Pattern Recognition 46.5 (2013): 1323 – 1336

DOI: http://dx.doi.org/10.1016/j.patcog.2012.10.021

How large should ensembles of classifiers be?

Daniel Hernández-Lobato[a,*], Gonzalo Martínez-Muñoz[a], Alberto Suárez[a]

[a] *Computer Science Department, Escuela Politécnica Superior, Universidad Autónoma de Madrid, C/ Francisco Tomás y Valiente, 11, Madrid 28049 Spain.*

Abstract

We propose to determine the size of a parallel ensemble by estimating the minimum number of classifiers that are required to obtain stable aggregate predictions. Assuming that majority voting is used, a statistical description of the convergence of the ensemble prediction to its asymptotic (infinite size) limit is given. The analysis of the voting process shows that for most test instances the ensemble prediction stabilizes after only a few classifiers are polled. By contrast, a small but non-negligible fraction of these instances require large numbers of classifier queries to reach stable predictions. Specifically, the fraction of instances whose stable predictions require more than T classifiers for $T \gg 1$ has a universal form and is proportional to $T^{-1/2}$. The ensemble size is determined as the minimum number of classifiers that are needed to estimate the infinite ensemble prediction at an average confidence level α, close to one. This approach differs from previous proposals, which are based on determining the size for which the prediction error (not the predictions themselves) stabilizes. In particular, it does not require estimates of the generalization performance of the ensemble, which can be unreliable. It has general validity because it is based solely on the statistical description of the convergence of majority voting to its asymptotic limit. Extensive experiments using representative parallel ensembles (bagging and random forest) illustrate the application of the proposed framework in a wide range of classification problems. These experiments show that the optimal ensemble size is very sensitive to the particular classification problem considered.

Keywords:
Ensemble Learning, Bagging, Random Forest, Asymptotic Ensemble Prediction, Ensemble Size.

1. Introduction

The use of ensembles in classification has been the object of numerous investigations in the machine learning literature [1, 2, 3, 4, 5, 6, 7]. These studies show that combining the decisions of *complementary* classifiers is an effective mechanism to improve the generalization performance of a single predictor. Furthermore, there is extensive empirical

*Corresponding author. Tel: +34-497-2200; fax: +34-497-2235.
Email addresses: daniel.hernandez@uam.es (Daniel Hernández-Lobato), gonzalo.martinez@uam.es (Gonzalo Martínez-Muñoz), alberto.suarez@uam.es (Alberto Suárez)

evidence that the generalization error of parallel ensembles decreases monotonically as the size of the ensemble increases [8, 9, 5]. However, the gains that can be achieved by incorporating additional classifiers become progressively smaller as the ensemble grows. Therefore, it is reasonable to stop aggregating classifiers when the probability of changes in the ensemble output that would result from considering additional predictions is sufficiently small. Determining an appropriate size for the ensemble requires balancing accuracy and efficiency: If, on the one hand, its size is too small, the aggregate ensemble will have poor prediction accuracy. On the other hand, if the ensemble size is too large, we would be wasting memory resources and slowing down the prediction process, which could be a serious disadvantage for online applications.

The main contribution of this work is to take advantage of the convergence properties of majority voting to determine the appropriate size of parallel ensembles composed of classifiers of the same kind. The statistical description of the evolution of the class prediction by majority voting in these types of ensembles has been extensively analyzed in the literature [10, 11, 12, 13, 14, 15, 16, 17]. Given an individual test example, we compute the probability that the class label predicted by an ensemble of classifiers will not change by performing further queries (i.e. by increasing the size of the ensemble). Our analysis shows that for most test instances only a small number of queries are needed to obtain class label predictions that coincide with a high confidence with the predictions given by an ensemble of infinite size. By contrast, a small number of these instances require polling exceedingly large numbers of classifiers to reach a stable prediction. Therefore, it is not possible to determine a fixed size for the ensemble so that the finite ensemble prediction coincides with the asymptotic (infinite ensemble) prediction for *every* test instance with a specified confidence level. Instead of enforcing convergence guarantees for every test instance, we propose to determine the ensemble size by requiring that *on average* the predictions coincide with the infinite ensemble classification with a probability larger than α. The value of α should be close to 1 (e.g. $\alpha = 99\%$). It is specified by the user, depending on the desired level of confidence in the stability of the predictions. We show, both theoretically and empirically, that some properties of the ensemble prediction exhibit universal behavior as α approaches 1 or, alternatively, as the ensemble becomes large ($T \to \infty$). Specifically, the size of the ensemble increases monotonically and becomes arbitrarily large as α approaches 1. Similarly, the fraction of instances for which the prediction of an ensemble of size T differs from the prediction of the ensemble of infinite size decreases as $\propto T^{-1/2}$ when $T \to \infty$. For binary classification problems the proportionality constant can be expressed in terms of the density of test instances for which the prediction probability by a single ensemble member is equal for both classes.

In summary, the observations that (i) the prediction accuracy of parallel ensembles generally improves with the size of the ensemble [9, 3, 5] and (ii) these improvements become smaller for larger ensembles, suggest that it should be possible to reach the accuracy of the infinite-size ensemble with a finite, though possibly large, ensemble. The size of this ensemble is determined as the minimum number of classifiers required for the finite ensemble prediction to coincide with the asymptotic one with the specified confidence level α. The differences between the prediction errors of the finite and the infinite ensemble are bounded from above by $1 - \alpha$, which is the probability that the predictions of the finite and of the infinite ensembles differ. In practice, the differences between the finite ensemble and the infinite ensemble predictions occur with approximately equal frequency in correctly and in incorrectly classified instances. Therefore, the differences in

accuracy between the finite and the infinite ensembles are generally much smaller than this bound.

