not require to set the value of any parameter. Weighted averaging is a more general and more flexible strategy, which however requires to estimate the best combination weights, usually from a labelled data set. The model by Tumer and Ghosh was applied to the analysis of the weighted average combining rule by the authors [8], with the main aim to investigating the reduction of the expected added error attainable both with respect to individual classifiers and to the simple average rule, provided that the optimal combination weights are used. In this section we summarise the main results of our analysis. We point out that the problem of weights

estimation was previously investigated by several authors (see section 2), and was not considered in [8].

Let us start by the expression of the estimated posterior probabilities using the weighted average rule:

$$f_k^{\mathrm{wa}}(x) = \sum_{m=1}^{N} w_m f_k^m(x) = P(\omega_k|x) + \varepsilon_k^{\mathrm{wa}}(x), \tag{21}$$

where

$$\varepsilon_k^{\mathrm{wa}}(x) = \sum_{m=1}^{N} w_m \varepsilon_k^m(x). \tag{22}$$

The analysis in [8] was focused on the case of non-negative weights, which is usually considered in works on linear combiners. Without loosing generality, to simplify computations it is useful to add the constraint the weights sum up to 1:

$$w_m \geq 0 \quad m = 1, ..., N, \quad \sum_{m=1}^{N} w_m = 1. \tag{23}$$

Under the same assumptions and approximations made by Tumer and Ghosh, reported in section 3.1, denoting with b^{sa} the shift between the ideal boundary and the one estimated by the weighted average combiner, one obtains:

$$b^{\mathrm{wa}} = \frac{\varepsilon_i^{\mathrm{wa}}(x_{b^{\mathrm{wa}}}) - \varepsilon_j^{\mathrm{wa}}(x_{b^{\mathrm{wa}}})}{t}, \tag{24}$$

while the expected added error is given by

$$E_{\mathrm{add}}^{\mathrm{wa}} = \frac{p(x^*)t}{2}(\beta_{b^{\mathrm{wa}}}^2 + \sigma_{b^{\mathrm{wa}}}^2). \tag{25}$$

It is easy to see that we have again a bias component given by

$$\begin{aligned} \beta_{b^{\mathrm{wa}}}^2 = & \frac{1}{t^2} \sum_{m=1}^{N} w_m^2 (\beta_i^m - \beta_j^m)^2 \\ & + \frac{1}{t^2} \sum_{m=1}^{N} \sum_{n \neq m} w_m w_n (\beta_i^m - \beta_j^m)(\beta_i^n - \beta_j^n), \end{aligned} \tag{26}$$

and a variance component given by

$$\begin{aligned} \sigma_{b^{\mathrm{wa}}}^2 = & \frac{1}{t^2} \sum_{m=1}^{N} w_m^2 [(\sigma_i^m)^2 + (\sigma_j^m)^2] \\ & + \frac{1}{t^2} \sum_{m=1}^{N} \sum_{n \neq m} w_m w_n (\rho_i^{mn} \sigma_i^m \sigma_i^n + \rho_j^{mn} \sigma_j^m \sigma_j^n). \end{aligned} \tag{27}$$

To compare the expected added error above with the one of individual classifiers and of their simple averaging, it is first necessary to compute the optimal weights, defined the ones which minimise the expected added error 25 under constraints 23. From the expressions above, it is easy to see that this is a quadratic optimisation problem, but it turns out that it can be analytically solved only for particular values of the parameters (namely, of the biases, variances and correlations of the estimation errors of individual classifiers). A case of particular interest is when the estimation errors are unbiased and uncorrelated. In this case, the expected added error of the *m*-th individual classifier and of the weighted average are respectively:

$$E^m_{\text{add}} = \frac{p(x^*)}{2t}[(\sigma^m_i)^2 + (\sigma^m_j)^2], \tag{28}$$

$$E^{\text{wa}}_{\text{add}} = \frac{p(x^*)}{2t}\sum_{m=1}^{N} w^2_m[(\sigma^m_i)^2 + (\sigma^m_j)^2] = \sum_{m=1}^{N} w^2_m E^m_{\text{add}}. \tag{29}$$

In other words, by weighted averaging an ensemble of networks with unbiased and uncorrelated estimation errors, the corresponding expected added error is equal to the linear combination of the ones of individual classifiers, with squared weights. The optimal weights can be found by using the technique of Lagrange multipliers, and turn out to be inversely proportional to to expected added error of the corresponding network (note that this result is analogous to the one obtained for regression problems by Perrone and Cooper [24] and by Hashem [13]):

$$w_m = \left(\sum_{n=1}^{N} \frac{1}{E^n_{\text{add}}}\right)^{-1} \frac{1}{E^m_{\text{add}}}. \tag{30}$$

Finally, substituting 30 into 29 one obtains:

$$E^{\text{wa}}_{\text{add}} = \frac{1}{\frac{1}{E^1_{\text{add}}} + ... + \frac{1}{E^N_{\text{add}}}}. \tag{31}$$

Clearly, if one uses instead the simple average rule, the corresponding expected added error is:

$$E^{\text{sa}}_{\text{add}} = \frac{1}{N^2}\sum_{m=1}^{N} E^m_{\text{add}}. \tag{32}$$

To sum up, in the case of uncorrelated and unbiased estimation errors, the expected added error of the simple and weighted average are, respectively, $\frac{1}{N}$ times the arithmetic mean and $\frac{1}{N}$ times the harmonic mean of the added error of the individual classifiers. Note that, as one can expect, the optimal weights are $1/N$ (namely, simple averaging is the best linear combination strategy), if and only if all individual classifiers exhibit the same performance (namely, if their expected added errors are all identical).

The above expressions of the expected added error allow to compare the performance of the simple and the weighted average rules by taking into account only the expected added errors of the individual classifiers, sine they do not depend explicitly on the means, variances and correlations of their estimation errors. The comparison can

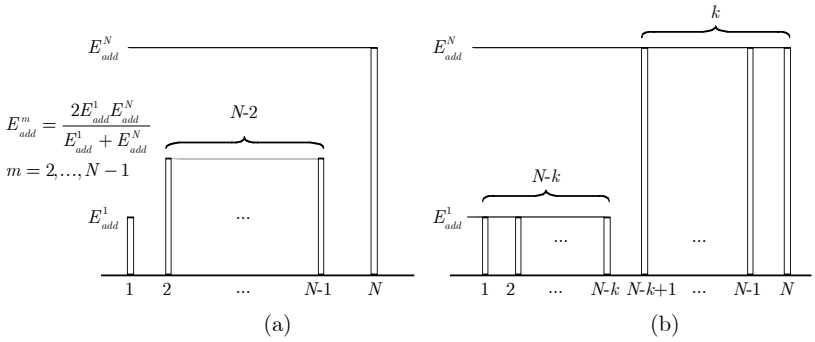

Fig. 2. For fixed N and error range width $E^N_{\rm add} - E^1_{\rm add}$, the patterns of the expected added error of the individual classifiers corresponding the maximum (a) and minimum performance imbalance (b) conditions is shown