The analysis developed is valid for any classification task and for ensembles composed of any type of base learners: decision trees, decision stumps, neural networks, SVM's, etc. The only assumption is that the classifiers that make up the ensemble are generated in independent applications of a randomized learning algorithm on the same training data, and that the final prediction is made using majority voting. Examples of ensembles of these types are bagging [1], variants of bagging [4, 18, 19], random forest [3], class-switching ensembles [20, 5], rotation forest [6] and extra-trees [21].

The rest of the manuscript is organized as follows: In Section 2 we analyze the evolution of the class prediction by majority voting as the number of classifiers in the ensemble increases. This analysis is used to determine when a sufficient number of classifiers have been included in the ensemble. Section 3 discusses the relation of the present research with previous methods that have been used to estimate the ensemble size. In Section 4 the results of experiments in a wide range of classification problems are used to illustrate the validity of the proposed framework for the estimation of the ensemble size. Finally, the results and conclusions of this investigation are summarized in Section 5.

2. Estimation of the Optimal Ensemble Size

Consider a binary classification problem, in which $\mathcal{Y} = \{y_1, y_2\}$ is the set of possible class labels. Let $\{h_i(\cdot)\}_{i=1}^{T}$ be an ensemble of classifiers of size T. Assuming that simple majority voting is used to combine the decisions of the individual classifiers, the global ensemble prediction for a given unlabeled instance $\mathbf{x}$ is

$$\hat{y}^T = \arg\max_{y} \sum_{i=1}^{T} \mathcal{I}(h_i(\mathbf{x}) = y),\ y \in \mathcal{Y}\,, \tag{1}$$

where $\mathcal{I}$ is an indicator function.

Parallel ensembles such as bagging and random forest are composed of classifiers generated independently when conditioned to the available training data. More precisely, these classifiers are built in independent executions of the same randomized learning algorithm applied on the observed training data. Therefore, their predictions on a particular instance $\mathbf{x}$ are independent random variables, when conditioned to the training data. This means that the probability distribution of the predictions for a fixed $\mathbf{x}$ of two random ensemble classifiers $h'(\mathbf{x})$ and $h''(\mathbf{x})$, generated on independent applications of a randomized learning algorithm, factorizes

$$\mathcal{P}(h'(\mathbf{x}) = y', h''(\mathbf{x}) = y'') = \mathcal{P}(h'(\mathbf{x}) = y')\mathcal{P}(h''(\mathbf{x}) = y'')\,, \tag{2}$$

where y' and y'' are any class labels from the set $\mathcal{Y}$. An empirical illustration of this independence relation can be found in [22]. Note, however, that independence between the individual predictions (2) does not in general imply that the prediction errors of the

two classifiers are independent

$$\int d\mathbf{x}dy\mathcal{P}(\mathbf{x},y)\mathcal{P}(h'(\mathbf{x})\neq y, h''(\mathbf{x})\neq y)$$
$$= \int d\mathbf{x}dy\mathcal{P}(\mathbf{x},y)\mathcal{P}(h'(\mathbf{x})\neq y)\mathcal{P}(h''(\mathbf{x})\neq y)$$
$$\neq \left[\int d\mathbf{x}dy\mathcal{P}(\mathbf{x},y)\mathcal{P}(h'(\mathbf{x})\neq y)\right]\left[\int d\mathbf{x}dy\mathcal{P}(\mathbf{x},y)\mathcal{P}(h''(\mathbf{x})\neq y)\right]. \quad (3)$$

For parallel randomized ensembles, because of the independence of the individual predictions, the polling process defined by (1) is a sequence of T independent trials [10, 16, 17]. The outcome of each trial is in the set $\mathcal{Y}$. In binary classification problems, the distribution of class votes for the test instance $\mathbf{x}$ in an ensemble of size T is binomial

$$\mathcal{P}(\mathbf{t}|T,\boldsymbol{\pi}(\mathbf{x})) = \frac{T!}{t_1!t_2!}\pi_1(\mathbf{x})^{t_1}\pi_2(\mathbf{x})^{t_2}\,, \quad (4)$$

where t_i is the number of classifiers that predict class label y_i, $i = 1, 2$. In terms of the vector of votes $\boldsymbol{t} = \{t_1, t_2;\, t_1 + t_2 = T\}$, the class predicted by the ensemble of size T is

$$\hat{y}^T = \arg\max_i \{t_i; i = 1, 2\}\,, \quad (5)$$

and $\boldsymbol{\pi}(\mathbf{x})$ is the probability vector

$$\boldsymbol{\pi}(\mathbf{x}) = \{\pi_1(\mathbf{x}), \pi_2(\mathbf{x})\}, \quad \pi_1(\mathbf{x}) + \pi_2(\mathbf{x}) = 1, \quad (6)$$

where $\pi_i(\mathbf{x})$ is the probability that an individual classifier of the ensemble assigns class label y_i to the instance characterized by the vector of attributes $\mathbf{x}$. The values of these probabilities are in general unknown. They depend on the algorithm used to build the base learners, on the particular classification problem and on $\mathbf{x}$, the instance considered. To simplify the notation, the dependence on $\mathbf{x}$ of the probability vector $\boldsymbol{\pi}$ is assumed to be implicit in the remainder of the article.

Assuming that $\boldsymbol{\pi}$ is known, the probability that an ensemble of size T assigns class label y_i to instance $\mathbf{x}$ is the sum of (4) over all the ensemble predictions in which class y_i receives more votes than the other class. In particular, for class y_1

$$\mathcal{P}(\hat{y}^T = y_1|T,\pi_1) = \sum_{\mathbf{t};t_1>t_2} \mathcal{P}(\mathbf{t}|T,\boldsymbol{\pi})$$
$$= \sum_{t_1=\lceil\frac{T}{2}\rceil}^{T} \binom{T}{t_1}\pi_1^{t_1}(1-\pi_1)^{T-t_1} = I_{\pi_1}\left(\lfloor\frac{T}{2}\rfloor + 1, T - \lfloor\frac{T}{2}\rfloor\right), \quad (7)$$

where $I_x(a, b)$ is the regularized incomplete beta function [23].

Figure 1 (left) displays the dependence of the probability that the ensemble predicts class y_1 (7) as a function of π_1, the probability that an individual classifier predicts class y_1, for different values of the ensemble size T. Note that for $T = 1$, (7) is simply the identity function. As T grows, (7) asymptotically approaches a step function.

$$\lim_{T\to\infty} \mathcal{P}(\hat{y}^T = y_1|T,\pi_1) = \begin{cases} 1 & \text{if } \pi_1 > 1/2, \\ 1/2 & \text{if } \pi_1 = 1/2, \\ 0 & \text{if } \pi_1 < 1/2. \end{cases} \quad (8)$$

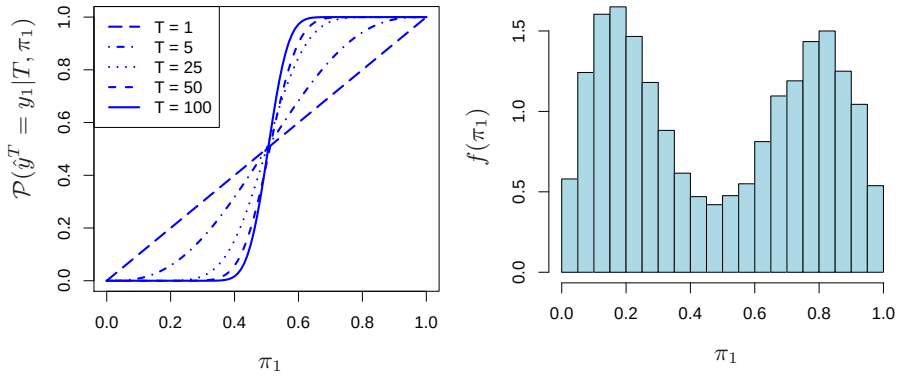

Figure 1: (left) Probability that an individual ensemble classifier predicts class y_1 as a function of π_1 for different values of T. (right) Histogram of $10,000$ samples from the probability distribution of π_1, denoted $f(\pi_1)$, for the *Twonorm* binary classification problem. The estimates are obtained using a random forest (RF) composed of $10,000$ trees. This ensemble is built on a training set composed of 300 labeled instances.

The right-hand side of Figure 1 displays a histogram of $10,000$ samples from the probability density function $f(\pi_1)$ for the *Twonorm* classification problem [24]. The estimations are performed using random forest (RF) [3] of $10,000$ trees. The individual decision trees are built using a training set composed of 300 instances. The estimations of π_1 are made on an independent test set of $10,000$ instances. For each test instance the value of π_1 is estimated as the fraction of classifiers that predict class label y_1. The probability density function estimated is bimodal. This means for some instances the classifiers tend to predict class y_1 more often and for other instances the prediction y_2 is more frequent. This bimodality should be expected, because the training set is composed of instances of both classes in approximately equal numbers. For instances located near the decision boundary, approximately one half of the predictions are class y_1 and the other half are class y_2. The values of π_1 for these instances are in the vicinity of $1/2$. This means that the uncertainty of the individual predictions is rather large. In consequence, for these instances, more classifiers need to be queried to converge to a stable ensemble prediction.

It is important that the estimations of $f(\pi_1)$ be made on a set that is independent of the training data: the training set estimate of the probability density, $f_{\text{train}}(\pi_1)$, is generally biased because the classifiers tend to agree more frequently on these instances. Therefore, the fraction of training instances whose probability π_1 is close to $1/2$ is expected to be smaller than the corresponding fraction for an independent test set. Furthermore, the modes of $f_{\text{train}}(\pi_1)$ are closer to the extreme values $\pi_1 = 0$ and $\pi_1 = 1$ than the corresponding modes in an independent test set. As a result of this bias, the size of the ensemble required to obtain stable predictions is generally smaller for the training set than for an independent test set.

2.1. Analysis of the ensemble prediction for an individual test instance

There is extensive evidence that the generalization error of parallel ensembles decreases monotonically as the size of the ensemble increases [9, 3, 5]. In practice, it is not possible to query an infinite number of classifiers to obtain the asymptotic ensemble

prediction. Nevertheless, assuming that the value of π_1 for the instance to be classified is known, or can be estimated in some way, the probability that an ensemble of size T assigns the same class label as the infinite ensemble is

$$\mathcal{P}(\hat{y}^T = \hat{y}^\infty | T, \pi_1) = I_{\max\{\pi_1, 1-\pi_1\}} \left(\lfloor \frac{T}{2} \rfloor + 1, T - \lfloor \frac{T}{2} \rfloor \right) . \tag{9}$$

Using this expression we can compute $T^\star(\alpha, \pi_1)$, the minimum ensemble size whose prediction for the instance characterized by the probability π_1 coincides with the infinite ensemble prediction with a confidence level α, by finding the smallest value of T that satisfies the inequality

$$\alpha \leq I_{\max\{\pi_1, 1-\pi_1\}} \left(\lfloor \frac{T}{2} \rfloor + 1, T - \lfloor \frac{T}{2} \rfloor \right) . \tag{10}$$

It is not possible to derive an explicit exact formula for $T^\star(\alpha, \pi_1)$. Nevertheless, this quantity can be readily calculated using numerical algorithms. If only odd values of T are considered (to avoid ties in the ensemble prediction) the right-hand side of (10) grows monotonically with T. Therefore, a simple binary search can be used to compute $T^\star(\alpha, \pi_1)$, given α and π_1.