be made, conveniently, in terms of the difference between the expected added errors of the simple and the weighted average, $E^{\rm sa}_{\rm add} - E^{\rm wa}_{\rm add}$, as a function of the lowest and the highest expected added error of individual networks (note that the above difference is always non negative, since when the optimal weights are used, the simple average can not outperform the weighted average). Without loosing generality, classifiers can be ordered for increasing values of their expected added error, so that classifier 1 is the best and classifier N is the worst: $E^1_{\rm add} \leq E^2_{\rm add} \leq ... \leq E^N_{\rm add}$. Now, for given values of $E^1_{\rm add}$ and $E^N_{\rm add}$, what is the behaviour of $E^{\rm sa}_{\rm add} - E^{\rm wa}_{\rm add}$ with respect to the performance of the other classifiers, $E^2_{\rm add}, \ldots, E^{N-1}_{\rm add}$? It turns out that $E^{\rm sa}_{\rm add} - E^{\rm wa}_{\rm add}$, namely the advantage of the weighted average over the simple average, is minimum when classifiers $2, \ldots, N$ exhibit the same expected added error, equal to

$$E^m_{\rm add} = 2\frac{E^1_{\rm add} \cdot E^N_{\rm add}}{E^1_{\rm add} + E^N_{\rm add}}. \tag{33}$$

This condition, depicted in Fig. 2(a), was named in [8] *minimum performance imbalance* condition, where the term *performance imbalance* was used to denote the fact that the individual networks exhibit different performances. Instead, $E^{\rm sa}_{\rm add} - E^{\rm wa}_{\rm add}$ is maximum when a subset of $N-k-1$ classifiers exhibit the same performance as the best one ($E^m_{\rm add} = E^1_{\rm add}, \quad m = 2, ..., N-k$), while the remaining ones exhibit the same performance of the worst one ($E^m_{\rm add} = E^N_{\rm add}, \quad m = N-k+1, ..., N-1$), where k is either given by $\lceil k^* \rceil$ or $\lfloor k^* \rfloor$, k^* being defined as:

$$k^* = N\frac{\sqrt{E^1_{\rm add} \cdot E^N_{\rm add}} - E^1_{\rm add}}{E^N_{\rm add} - E^1_{\rm add}}. \tag{34}$$

In particular, if $N = 3$, k always equals 2. This condition, named *maximum performance imbalance* condition, is depicted in Fig. 2(b).

Besides this qualitative analysis, a quantitative analysis was also given in [8]. As an example, Fig. 3 reports the values of $E^{\rm sa}_{\rm add} - E^{\rm wa}_{\rm add}$ under both the minimum and maximum performance imbalance condition, as a function of the difference between

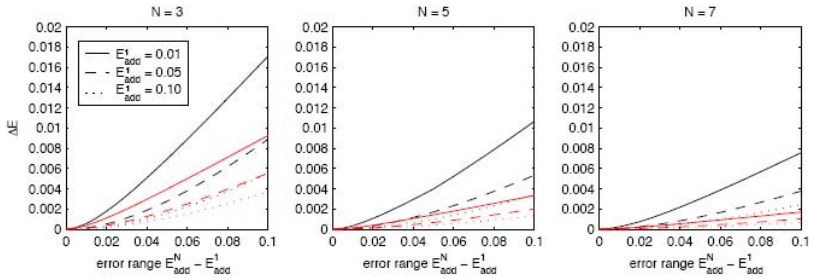

Fig. 3. Behaviour of $\Delta E = E^{\mathrm{sa}}_{\mathrm{add}} - E^{\mathrm{wa}}_{\mathrm{add}}$ under the maximum (black curves) and minimum (red curves) performance imbalance conditions, as a function of the error range width $E^N_{\mathrm{add}} - E^1_{\mathrm{add}}$. Each plot refers to a different ensemble size N. In each plot, three different values of E^1_{add} have been considered (0.01, 0.05, 0.10).

the expected added errors of the worst and the best individual classifiers, $E^N_{\mathrm{add}} - E^1_{\mathrm{add}}$. Each of the three plots refers to a different value of the ensemble size N. In each plot, three different values of E^1_{add} have been considered. A clear pattern of behaviour can be deduced from these plots: being equal the other conditions, the advantage of the weighted average over the simple average increases as the ensemble error range $E^N_{\mathrm{add}} - E^1_{\mathrm{add}}$ increases, as the performance of the best classifier increases and as the ensemble size decreases. However, the advantage predicted by the model by Tumer and Ghosh is quantitatively rather low, perhaps lower than one can expect: indeed the plots suggest that for $E^{\mathrm{sa}}_{\mathrm{add}} - E^{\mathrm{wa}}_{\mathrm{add}}$ to be higher than 0.01, one should combine a small ensemble of networks (say, no more than 5) exhibiting a high performance imbalance (in the sense defined above) and a high error range, namely, the ensemble should include at least one very accurate classifier (the expected added error E^1_{add} should be lower than 0.05) and a very poor one (the error range should be almost 0.10). In particular, it should be noted that, being equal the other conditions, the advantage of the weighted average strongly depends on the kind of performance imbalance, namely on the particular distribution of the performances of classifiers $2, \ldots, N-1$ with respect to the best and worst ones. In other words, a high error range $E^N_{\mathrm{add}} - E^1_{\mathrm{add}}$ is not a sufficient condition for the weighted average to be much more advantageous (provided that the optimal weights can be accurately estimated) than the simple average.

This analysis was extended to the case of unbiased and correlated classifiers in [8], but in this case it was not possible to derive analytically the optimal weights and the corresponding expected added error. Only a numerical analysis was therefore carried out. To simplify the analysis, only the case of variances and correlations identical for all network outputs was considered, which leads to an expression of $E^{\mathrm{wa}}_{\mathrm{add}}$ involving only the expected added errors of individual classifiers and the correlations between their outputs. Furthermore, only the ensemble sizes of 3 and 5 were considered. Here we omit the details, and report just the results. First, for any values of the correlations, the same maximum performance imbalance condition as in the case of uncorrelated errors holds (the value of k can not be determined analytically), while the minimum performance imbalance condition above does not hold. Second, for any values of the expected added error of individual networks and for any given range of correlation values, the

patterns of correlation values that lead to the maximum and minimum $E_{\text{add}}^{\text{sa}} - E_{\text{add}}^{\text{wa}}$ are analogous to the maximum and minimum performance imbalance conditions. This implies that, being equal the other conditions, the advantage of the weighted average over the simple average increases as the correlation range increases, and as the correlation assume either of the two extreme values of such range. In particular, the higher the correlation, the higher such advantage. The numerical analysis also showed that the advantage of the weighted average rule over the simple average decreases (being equal the other conditions) as the ensemble size increases, and that its amount is higher than in the case of uncorrelated classifiers, but only when the correlation is positive.