For values of π_1 close to $1/2$ a closed-form approximation for $T^\star(\alpha, \pi_1)$ can be obtained. In this limit $T^\star(\alpha, \pi_1)$ is large. Therefore, the binomial distribution in (7) can be approximated by a Gaussian distribution with the same mean and variance

$$\mathcal{P}(\hat{y}^T = \hat{y}^\infty | T, \pi_1) \approx \Phi \left(\frac{T \max\{\pi_1, 1-\pi_1\} - T/2}{\sqrt{T \pi_1 (1-\pi_1)}} \right) , \tag{11}$$

where $\Phi(\cdot)$ is the cumulative distribution function of a standard Gaussian distribution. With this approximation (10) becomes

$$T^\star(\alpha, \pi_1) \approx \frac{\Phi^{-1}(\alpha)^2 \pi_1 (1-\pi_1)}{(\pi_1 - 1/2)^2}, \qquad \pi_1 \approx 1/2 , \tag{12}$$

where $\Phi^{-1}(\cdot)$ is the quantile function of a standard Gaussian distribution. For a fixed value of α, expression (12) shows that $T^\star(\alpha, \pi_1)$ becomes infinite in the limit $\pi_1 \to 1/2$. Therefore, the limiting factor that determines the ensemble size is the presence of instances for which π_1 is close to $1/2$. For these instances the ensemble decision is uncertain and a very large number of classifiers needs to be queried to produce a reliable estimate of the infinite ensemble prediction.

Since different examples have different values of π_1, this quantity can be modeled as a random variable whose probability density function is $f(\pi_1)$ (see the right-hand-side of Figure 1). $T^\star(\alpha, \pi_1)$ is also a random variable because it depends on π_1. Let $\mathcal{P}(T^\star(\alpha, \pi_1) > T)$ be the probability that the minimum number of queries required for convergence of the ensemble prediction is above threshold T when π_1 follows the distribution $f(\pi_1)$. In the limit $T \to \infty$, this probability can be approximated as

$$\mathcal{P}(T^\star(\alpha, \pi_1) > T) \approx \frac{f(\frac{1}{2}) \Phi^{-1}(\alpha)}{\sqrt{T}} . \tag{13}$$

assuming that the density function of π_1 evaluated at $\pi_1 = 1/2$ is positive $f(\pi_1 = 1/2) > 0$. The details of the derivation are given in Appendix A.

This is an important result showing that, for a fixed value of α, the asymptotic decay of the probability is algebraic with a universal behavior $\propto T^{-1/2}$. The only dependence on the classification problem considered and on the ensemble method used is via the proportionality constant $f(\pi_1 = 1/2) > 0$. The heavy-tailed form of $\mathcal{P}(T^\star(\alpha, \pi_1) > T)$ implies that the fraction of instances that require a very large number of classifiers to converge to the asymptotic prediction with a level of confidence α is significant.

Figure 2 illustrates this phenomenon. The left-hand side of this figure displays the histogram for the minimum number of classifiers required to estimate the asymptotic class label of test instances with a confidence level of at least $\alpha = 99\%$ for the *Twonorm* problem. The distribution of the probabilities π_1 is estimated under the same conditions as the experiments whose results are displayed in Figure 1 (right). The histogram shows that the right tail of the distribution has a very slow decay. The origin of this heavy-tailedness is the presence of instances close to the decision border ($\pi_1 \approx 1/2$), whose stable prediction requires querying a large number of classifiers. The right-hand side of this figure displays, in double logarithmic axes, an empirical estimate of $\mathcal{P}(T^\star(\alpha, \pi_1) > T)$ and the asymptotic approximation of $\mathcal{P}(T^\star(\alpha, \pi_1) > T)$ when $T \to \infty$ given by (13). As expected, the predicted dependence is very close to the empirical estimate for large T.

It is also possible to derive an asymptotic approximation for the probability that the infinite ensemble prediction differs from the prediction of an ensemble of size T, for sufficiently large T

$$\mathcal{P}(\hat{y}^T \neq \hat{y}^\infty | T) \approx \frac{f(\frac{1}{2}) \int_{-\infty}^{0} \Phi(z) dz}{\sqrt{T}} \quad T \to \infty \,, \tag{14}$$

assuming $f(\frac{1}{2}) > 0$; that is, a non-vanishing probability density for instances whose classification is uncertain ($\pi_1 = 1/2$). The details of the derivation are given in Appendix B. The asymptotic algebraic decay of this probability $\propto T^{-1/2}$ is also universal. The only dependence on the classification problem considered and on the ensemble method used is through $f(\pi_1 = 1/2) > 0$.

In summary, most of the data instances require querying a fairly small number of classifiers to produce an estimate of the asymptotic prediction with a high confidence. By contrast, a small but not negligible fraction of test instances require querying extremely large numbers of classifiers for the ensemble predictions to stabilize. The fraction of instances whose stable predictions require at least T classifiers has a universal form proportional to $T^{-1/2}$ as T becomes large. The convergence of the ensemble prediction to its asymptotic limit is dominated by these borderline instances. In consequence, it is not possible to choose a finite ensemble size T so that the asymptotic prediction is reached for every test instance with a specified level of confidence $\alpha > 0$. This is a general result that applies to any binary classification problem and any ensemble learning algorithm provided that (i) the individual classifiers are built independently when conditioned to the training data; (ii) majority voting is used to combine the outputs of the ensemble classifiers; and (iii) $f(1/2) > 0$.

2.2. Ensemble Size

From the analysis presented one concludes that it is not possible to give convergence guarantees for the ensemble prediction in every instance. As an alternative, we propose

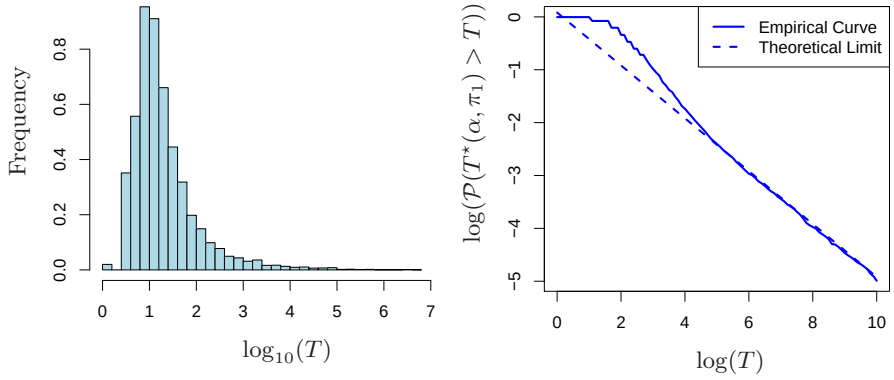

Figure 2: (left) Histogram for the values of $T^\star(\alpha, \pi_1)$ for the classification problem *Twonorm* and $\alpha = 99\%$. (right) Empirical and theoretical estimations of the distribution $\mathcal{P}(T^\star(\alpha, \pi_1) > T)$ as $T \to \infty$ displayed in double logarithmic axes. The slope of the straight line is $-1/2$.

to determine the size of the ensemble by requiring that the *average* confidence in the asymptotic prediction be larger or equal to α. With this less restrictive condition, the ensemble size $T^\star(\alpha)$ is the minimum value of T that satisfies

$$\begin{aligned} \alpha &\leq \mathcal{P}(\hat{y}^T = \hat{y}^\infty | T) \\ &= \int_0^1 \mathcal{P}(\hat{y}^T = \hat{y}^\infty | \pi_1, T) f(\pi_1) d\pi_1 \\ &= \int_0^1 I_{\max\{\pi_1, 1-\pi_1\}} \left(\lfloor \frac{T}{2} \rfloor + 1, T - \lfloor \frac{T}{2} \rfloor \right) f(\pi_1) d\pi_1 \,. \end{aligned} \tag{15}$$

For values of α close to one, the value of $T^\star(\alpha)$ is approximately

$$T^\star(\alpha) \approx \left(\frac{f(\frac{1}{2}) \int_{-\infty}^{0} \Phi(z) dz}{1 - \alpha} \right)^2 , \tag{16}$$

where we have used that $1 - \alpha \approx \mathcal{P}(\hat{y}^T \neq \hat{y}^\infty | T = T^\star(\alpha))$. This result shows that $T^\star(\alpha) \to \infty$ when $\alpha \to 1$, as expected. The ensemble size is proportional to the square of $f(\pi_1 = 1/2)$, the density of test instances that are close to the decision boundary. The more frequent these borderline instances are, the larger the ensembles required to obtain stable predictions. Because of the large variability of this density, the ensemble sizes $T^\star(\alpha)$ can be very different in different classification tasks.

To compute $T^\star(\alpha)$ we use (15) and an estimate of $f(\pi_1)$. This estimate can be obtained from a validation set, by cross-validation, using out-of-bag data [25], or, since the class labels are not required for this purpose, using the test set. The complete training data should not be used because, as discussed earlier, it yields biased estimates of $f(\pi_1)$. Given a set of N instances (either out-of-bag data, a validation set or the test set), the integral in (15) is approximated by Monte Carlo

$$\alpha \leq \frac{1}{N} \sum_{i=1}^{N} I_{\max\{\hat{\pi}_1^{(i)}, 1-\hat{\pi}_1^{(i)}\}} \left(\lfloor \frac{T}{2} \rfloor + 1, T - \lfloor \frac{T}{2} \rfloor \right) , \tag{17}$$

Input: Confidence level α and training data $\mathcal{D}$

1. size $\leftarrow 100$
2. $E \leftarrow$ buildEnsemble(size, $\mathcal{D}$)
3. **Repeat**
   (a) $T^\star(\alpha) \leftarrow$ estimateEnsembleSize by Eq. (17)
   (b) **if** $T^\star(\alpha) <$ size
      i. **break**
   (c) **else**
      i. size $\leftarrow \min(T^\star(\alpha), 2 \cdot \text{size})$
      ii. $E \leftarrow E \cup$ buildEnsemble(size $- |E|, \mathcal{D}$)
   (d) **End if**
4. **End Repeat**
5. $E \leftarrow$ truncateEnsemble$(E, T^\star(\alpha))$

Output: Ensemble E of size $|E| = T^\star(\alpha)$.

Figure 3: Pseudo-code for the estimation of the optimal ensemble size $T^\star(\alpha)$.

where $\{\hat{\pi}_1^{(i)}\}_{i=1}^N$ are the estimates of π_1 for each of the N instances. Since (17) grows monotonically with increasing values of T, assuming T odd, binary search can be used to find $T^\star(\alpha)$. Note that it is also possible to use (16) to estimate $T^\star(\alpha)$ when α is close to one. Nevertheless, this requires an estimate of $f(\frac{1}{2})^2$ which can be unreliable as a consequence of the variance of the estimator.

Figure 3 displays the pseudo-code of an algorithm that can be used to determine the optimal ensemble size $T^\star(\alpha)$. The starting point is an ensemble of size 100, which is a typical value used in the literature on classification ensembles. This ensemble is used to compute an initial estimate of $T^\star(\alpha)$ using (17). If the value of $T^\star(\alpha)$ is smaller than or equal to the current size of the ensemble, the algorithm stops and the ensemble is pruned by retaining only $T^\star(\alpha)$ classifiers. Otherwise, new classifiers are incorporated in the ensemble. The number of additional classifiers incorporated is the minimum of $T^\star(\alpha)$ and twice the size of the current ensemble. The process is repeated until the value estimated for $T^\star(\alpha)$ is smaller or equal than the size of the current ensemble. The rule used to update the size of ensemble (step 3-(c)-i in the algorithm displayed in Figure 3) provides a good balance between the number of times that $T^\star(\alpha)$ has to be evaluated and the number of final classifiers that have to be discarded due to the truncation step. In all the cases investigated the algorithm converges after a few (typically no more than 5) iterations.