The above results can be summarised in terms of the following practical guidelines for the design of linear combiners: the weighted average rule can provide a significantly better performance than the simple average, especially in small ensembles including individual networks that exhibit high error and correlation ranges, and are positively correlated. Otherwise, the attainable improvement in misclassification probability is likely to be small, and to be even cancelled out if the quality or the size of the data set at hand does not allow a reliable weight estimation. These results represented an interesting novelty in the classifier ensemble literature, since no work up to then (not even experimental works) provided a so detailed analysis on the behaviour of linear combiners. However, it should be taken into account that these results have been derived from an analytical model based on several assumptions and approximations, and exhibiting some limitations. These were discussed by the authors in [8], and are reported in detail in section 3.3. Nevertheless, an experimental investigation on some real data sets carried out in [8], and further experiments made by the authors, reported in section 4, showed that experimental results agreed with good accuracy with the above theoretical predictions. All these facts immediately raised two questions. First, to what extent can the predictions drawn from Tumer and Ghosh model be expected to hold? In other words, are there conditions under which they cease to hold? Second, is it possible to derive an analytical model for linear combiners under less strict assumptions than the ones by Tumer and Ghosh? A further investigation on this issues lead the authors to a generalisation of Tumer and Ghosh model. This work will be described in section 3.4.

3.3 Limitations of the Model by Tumer and Ghosh

We described in the previous section the model by Tumer and Ghosh and its application to the analysis of the simple and weighted average combining rules. Here we point out the assumptions on which it is based and the corresponding limitations, and summarise the approximations used in developing the model. This discussion will help in understanding the scope of this model, and the extension developed by the authors, which is described in section 3.4.

Limitations

– The model focus on a neighbourhood of a class boundary and considers one of the possible effects of estimation errors on class posteriors. Namely, it assumes that the estimated posteriors lead to a boundary between the same two classes, which is just shifted with respect to the ideal one. We already pointed out that this assumption

is reasonable, if a neural network provides good approximations of the posteriors. However this does not allow to analyse other possible effects of estimation errors pointed out in [21], like introducing a boundary in a region where there is none, or missing a boundary. This is perhaps the strongest limitation of the model.
- A more subtle assumption is that each realisation of all the classifiers of the ensemble leads to a boundary between the two given classes in the neighbourhood of a given ideal boundary. This is necessary for the computation of the expected added error to make sense. Moreover, for the same reason the same assumption is made for the linear combination of an ensemble of classifiers. We point out that this assumption was not explicitly stated in works by Tumer and Ghosh, although it is not implied by the one discussed in the previous point. Indeed, it can be easily shown that linearly combining (even by simple average) classifiers which provide a boundary between two classes in a given region of the feature space can lead to obtain completely different decision regions (for instance, more than one boundary can be obtained, or even none).
- The model applies to one-dimensional feature spaces only. The obvious question is: does the results apply also to multi-dimensional feature spaces? An extension of the model to this case was discussed in [30], but it is much more complicated to deal with and requires some additional simplifying assumptions to make it analytically tractable.
- Last but not least, the model does not take into account the overall added error over Bayes error, but only the contribution to the added error due to a subset of the feature space. One could argue that the model can be applied separately to each region around an ideal class boundary (in a one-dimensional feature space). However this requires the underlying assumption that the classifier provides accurate estimates of *all* class boundaries, so that the estimation errors cause just a shift of all the ideal boundaries.

Clearly, the limitations discussed above seem rather strong, although it should be pointed out that they do not imply that the model can not provide accurate predictions when the above assumptions are violated, as suggested by the experiments reported in [8] and in section 4. Let us now summarise, for the sake of completeness, the approximations made in developing the model.

Approximations used in the model

- First order approximation of the posterior probabilities around the ideal boundary x^*, and zero-order approximation of $p(x)$. This approximation is necessary to obtain an expression of the added error which is a second-order polynomial with respect to both the boundary shift b and the estimation errors $\varepsilon_k(x_\mathrm{b})$. This way, the expected added error is a function only of the first and second-order moments (namely, mean and variance) of the distribution of the estimation errors. This is a reasonable approximation, if the boundary shift b is relatively small (namely when the individual classifiers provide good estimates of the class posteriors).
- The estimation errors on different classes, made either by the same classifier or by different classifiers, are uncorrelated (namely, $\varepsilon_i^m(x)$ and $\varepsilon_j^n(x)$, for any m, n and

for any $i \neq j$). This assumption was made just for simplifying computations, but can be considered reasonable to some extent. It should however be noted that it does not hold when the estimated posterior probabilities are constrained to sum up to 1, given that $\sum_k f_k(x) = 1$ implies, from eq. 5, that $\sum_k \varepsilon_k(x) = 0$. Usually this constrain is not enforced in neural networks, but in other kind of classifiers (like parametric classifiers) the posteriors estimates always sum up to 1.

3.4 A Generalisation of the Model by Tumer and Ghosh

As explained in section 3.1, the model by Tumer and Ghosh is based on analysing the added error over Bayes error in a neighbourhood of a given ideal class boundary, assuming that the effect of estimation errors is a shift of such boundary. As pointed out in [31,32] and in section 3.3, this assumption is reasonable if the individual classifiers provide good approximations of the ideal boundary, but in general the estimation errors can cause other effects. The authors developed in [4] a generalisation of this model based on the idea of focusing not on an ideal class boundary, but on a estimated class boundary. More precisely, our aim was to analyse the contribution to the added error over the Bayes error in a subset of the feature space around any given estimated boundary, relaxing the assumption of the presence of an ideal boundary between the same classes in that neighbourhood. This implies that our analysis applies also to the case in which the estimation errors do not provide accurate approximations of the ideal boundaries.

Except from the assumption mentioned above, we used all the other assumptions and approximations in the model by Tumer and Ghosh. To describe our model, let us consider two possible realisations of an estimated boundary x_b between any two classes ω_i and ω_j, as in the example of Fig. 4. Note that in this example there is no ideal boundary between these classes, since their true posteriors do not intersect. Denoting with $\omega(x)$ the class exhibiting the highest true a posteriori probability for the sample x, namely $\omega(x) = \arg\max_{\omega_k} P(\omega_k|x)$, and assuming without loss of generality that $f_i(x_b) > f_j(x_b)$ for $x < x_b$, so that x is assigned to ω_i, if $x < x_b$, the added error in any interval $[x_1, x_2]$ containing the estimated boundary x_b can be written as a function of x_b, as:

$$e_\mathrm{add}(x_b) = \int_{x_1}^{x_b} (P(\omega(x)|x) - P(\omega_i|x))p(x)\mathrm{d}x + \int_{x_b}^{x_2} (P(\omega(x)|x) - P(\omega_j|x))p(x)\mathrm{d}x. \quad (35)$$

It is now convenient to remove the dependence on $P(\omega(x)|x)$. This can be attained by considering any fixed reference point $x_\mathrm{ref} \in [x_1, x_2]$, and by rewriting $e_\mathrm{add}(x_\mathrm{b})$ as $e_\mathrm{add}(x_\mathrm{ref}) + [e_\mathrm{add}(x_\mathrm{b}) - e_\mathrm{add}(x_\mathrm{ref})]$, where $e_\mathrm{add}(x_\mathrm{ref})$ is the added error that one would get if the estimated boundary coincided with the chosen reference point, namely if $x_b = x_\mathrm{ref}$. The difference $[e_\mathrm{add}(x_\mathrm{b}) - e_\mathrm{add}(x_\mathrm{ref})]$, denoted in the following as $\Delta e_\mathrm{add}(x_\mathrm{ref}, x_b)$, can be written as:

$$\Delta e_\mathrm{add}(x_\mathrm{ref}, x_b) = \int_{x_\mathrm{ref}}^{x_b} (P(\omega_j|x) - P(\omega_i|x))\, p(x)\mathrm{d}x. \quad (36)$$