3. Related Work

The analysis presented in Section 2 relies exclusively on the convergence properties of the class prediction by majority voting as the number of classifiers in the ensemble increases [10, 11, 12, 13, 14, 15, 16, 17]. A similar analysis has been applied in [10] to describe the evolution of the ensemble generalization error a as a function of its size. The infinite ensemble limit is not considered explicitly by these authors. Nevertheless,

expressions that are valid in this limit can be readily obtained from their results. The statistical description of majority voting was used also in [16] to address inference on the prediction of an ensemble of finite size based on the predictions of only a fraction of the ensemble classifiers. In contrast to the current research, that work assumes that the original prediction ensemble is given. Therefore, the results of [16] cannot be used to determine the appropriate size of the initial ensemble. Assuming a uniform prior distribution for π_1, the analysis shows that it is possible to determine, for each instance to be classified, the fraction of ensemble classifiers that need to be queried to estimate the prediction of the complete ensemble with a confidence level above a specified threshold. This fraction depends strongly on the particular test instance that is being processed. Therefore, for a particular instance, the querying of the ensemble classifiers can be halted when the probability that the current majority class would change because of the remaining votes is sufficiently low. This dynamical (instance-based) ensemble pruning method leads to a significant speed-up of the classification process with only a small deterioration in the accuracy of the predictions. Notwithstanding, all the ensemble classifiers need to be retained in memory and be available to resolve potential queries.

The dynamic pruning technique described in [16] has been extended to make inference on the prediction of ensembles of infinite size in [17]. These authors propose to halt the voting process for a particular test instance when the probability that the current majority class coincides with the prediction of a hypothetical ensemble of infinite size is above a user-specified confidence level. Given an initial ensemble of fixed size, one finds that for some test instances it is not possible to estimate the prediction of an ensemble of infinite size with the specified confidence, even after querying all the classifiers in the ensemble. Thus, in practice, the guarantees on the convergence of the predictions can only be made for a fraction of the test instances. For the remaining test instances (typically small, but not negligible fraction), the stability of the predictions is uncertain because all the classifiers contained in the ensemble are queried without reaching the specified confidence level.

The two instance-based pruning techniques described in the previous paragraphs are hence only useful to reduce the number of classifiers that need to be queried for prediction, given an initial ensemble of a fixed size. They cannot be used to estimate an appropriate size for the ensemble, which is the objective of the present investigation. In the current manuscript we analyze the asymptotic behavior of the prediction of parallel ensembles as their size approaches infinity. In particular, we show that the fraction of instances whose stable predictions require more than T classifiers for $T \gg 1$ has a universal form and is proportional to $T^{-1/2}$. This is an important observation that needs to be taken into account in the design of effective ensembles. The results of this analysis are then used to determine an appropriate ensemble size by requiring that, on average, the ensemble predictions have converged to the asymptotic (infinite-size) ensemble with a specified level of confidence. In most previous approaches to this problem the suitable ensemble size is determined by aggregating classifiers until an estimate of the generalization error stabilizes [26, 7, 27]. Methods based on the convergence of the prediction error require the design of reliable estimators of the generalization performance and could be affected by the variance of the estimations and also by overfitting. Our approach differs from these in that it does not require an estimate of the generalization error: To determine the probability that the ensemble prediction has converged to the infinite-size limit it is not necessary to use the actual class labels. Convergence of the predictions of the

ensemble is a sufficient condition for convergence of the generalization error. However, it is not a necessary condition: the generalization error generally stabilizes earlier than the predictions themselves.

One of the earlier proposals to determine the ensemble size is based on estimating minimum number of classifiers which are needed to obtain a prediction accuracy similar to a larger ensemble [26]. In that work, the McNemar non-parametric test is used to determine whether the differences between the predictions on a validation set of ensembles of sizes T and t with $t < T$ are statistically significant. The size of the ensemble is set to $T^\star$, which is the minimum size of a subensemble whose predictive accuracy does not significantly differ from an ensemble of T classifiers, with T sufficiently large. In a different work, [7] use out-of-bag data [25] to determine whether the generalization error has converged. First, the dependence of the out-of-bag error estimate on the size of the ensemble is smoothed by averaging over a sliding window of size 5. Then, the algorithm identifies the ensemble that has the best accuracy (i.e. the lowest smoothed value the error estimated on the out-of-bag data) among ensembles of sizes 1 to 20. Progressively larger ensembles are processed in batches of 20, until no improvement in accuracy is found. At this point, the algorithm outputs the ensemble with the maximum accuracy. The major advantage of this approach is that it does not require to overproduce and then discard classifiers. This algorithm will be used as a benchmark for comparison in the experiments section. Finally, the theoretical analysis of the dependence of the generalization error on the size of the ensemble size performed in [27] allows for the formulation of simple, quantitative and theoretically grounded guidelines for choosing a suitable size for bagging ensembles: By combining m bagged classifiers, one can expect to reach, on average, a fraction of (m-1)/m of the overall error reduction that can be attained by bagging an infinite number of classifiers with respect to using a single ensemble classifier. Even though their analysis assumes that the individual classifiers output a probability level and that the global prediction is obtained by a linear combination of these probabilities a similar analysis can be carried out for majority voting [28]. However, this rule has two drawbacks. First, it is based on relative gains. Therefore it cannot be used to provide absolute bounds for the improvement. Second, it does not depend on the specific properties of the task considered. This means the same bagging size should be suitable for every possible classification problem, which is contrary to the empirical evidence.

4. Experiments

In this section we illustrate the application of the proposed framework to determine the size of parallel classification ensembles. For this purpose experiments are carried out in a suite of binary classification problems from the UCI repository [29], from the R statistics software [30] and from the KEEL repository [31]. Table 1 displays the number of attributes and instances for each problem. For the synthetic classification problems (i.e. *Twonorm*, *Ringnorm*, *Circle* and *Spiral*) we use 100 independent realizations of the problem. In each realization, we generate a training set composed of 300 instances and a test set with 1000 instances, except for the *Spiral* problem, where $5,000$ instances are used for training and $10,000$ instances for testing. For the non-synthetic classification problems we make 100 independent random partitions into a training set and a test set using 2/3 and 1/3 of the total available data, respectively. Finally, in the dataset *Whitewine* the classification problem consists in discriminate between low (below 6) and

high-quality wines. In the *Abalone* dataset, the goal is to discriminate between infants and non-infants.

Table 1: Datasets used in the experiments.

Problem	**Attributes**	**Instances**	**Source**
Abalone	9	4,177	UCI
Australian	14	690	UCI
Banana	2	5,300	KEEL
Breast	9	699	UCI
Circle	2	300	R
Echo	12	131	UCI
German	20	1,000	UCI
Heart	13	270	UCI
Hepatitis	19	155	UCI
Horse	27	368	UCI
Ionosphere	34	351	UCI
Labor	16	57	UCI
Liver	6	345	UCI
Magic	10	19,020	UCI
Musk	166	6,598	UCI
Phoneme	5	5,404	UCI
Pima	8	768	UCI
Ringnorm	20	300	R
Sonar	60	208	UCI
Spam	57	4,601	UCI
Spiral	2	5,000	R
Tic-tac-toe	9	958	UCI
Twonorm	20	300	R
Votes	16	435	UCI
Whitewine	11	4,897	UCI

To build the classification ensembles we use two representative ensemble learning algorithms: bagging [1] with unpruned CART trees [32] and random forest (RF) [3]. In bagging, the individual classifiers are built by applying the CART algorithm to independent bootstrap samples of the training set [1]. Each bootstrap sample has the same size as the original training set and is obtained by drawing with replacement from this set. Random forest [3] was introduced as an improvement over bagging when the classifiers of the ensemble are decision trees. Besides resampling, random forest uses randomized decision trees. Specifically, the splits of the data in the internal nodes of these trees are generated by a greedy algorithm that considers for each split only a randomly selected subset of attributes. The number of randomly selected attributes is set to the default value in RF for classification, *i.e.* the square root of the total number of attributes. This value has been shown to provide good generalization accuracy in a large range of classification problems [33, 21]. Both RF and bagging are parallel ensemble learning algorithms in which the individual classifiers are built independently when conditioned

to the training data. Therefore, the framework introduced in Section 2 is appropriate to estimate an appropriate size for the ensemble.

4.1. Universal Asymptotic Behavior

We first present results that illustrate the universal behavior of $\mathcal{P}(\hat{y}^T \neq \hat{y}^\infty|T)$ (14) asymptotically, as $T \to \infty$. The experiments consist in comparing, for each test partition of a given problem, the predictions of an ensemble of size T, where T ranges from 1 to 5000, with the predictions of an ensemble of size $10,000$. This large ensemble serves as a proxy for the infinite-size ensemble because it assigns the asymptotic class label to all but a very small fraction of the test instances. The disagreement rate between the predictions of both ensembles provides an empirical estimate of $\mathcal{P}(\hat{y}^T \neq \hat{y}^\infty|T)$ for each test partition of the data.

Figure 4 displays the results of these experiments for a representative subset of problems considered in this work (*Heart*, *Pima*, *German*, *Echo* and *Phoneme*) and for both types of ensembles (bagging and random forest). Similar curves are obtained for the other classification problems investigated. The plots in this figure display the logarithm of the empirical estimates of $\mathcal{P}(\hat{y}^T \neq \hat{y}^\infty|T)$, averaged over the different test partitions, as a function of ensemble of size T for $T = 1, \ldots, 5000$. The corresponding asymptotic approximation

$$\log \mathcal{P}(\hat{y}^T \neq \hat{y}^\infty|T) \approx \log \left(f(\pi_1 = 1/2) \int_{-\infty}^{0} \Phi(z) dz \right) - \frac{1}{2} \log T, \tag{18}$$

derived from Eq. (14) is also depicted in each plot as a dashed line. In these curves, the value of $f(\pi_1 = 1/2)$ is estimated using the test set predictions of the proxy for the infinite ensemble. The approximately linear dependence of the empirical estimates of $\log \mathcal{P}(\hat{y}^T \neq \hat{y}^\infty|T)$ with $\log T$ with a slope $-1/2$ as $T \to \infty$ illustrates the validity of the asymptotic analysis presented in Section 2.2.

4.2. Estimation of the Ensemble Size

For each problem, ensemble method, and train and test partition, the size $T^\star(\alpha)$ of the classification ensemble is estimated using the procedure described in the previous section with $\alpha = 99\%$. We also compare two strategies for estimating $T^\star(\alpha)$ that differ in whether out-of-bag or test data are used to approximate $f(\pi_1)$. Once the estimate of $f(\pi_1)$ has been computed, a binary search procedure is used to find $T^\star(\alpha)$, the minimum value of T that fulfills (17). Only odd values of T are considered to avoid ties in the ensemble prediction. We refer to these ensembles as optimal, to reflect the fact that they are the smallest ensembles whose prediction coincides with the asymptotic limit with probability α, on average. For the purpose of comparison, we also generate ensembles whose sizes are estimated using the method proposed in [7]. To determine whether the average fraction of examples in which the prediction by the finite and the infinite ensemble differ is close to $1 - \alpha$, we compare the predictions of the finite ensembles with an ensemble of $10,000$ trees, which, as in the previous section, serves as a proxy for the infinite ensemble. The error rate of the different ensembles is estimated on the corresponding test set. For the ensembles whose size is determined using the method described in the previous section, the differences in error rate should be smaller than $1 - \alpha$, which is the average fraction of