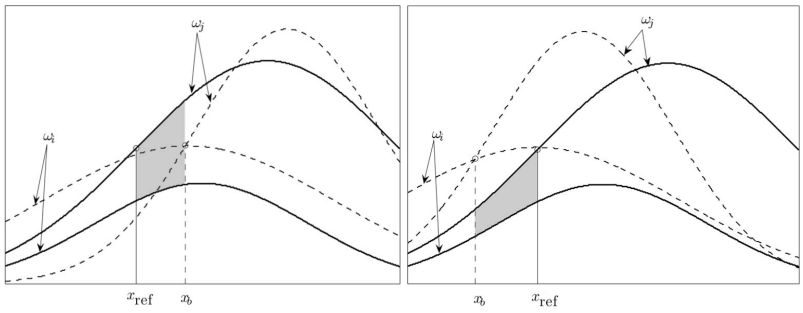

Fig. 4. Two possible realisations of the estimates of the posteriors of classes ω_i and ω_j (dashed lines), leading to an estimated class boundary x_b. The true posteriors are shown as solid lines. The difference $\Delta e(x_{\rm ref}, x_b)$ ($x_{\rm ref}$ is the same in both plots) corresponds to the grey areas: it is positive in the left and negative in the right.

In the example of Fig. 4, the shaded area corresponds to $\Delta e_{\rm add}(x_{\rm ref}, x_b)$. Note now that $\Delta e_{\rm add}(x_{\rm ref}, x_b)$ depends on the posteriors of ω_i or ω_j only, contrary to both $e_{\rm add}(x_{\rm ref})$ and $e_{\rm add}(x_b)$. Now, if we use the same reference point for each individual classifier and for their linear combination, their added error can be written as the sum of $e_{\rm add}(x_{\rm ref})$, which is a constant term *identical* for all classifiers and for the ensemble, and the term $\Delta e_{\rm add}(x_{\rm ref}, x_{\rm b})$, which depends on the position of the estimated boundary. This allows to evaluate the reduction of the added error which can be attained by the linear combination by comparing only the added error difference $\Delta e_{\rm add}(x_{\rm ref}, x_{\rm b})$.

We can now carry out the same computations of Tumer and Ghosh model to derive an expression of the expected added error difference as a function of the estimation errors. We denote with b the offset $x_{\rm b} - x_{\rm ref}$, and make a first-order approximation of the posteriors and a zero-order approximation of $p(x)$ around the reference point $x_{\rm ref}$:

$$P(\omega_k|x) = P(\omega_k|x_{\rm ref}) + b \cdot P'(\omega_k|x_{\rm ref}),$$
$$p(x) = p(x_{\rm ref}).$$

Note that the above approximation is reasonable, if the offset $x_{\rm b} - x_{\rm ref}$ is small, which is an analogous assumption as in Tumer and Ghosh model (in that case the ideal boundary x^* has the role played here by $x_{\rm ref}$). Using this approximation, the added error 36 can be written as:

$$\Delta e_{\rm add}(x_{\rm ref}, x_b) = \int_{x_{\rm ref}}^{x_{\rm ref}+b} \Big[P(\omega_j|x) - P(\omega_i|x)\Big] \cdot p(x)dx$$
$$\simeq p(x_{\rm ref})(u \cdot b + \frac{t}{2}b^2), \tag{37}$$

where $t = P'(\omega_j|x_b) - P'(\omega_i|x_b)$ and $u = P(\omega_j|x_{\rm ref}) - P(\omega_i|x_{\rm ref})$. The expected value of the added error difference, with respect to b, is:

$$\Delta E_{\rm add} = p(x_{\rm ref}) \left[u\beta_b + \frac{t}{2}\beta_b^2 + \frac{t}{2}\sigma_b^2\right] \quad . \tag{38}$$

Let us now derive an expression of b as a function of the estimation errors. The estimated boundary x_{b} is characterised by $f_i(x_b) = f_j(x_b) > f_k(x_b), k \neq i, j$, where the equality can be written as $P(\omega_i|x_b)+\varepsilon_i(x_b) = P(\omega_j|x_b)+\varepsilon_j(x_b)$. Using the first order approximation for the posteriors, such equality becomes:

$$P(\omega_i|x_{\mathrm{ref}}) + b \cdot P'(\omega_i|x_b) + \varepsilon_i(x_b) = P(\omega_j|x_{\mathrm{ref}}) + b \cdot P'(\omega_j|x_b) + \varepsilon_j(x_b),$$

from which it easily follows that:

$$b = -\frac{u}{t} + \frac{\varepsilon_j(x_{\mathrm{b}}) - \varepsilon_i(x_{\mathrm{b}})}{t}. \tag{39}$$

The mean and variance of b which appear in 38 are thus given by:

$$\beta_b = \frac{\beta_i - \beta_j}{t} - \frac{u}{t},\ \sigma_b^2 = \frac{\sigma_i^2 + \sigma_j^2}{t^2}\ . \tag{40}$$

After substituting eq. 40 into eq. 38, one finally obtains the following expression for the expected added error difference of an individual network:

$$\Delta E_{\mathrm{add}} = \frac{p[x_{\mathrm{ref}}]t}{2}\left[-\frac{u^2}{t^2} + \frac{1}{t^2}(\beta_i - \beta_j)^2 + \frac{1}{t^2}(\sigma_i^2 + \sigma_j^2)\right]\ . \tag{41}$$

Note that u^2/t^2 is a constant term which depends only on the choice of the reference point x_{ref}.

The expected added error difference for the linear combiner (considering non-negative weights which sum up to 1 as in section 3.2) can be derived similarly. Omitting the derivation, one obtains:

$$\Delta E_{\mathrm{add}}^{\mathrm{ave}} = \\ \frac{p[x_{\mathrm{ref}}]t}{2}\left\{-\frac{u^2}{t^2} + \frac{1}{t^2}(\beta_i^{\mathrm{ave}} - \beta_j^{\mathrm{ave}})^2 + \frac{1}{t^2}\left[(\sigma_i^{\mathrm{ave}})^2 + (\sigma_j^{\mathrm{ave}})^2\right]\right\}, \tag{42}$$

where

$$\beta_k^{\mathrm{ave}} = \sum_{n=1}^{N} w_n \beta_k^n, \tag{43}$$

and

$$(\sigma_k^{\mathrm{ave}})^2 = \sum_{n=1}^{N} w_n^2 (\sigma_k^n)^2 + \sum_{n=1}^{N} w_n^2 \sum_{m \neq n} \rho_k^{mn} \sigma_k^m \sigma_k^n,\ \ k = i, j\ , \tag{44}$$

where the symbols in the above expressions have the same meaning as in section 3.1.

Before proceeding in the analysis of the above results, we point out that the model by Tumer and Ghosh can be obtained as a particular case of the one described above. It is sufficient to note that, if in a neighbourhood of the estimated boundary x_{b} there is an ideal boundary x^* between the same two classes, then by taking x^* as the reference point one obtains exactly the same expressions for the expected added error as in Tumer and Ghosh model. In particular note that, if $x_{\mathrm{ref}} = x^*$, then $e_{\mathrm{add}}(x_{\mathrm{ref}}) = 0$, and thus

the added error difference $\Delta e_{\mathrm{add}}(x_{\mathrm{ref}}, x_b)$ equals the added error e_{add} of Tumer and Ghosh model.

The overall expected added error in our model is thus given by $e_{\mathrm{add}}(x_{\mathrm{ref}}) + \Delta E_{\mathrm{add}}$, for an individual network, and by $e_{\mathrm{add}}(x_{\mathrm{ref}}) + \Delta E^{\mathrm{ave}}_{\mathrm{add}}$, for the linear combiner. These expressions can be subdivided into the sum of three terms: the first one is a constant term $e_{\mathrm{add}}(x_{\mathrm{ref}}) - \frac{P(x_{\mathrm{ref}})u^2}{2t}$, whose value depends only on the choice of the reference point x_{ref} and is identical for all individual networks and for their linear combination; the second term depends on the biases of estimation errors (40 and 43), and the third one on their variances, as well as the correlations for the linear combiner (40 and 44). It follows that the comparison between the performance of the individual networks and of their linear combination can be carried out taking into account only the bias and variance components of the corresponding expected added errors. Consider now that the expressions of the bias and variance components of the expected added error are identical to the ones derived from Tumer and Ghosh model (see eqs. 9 and 25, respectively for an individual network and for the weighted average), except for the fact that they are computed with respect to the reference point x_{ref} instead of the ideal boundary x^*. Therefore, all the conclusions derived in sections 3.1 and 3.2 from the analysis of Tumer and Ghosh model about the reduction of the bias and variance components attainable by simple averaging a neural network ensemble, and about the comparison between the bias and variance components of the weighted and simple average rules, are valid also for the model described in this section. We summarise here these conclusions:

- The bias component of the expected added error of the simple average rule is between the minimum and the maximum of the bias terms of individual networks: $\min_m \beta_b^m \leq \beta_{b^{\mathrm{sa}}} \leq \max_m \beta_b^m$.
- The variance component of the expected added error of the simple average rule is between 0 and the highest variance component among individual networks: $0 \leq (\sigma^{\mathrm{sa}})^2 \leq \max_m (\sigma^{\mathrm{sa}})^2$. In particular, the variance component attains the maximum value above when all individual networks exhibit identical variances and all their correlations are equal to 1, while it vanishes when the individual networks exhibit the lowest possible (negative) correlation.
- Simple averaging is the best linear combination strategy, if and only if all the individual networks exhibit identical bias and variance components of their expected added error, and identical correlations.
- If the estimation errors of individual networks are unbiased and uncorrelated, then the advantage of the weighted average over the simple average rule (in terms of the difference between the corresponding bias and variance components of the expected added error) depends on the degree of performance imbalance, as explained in section 3.2. In particular, being equal all the other factors, the difference increases as the error range of the ensemble increases, as the performance of the best individual network improves, and as the ensemble size decreases.
- If the estimation errors of individual networks are unbiased but correlated, then the advantage of the weighted average over the simple average rule depends on both the degree of performance and of correlation imbalance, as explained in section 3.2. Being equal all the other factors, the difference increases under the same conditions

of the uncorrelated case, and also as the range of correlation values of the ensemble increases, and as the higher value in such range increases.

In the above discussion we took into account the bias and variance components of the expected added error, which individually take on positive values both in Tumer and Ghosh and in our model. There is however a subtle difference between the two models. Note indeed that in Tumer and Ghosh model the multiplicative factor of the bias and variance components which lead to the overall expected added error is $\frac{p(x^*)t}{2}$, which is always positive. The reason is that the term t is defined as the difference between the first derivative of the posteriors of ω_i and ω_j in x_{b}, $P'(\omega_j|x_b) - P'(\omega_i|x_b)$, which is positive by construction under the assumptions of Tumer and Ghosh model (this can be easily be understood by reasoning on Fig. 1). Instead, in our model the multiplicative factor $\frac{p(x_{\mathrm{ref}})t}{2}$ can also be negative, since $t = P'(\omega_j|x_b) - P'(\omega_i|x_b)$ can be either positive or negative, depending on the behaviour of the two posteriors around x_{ref} (for instance, in the example of Fig. 4 t is positive, but it would be negative, if the first derivative of $P(\omega_i|x)$ on x_b were higher than the one of $P(\omega_j|x_b)$ on the same point). The implication of the above fact is the following. If the estimated boundary x_b lies in a region in which the term t is positive, then the behaviour of the linearly combined ensemble with respect to the individual classifiers, and of the weighted vs. the simple average rule, summarised above, is the same as predicted by Tumer and Ghosh model. Instead, if t is negative, then it is easy to see that some of the conclusions summarised above do not hold anymore. In particular, in this case the reduction of the variance component of the expected added error of individual classifier attained by simple averaging results in an *increase* of the expected added error: the lower the correlation between individual networks, the worse the performance of the simple average rule. Moreover, the optimal weights considered in section 3.2 become the *worst* possible weights, and so the advantage of the weighted average over the simple average rule does not follow the pattern summarised above.

To sum up, on the one hand the model described in this section shows that the behaviour of the linear combining rule predicted by Tumer and Ghosh model can hold also under less strict assumptions. In particular, it can hold even when the main assumption of Tumer and Ghosh model is relaxed, namely when an estimated boundary does not lie in a neighbourhood of an ideal boundary between the same classes. This gives a partial explanation of the experimental results observed in [8] and [4], mentioned in section 3.2 and described in the next section. From a practical viewpoint, this means that the guidelines derived from Tumer and Ghosh could hold even when the underlying assumptions are violated. On the other hand, this model also points out some conditions under which the conclusions drawn from Tumer and Ghosh model are no more valid.

4 Some Experimental Results

In this section we report the results of some experiments carried out with the aim of investigating the behaviour of linear combiners on real data set, in light of the results provided by the analysis of Tumer and Ghosh model and of our model, described in the previous sections. The experimental setting is the same considered in [8]. In particular,

the aim of our experiments was to check whether and to what extent the behaviour of linear combiners on real data set agrees with the predictions derived from these two models. To this aim, it is necessary to take into account that such predictions have been derived under several assumptions (discussed in section 3.3) which do not necessarily hold in real data sets, like a one-dimensional feature space, and involve quantities that are unknown in experiments made on real data sets, like the added error over Bayes error, the contribution of the misclassification probability around a given ideal class boundary, or the bias of estimation errors. For this reason, in the experiments we focused only on quantities that can be estimated, which are the overall misclassification error of a neural network and the correlation between the outputs of different networks (which coincides with the correlation between the estimation errors). More precisely, we checked whether and to what extent the behaviour of the overall misclassification probability of the simple and weighted average rule with respect to the overall misclassification probability of individual networks and on the average correlation between their outputs (which was measured as described below) agrees with the predictions of the models involving the contribution of the expected added error around an ideal or estimated boundary, and the correlation between estimation errors around such boundary. For the reader's ease, we summarise here such predictions in terms of the quantities that can be estimated from real data sets:

- The simple average combiner should perform not worse than the worst individual network.
- Being equal all the other conditions, the advantage of the weighted average over the simple average rule, when the optimal weights are used, increases as the range of misclassification errors of the individual network increases, as the performance of the best individual network improves.
- Similarly, the advantage of the weighted average over the simple average rule, increases as the range of correlation values exhibited by the individual networks increases, and as the maximum value of this range increases.

These predictions involve the use of the optimal weights in the weighted average rule (as explained at the beginning of section 3.2, we are interested in the ideal performance of the weighted average and do not consider the problem of weights estimation), which on real data sets can be found only by some sub-optimal search algorithm. To this aim, we chose a simple exhaustive search over discretised weights values, with a discretisation step of 0.01. To keep the computational complexity acceptable, we considered only an ensemble size of $N = 3$.

Given that the only parameter which can be controlled with some precision on real data set is the overall misclassification probability of individual classifiers, we constructed several ensembles of three classifiers characterised by different ranges of misclassification probabilities and different degrees of performance imbalance. In the following we will denote with E_i the misclassification rate of the i-th individual classifiers, and will order the classifiers of each ensemble such that $E_1 \leq E_2 \leq E_3$. We will refer to the interval $[E_1, E_3]$ as the error range. The misclassification rate of the simple and weighted average combiners will be denoted respectively as E^{sa} and E^{wa}, and the difference $E^{\mathrm{sa}} - E^{\mathrm{wa}}$ as ΔE. We considered 16 ensembles characterised by different

Table 1. Example of misclassification probabilities of the individual classifiers in the twelve unbalanced ensembles considered in the experiments.

Ensemble ID	E_1	E_2	E_3
1	0.05	0.05	0.05
2	0.10	0.10	0.10
3	0.15	0.15	0.15
4	0.20	0.20	0.20
5	0.05	0.10	0.10
6	0.05	0.05	0.10
7	0.10	0.15	0.15
8	0.10	0.10	0.15
9	0.15	0.20	0.20
10	0.15	0.15	0.20
11	0.05	0.15	0.15
12	0.05	0.10	0.15
13	0.05	0.05	0.15
14	0.10	0.20	0.20
15	0.10	0.15	0.20
16	0.10	0.10	0.20

combinations of their performances and different degrees of performance imbalance. Among them, we considered four balanced ensembles (namely, made up of classifiers with identical performances), denoted in the following with numbers 1 to 4. We chose misclassification probabilities with values increasing of 0.05 across these ensembles (for instance, if the misclassification probability of classifiers in ensemble 1 is 0.05, then it is 0.10 in ensemble 2, 0.15 in ensemble 3 and 0.20 in ensemble 4). We considered then 12 unbalanced ensembles characterised by five different error ranges. For a fixed error range, two or three different degrees of performance imbalance were considered by choosing different values of E_2. A possible set of values of E_1, E_2, E_3 for these 12 ensembles is shown in table 1, where each group of rows corresponds to a different error range. Note that unbalanced ensembles 5, 7, 9, 11 and 14 are characterised by $E_2 = E_3$, which corresponds to the condition of maximum performance imbalance (for $N = 3$) according to theoretical results derived in section 3.2, for fixed values of E_1 and E_3. Similarly, ensembles 12 and 15 are characterised by values of E_2 between E_1 and E_3, and should be close to the condition of minimum performance imbalance. In practice, this means that according to the results of section 3.2 we should expect that the improvement ΔE attained by the weighted average rule over the simple average, among ensembles with the same values of E_1 and E_3, is maximum when $E_2 = E_1$ and is minimum when E_2 is between E_1 and E_3.

The experiments have been carried out using multi-layer feed-forward neural networks, with one hidden layer, a number of input units equal to the number of features and a number of output units equal to the number of classes (except for two-class data sets, in which case only one output unit was used). The networks were trained using the standard back-propagation algorithm with fixed learning rate of 0.05, a one-shot coding for the target values (1 for the output unit corresponding to the correct class of a training

Table 2. Data sets used in the experiments, with the size of the training and testing set, the number of features and the number of classes.

Data set	Training set	Test Set	Features	Classes
Optdigits	3823	1797	8	9
Satimage	4435	2000	36	6
Pendigits	7494	3498	16	10
Letter	16000	4000	16	26
Segmentation	210	2100	19	7
Satellite	7939	7848	8	2
DNA	2000	1186	180	3
Feltwell	5124	5829	15	5
Ionosphere	176	175	34	2

sample, 0 for all the other units) and the sum of squared distances to the targets as error measure. The outputs of each network were *not* constrained to sum up to 1.

We point out that the error rates mentioned above are "desired" error rates. To obtain neural networks with error rates close to the desired ones, we trained a large number of networks with a different number of hidden units, different training set sizes and different training sets of the same size (obtained by randomly drawing subsets of the original training set). Moreover, we constructed ten different ensembles for each of the 16 combinations of the desired error rates described above: all the results reported below refer to the average error rates over the ten ensembles.

Besides the error rates, we also computed an estimate of the average correlation between network outputs, over the ten different ensembles with fixed values of E_1, E_2, E_3. The average correlation ρ^{mn} between the outputs of the m-th and the n-th neural network ($m, n = 1, 2, 3,\ m \neq n$) was computed as follows. We first computed the correlation coefficient $\rho^{mn}(x)$ between the outputs $f_k^m(x)$ and $f_k^n(x)$ on each test sample x, for each class k. Then we averaged the $\rho^{mn}(x)$ values over all classes and all test samples.

The experiments were carried out on nine publicly available real data sets. Except for Feltwell, eight of them have been taken from the well known UCI repository [1]. The data sets and their main characteristics are listed in table 2.

In tables 3, 4 and 5 we report the results on three out of the nine data sets, namely Letter, Pendigits and Ionosphere, which are representative of the behaviour observed in the other six data sets.

Considering first the qualitative behaviour of the simple and weighted average rule, the following observations can be made:

- As expected, the weighted average rule always outperformed the simple average, given that the optimal weights were used. Nevertheless, it is worth noting that the SA rule always outperformed the worst classifier of the ensemble.
- With few exceptions, among ensembles with the same error range (namely ensembles (5,6), (7,8), (9,10), (11,12,13) and (14,15,16)), the error rate of the simple and weighted average increases for increasing values of E_2, in agreement with the theoretical predictions.

Table 3. Results on the Letter data set.

#	E_1	E_2	E_3	ρ_{12}	ρ_{13}	ρ_{23}	E_{sa}	E_{wa}	ΔE
1	0.205	0.208	0.205	0.01	0	-0.01	0.185	0.183	0.003
2	0.261	0.26	0.259	0.01	0.01	0	0.231	0.229	0.002
3	0.298	0.301	0.299	0.01	0	0.05	0.271	0.268	0.003
4	0.352	0.345	0.35	-0.01	0.01	0.01	0.307	0.305	0.002
5	0.202	0.261	0.257	0	0	0.01	0.203	0.194	0.008
6	0.206	0.208	0.259	0	-0.01	0.01	0.194	0.189	0.005
7	0.259	0.302	0.297	0.02	0	0	0.25	0.244	0.006
8	0.256	0.259	0.295	0.01	-0.01	-0.01	0.239	0.234	0.004
9	0.3	0.351	0.352	0.01	-0.01	0	0.289	0.282	0.007
10	0.298	0.301	0.355	-0.02	-0.02	-0.04	0.276	0.271	0.005
11	0.201	0.3	0.289	0	0	0.03	0.212	0.197	0.016
12	0.208	0.261	0.292	-0.01	-0.01	-0.01	0.217	0.206	0.01
13	0.207	0.208	0.295	-0.02	-0.01	0.01	0.196	0.189	0.007
14	0.258	0.346	0.348	0	0.01	0	0.263	0.251	0.011
15	0.26	0.304	0.354	0.02	0.01	-0.03	0.261	0.254	0.007
16	0.258	0.26	0.357	0	-0.02	0.01	0.24	0.234	0.006

Table 4. Results on the Pendigits data set.

#	E_1	E_2	E_3	ρ_{12}	ρ_{13}	ρ_{23}	E_{sa}	E_{wa}	ΔE
1	0.091	0.087	0.088	0.01	0.03	0.01	0.082	0.08	0.002
2	0.123	0.121	0.118	-0.03	-0.02	0.02	0.115	0.111	0.004
3	0.174	0.174	0.178	0.14	-0.03	0.02	0.151	0.145	0.006
4	0.217	0.218	0.224	0	0.14	0.04	0.189	0.183	0.005
5	0.086	0.118	0.116	0	-0.02	-0.04	0.095	0.086	0.009
6	0.091	0.089	0.119	-0.01	0	0.04	0.089	0.085	0.005
7	0.115	0.176	0.176	0	0.03	0.01	0.138	0.113	0.025
8	0.118	0.117	0.173	0	0.02	0.01	0.124	0.111	0.013
9	0.177	0.219	0.22	-0.03	0.06	-0.06	0.159	0.156	0.004
10	0.176	0.177	0.219	0.03	0.05	0.02	0.162	0.157	0.005
11	0.088	0.172	0.179	0	0	-0.03	0.115	0.087	0.028
12	0.09	0.117	0.177	0.03	0	0.01	0.106	0.093	0.013
13	0.091	0.088	0.179	-0.02	0.02	0.02	0.092	0.083	0.009
14	0.118	0.218	0.222	0	0.02	0.23	0.142	0.117	0.025
15	0.118	0.178	0.22	0.01	0.03	0.02	0.137	0.116	0.02
16	0.12	0.117	0.222	0.01	0	-0.03	0.12	0.111	0.01

– In the balanced ensembles 1 to 4, the improvement of the weighted average over the simple average (the value ΔE) is often smaller than in the imbalanced ensembles 5 to 16, for the Letter and Pendigits data sets, while there are several exceptions on Ionosphere (as well as in Segmentation, among the other six data sets).
– Inside each of the five groups of ensembles with the same error range, with some exceptions the maximum of ΔE is obtained when $E_2 = E_3$, which corresponds to the condition of maximum performance imbalance.

Table 5. Results on the Ionosphere data set.

#	E_1	E_2	E_3	ρ_{12}	ρ_{13}	ρ_{23}	E_{sa}	E_{wa}	ΔE
1	0.153	0.155	0.155	0.06	0.00	0.00	0.151	0.133	0.018
2	0.193	0.191	0.196	-0.02	0.14	0.10	0.183	0.166	0.017
3	0.253	0.248	0.250	-0.01	0.09	-0.01	0.246	0.223	0.024
4	0.299	0.299	0.300	-0.04	0.01	0.05	0.298	0.273	0.025
5	0.155	0.191	0.196	-0.05	0.00	-0.01	0.171	0.145	0.026
6	0.155	0.155	0.196	0.00	0.09	-0.01	0.160	0.139	0.021
7	0.193	0.250	0.251	0.02	0.02	0.03	0.217	0.176	0.041
8	0.194	0.191	0.253	-0.05	0.03	-0.02	0.193	0.167	0.025
9	0.254	0.301	0.301	0.03	-0.01	-0.14	0.284	0.245	0.040
10	0.249	0.248	0.300	0.07	0.07	0.06	0.256	0.225	0.031
11	0.155	0.251	0.249	-0.01	0.05	-0.04	0.205	0.153	0.051
12	0.155	0.191	0.251	-0.03	-0.07	0.01	0.178	0.148	0.030
13	0.154	0.155	0.251	0.05	0.01	-0.01	0.163	0.141	0.022
14	0.193	0.300	0.300	-0.04	0.18	0.06	0.245	0.187	0.059
15	0.192	0.249	0.300	0.01	0.00	0.05	0.223	0.179	0.044
16	0.194	0.192	0.300	-0.01	0.01	0.06	0.197	0.170	0.028

– Finally, among ensembles exhibiting the same value of E_1 and E_2, $|DeltaE$ increases as E_3 increases, which is again in agreement with the theoretical predictions.

Consider now the quantitative behaviour of the two combining rules:

– As already pointed out, the lower values of ΔE were almost always obtained for the balanced ensembles 1 to 4. These values are almost always below 0.01. The exceptions are the Ionosphere and Segmentation data sets, were values up to 0.025 were observed. Higher values of were obtained for the imbalanced ensembles 5 to 16. In particular, the maximum values of ΔE were obtained for ensembles with the greatest error range width (0.10). However, for all imbalanced ensembles with identical error range, the value of ΔE depends strongly on the value of E_2 , namely on the kind of performance imbalance. This means that the improvement achievable using the weighted average may be small even for ensembles with a large error range width. All these results agree with the theoretical predictions.
– Consider finally the correlation between classier outputs. The ones observed in the experiments are close to 0, due to the fact that the classifiers were trained on randomly drawn training sets. According to the theoretical predictions, for uncorrelated outputs the simple average should reduce the variance component of the expected added error by a factor of N ($N = 3$ in our experiments). The reduction of the overall expected added error could however be lower, given that the bias term is not necessarily reduced. The experimental results show that in fact the performance of the simple average is almost always close to that of the best classifier of the ensemble (sometimes it is even better), even for highly imbalanced ensembles.

To sum up, we can say that these experiments provided evidence that the qualitative behaviour of the two combining rules on real data sets agrees with rather good

accuracy with the predictions of the model by Tumer and Ghosh, despite it is based on strict assumptions as discussed in section 3.3. The most evident violations of the theoretical predictions were observed on the Ionosphere and Segmentation data sets. In particular, even in the balanced ensembles 1 to 4 the advantage of the weighted average rule over the simple average was substantially large. However these results can partly be explained by the fact that, due to the small training set size (one order of magnitude smaller than in the other data sets), the ten different ensembles constructed for each of the 16 combinations of the desired error rates exhibited an average error rate close to the desired one, but with a high variance. This means that each of the ten ensembles was often imbalanced.

5 Discussion and Conclusions

The linear combination is one of the simplest and most used combining strategies in the multiple classifier systems field for classifiers that provide soft outputs, and in particular estimates of the class posterior probabilities, like neural networks.

So far the literature on linear combiners mainly considered two topics related to general issues in the multiple classifier systems field: methods for weight estimation, and the theoretical analysis of the behaviour of linear combiners. In this chapter we provided an overview of the state of the art on linear combiners, focusing on works dealing with the second of the above topics, and in particular on an analytical model originally developed in works by K. Tumer and J. Ghosh and subsequently extended by the authors.

According to the model by Tumer and Ghosh and to the results derived in subsequent works by the authors, the following practical guidelines for the design of linearly combined classifier ensembles can be given:

- Looking at the ensemble performance in terms of the bias-variance trade-off, if the simple average rule is used an effective ensemble design strategy consists in constructing individual classifier with low bias and low (possibly negative) correlation among their outputs. The variance will be reduced by combining.
- With regard to the choice between the simple and the weighted average rules, which mirrors the problem well known in the multiple classifier systems field of the choice between fixed and trained fusion rules, it can be said that the weighted average can be advantageous (provided that a large data set is available for reliable weight estimation), if the individual classifier exhibit significantly different performance and high correlation between their outputs (it should however be pointed out that this applies only to the case of non-negative weights).

The linear combination strategy is perhaps the less difficult to deal with from a theoretical viewpoint, and the one for which the most relevant results have been obtained so far. However, as acknowledged by the MCS research community, developing general theoretical models to study the behaviour of combining strategies and to develop guidelines for their design is still an open issue [21]. We believe indeed that an interesting research direction for future works is the development of a more general framework for the analysis and comparison of different classifier combination strategies, possibly

trying to unify theoretical results like the ones reported in works by Tumer and Ghosh and by the authors, by Kittler et al. [15,18] and by Kuncheva [20].

References

1. Asuncion, A., Newman, D.J.: UCI Machine Learning Repository (2007), http://www.ics.uci.edu/~mlearn/MLRepository.html
2. Battiti, R., Colla, A.M.: Democracy in neural nets: Voting schemes for classification. Neural Networks 7, 691–707 (1994)
3. Benediktsson, J.A., Sveinsson, J.R., Ersoy, O.K., Swain, P.H.: Parallel Consensual Neural Networks. IEEE Transactions on Neural Networks 8(1), 54–64 (1997)
4. Biggio, B., Fumera, G., Roli, F.: Bayesian Analysis of Linear Combiners. In: [11], pp. 292–301
5. Bishop, C.M.: Neural Networks for Pattern Recognition. Clarendon Press, Oxford (1995)
6. Brown, G., Wyatt, J.L., Tino, P.: Managing Diversity in Regression Ensembles. Journal of Machine Learning Research 6, 1621–1650 (2005)
7. Eckhardt, D.E., Lee, L.D.: A theoretical basis for the analysis of multiversion software subject to coincident errors. IEEE Transactions on Software Engineering 11(12), 1511–1517 (1985)
8. Fumera, G., Roli, F.: A Theoretical and Experimental Analysis of Linear Combiners for Multiple Classifier Systems. IEEE Transactions on Pattern Analysis and Machine Intelligence 27, 942–956 (2005)
9. Fumera, G., Roli, F., Serrau, A.: A Theoretical Analysis of Bagging as a Linear Combination of Classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(7), 1293–1299 (2008)
10. Geman, S., Bienenstock, E., Doursat, R.: Neural Networks and the bias/variance dilemma. Neural Computation 4, 1–58 (1992)
11. Haindl, M., Kittler, J., Roli, F. (eds.): MCS 2007. LNCS, vol. 4472. Springer, Heidelberg (2007)
12. Hansen, L.K., Salamon, P.: Neural Network Ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence 12, 993–1001 (1990)
13. Hashem, S.: Optimal Linear Combination of Neural Networks. Neural Networks 10, 599–614 (1997)
14. Hashem, S., Schmeiser, B.: Improving Model Accuracy Using Optimal Linear Combinations of Trained Neural Networks. IEEE Transactions on Neural Networks 6, 792–794 (1995)
15. Kittler, J., Hatef, M., Duin, R.P.W., Matas, J.: On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence 20, 226–239 (1998)
16. Kittler, J., Roli, F. (eds.): MCS 2000. LNCS, vol. 1857. Springer, Heidelberg (2000)
17. Kittler, J., Roli, F. (eds.): MCS 2001. LNCS, vol. 2096. Springer, Heidelberg (2001)
18. Kittler, J., Alkoot, F.M.: Sum versus Vote Fusion in Multiple Classifier Systems. IEEE Transactions on Pattern Analysis and Machine Intelligence 25, 110–115 (2003)
19. Krogh, A., Vedelsby, J.: Neural network ensembles, cross validation, and active learning. In: Advances in Neural Information Processing Systems, pp. 231–238. MIT Press, Cambridge (1995)
20. Kuncheva, L.I.: A theoretical study on six classifier fusion strategies. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 281–286 (2002)
21. Kuncheva, L.I.: Combining Pattern Classifiers: Methods and Algorithms. Wiley, Hoboken (2004)

22. Liu, Y.: Negative Correlation Learning and Evolutionary Neural Network Ensembles. PhD thesis, University College, The University of New South Wales, Australian Defence Force Academy, Canberra, Australia (1998)
23. Oza, N.C., Polikar, R., Kittler, J., Roli, F. (eds.): MCS 2005. LNCS, vol. 3541. Springer, Heidelberg (2005)
24. Perrone, M.P., Cooper, L.N.: When Networks Disagree: Ensemble Methods for Hybrid Neural Networks. In: Mammone, R.J. (ed.) Neural Networks for Speech and Vision, pp. 126–142. Chapman-Hall, New York (1993)
25. Rogova, G.: Combining the results of several neural network classifiers. Neural Networks 7, 777–781 (1994)
26. Roli, F., Kittler, J. (eds.): MCS 2002. LNCS, vol. 2364. Springer, Heidelberg (2002)
27. Roli, F., Kittler, J., Windeatt, T. (eds.): MCS 2004. LNCS, vol. 3077. Springer, Heidelberg (2004)
28. Sharkey, A.J.C. (ed.): Combining Artificial Neural Nets. Springer, London (1999)
29. Tresp, V., Taniguchi, M.: Combining estimators using non-constant weighting functions. In: Tesauro, G., Touretzky, D.S., Leen, T.K. (eds.) Advances in Neural Information Processing Systems. MIT Press, Cambridge (1995)
30. Tumer, K.: Linear and order statistics combiners for reliable pattern classification. PhD thesis, The University of Texas, Austin (1996)
31. Tumer, K., Ghosh, J.: Analysis of Decision Boundaries in Linearly Combined Neural Classifiers. Pattern Recognition 29, 341–348 (1996)
32. Tumer, K., Ghosh, J.: Linear and order statistics combiners for pattern classification. In: [28], pp. 127–155
33. Ueda, N.: Optimal Linear Combination of Neural Networks for Improving Classification Performance. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 207–215 (2000)
34. Verikas, A., Lipnickas, A., Malmqvist, K., Bacauskiene, M., Gelzinis, A.: Soft Combination of Neural Classifiers: A Comparative Study. Pattern Recognition Letters 20, 429–444 (1999)
35. Wolpert, D.H.: Stacked generalization. Neural Networks 5, 241–259 (1992)
36. Windeatt, T., Roli, F. (eds.): MCS 2003. LNCS, vol. 2709. Springer, Heidelberg (2003)
37. Zanda, M., Brown, G., Fumera, G., Roli, F.: Ensemble Learning in Linearly Combined Classifiers Via Negative Correlation. In: [11], pp. 440–449