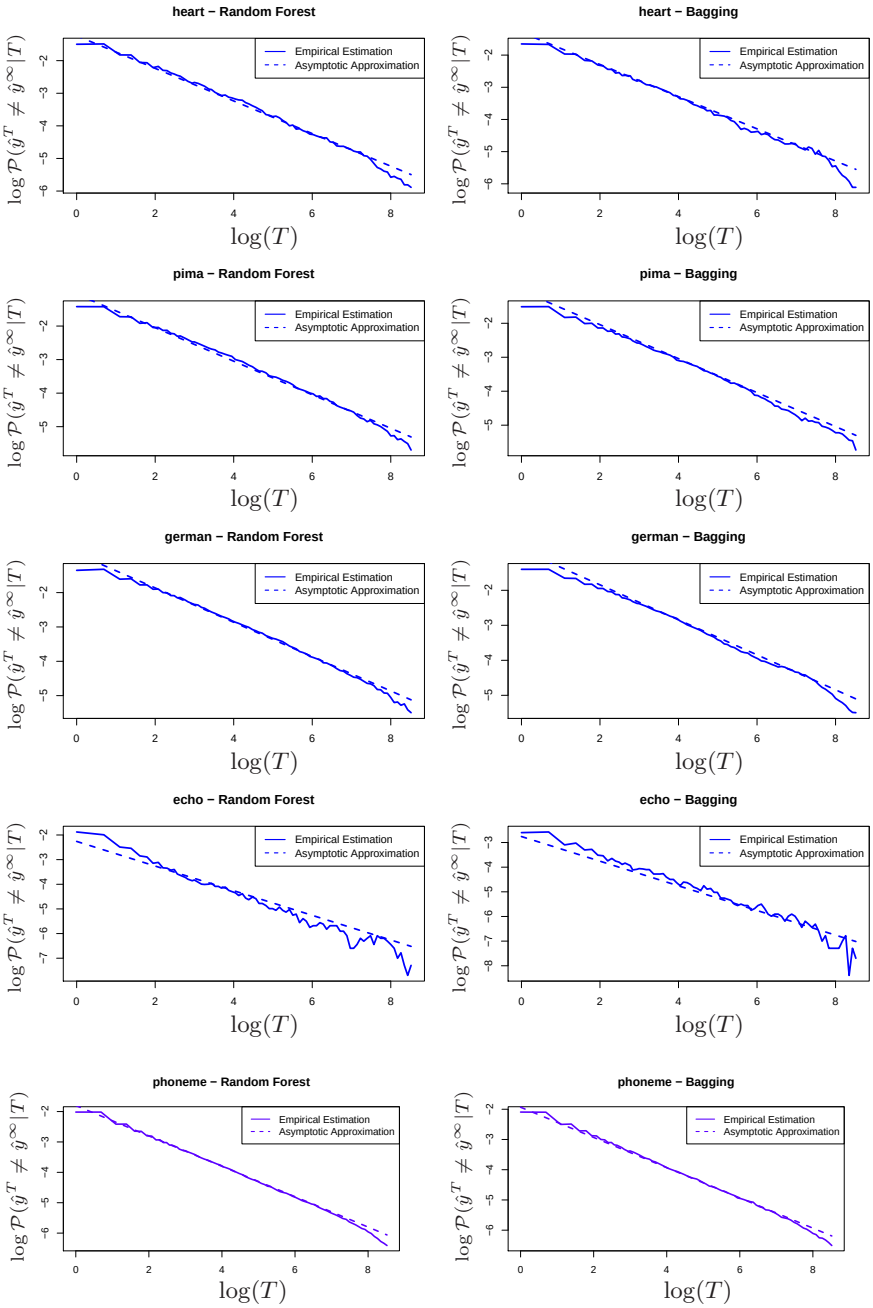

Figure 4: Logarithm of the fraction of disagreements in the test predictions between an ensemble of size T and an ensemble composed of $10,000$ classifiers, which is a proxy for the ensemble of infinite size. The dashed line corresponds to the asymptotic approximation Eq. (18).

Table 2: Average disagreement rates in the test set % between the finite and the asymptotic (infinite-size) ensemble prediction for bagging and RF.

Problem	RF-Test	RF-OOB	RF-BAN	Bag-Test	Bag-OOB	Bag-BAN
Abalone	1.0±0.2	1.0±0.3	2.2±0.7	1.1±0.2	1.0±0.3	2.1±0.5
Australian	1.0±0.6	1.2±0.7	2.3±1.1	1.0±0.6	1.1±0.7	2.3±1.3
Banana	1.0±0.2	1.0±0.2	1.4±0.5	1.0±0.2	1.0±0.3	1.4±0.4
Breast	0.9±0.6	1.0±0.7	0.6±0.5	0.9±0.5	0.9±0.7	0.8±0.6
Circle	1.0±0.4	1.1±0.5	1.3±0.6	1.0±0.4	1.0±0.5	1.3±0.7
Echo	1.0±1.5	1.1±1.8	2.2±2.4	1.2±1.5	1.1±2.0	2.0±2.6
German	1.1±0.5	1.2±0.6	5.1±1.5	1.1±0.6	1.2±0.6	5.7±2.1
Heart	1.2±1.1	1.3±1.2	4.7±3.1	1.3±1.0	1.2±1.1	4.9±3.4
Hepatitis	1.5±1.4	1.5±1.8	4.7±3.4	1.3±1.5	1.2±1.8	5.2±3.6
Horse	1.2±1.0	1.1±1.1	2.4±1.7	1.1±0.8	1.2±1.1	2.6±2.0
Ionosphere	0.9±0.8	1.0±0.8	1.5±1.2	0.9±0.8	1.1±1.0	1.8±1.5
Labor	1.8±2.8	1.9±2.9	3.5±4.9	1.4±2.6	1.7±3.7	3.2±4.2
Liver	1.5±1.1	1.5±1.2	8.5±3.5	1.3±1.0	1.2±0.9	7.6±4.0
Magic	1.0±0.1	1.0±0.1	1.4±0.3	1.0±0.1	1.0±0.1	1.4±0.3
Musk	0.9±0.2	0.8±0.2	0.4±0.1	1.0±0.2	0.9±0.2	0.4±0.2
Phoneme	1.0±0.2	1.0±0.2	1.7±0.5	1.0±0.2	1.0±0.3	1.6±0.4
Pima	1.1±0.7	1.0±0.7	5.2±2.1	1.3±0.6	1.2±0.7	5.2±2.2
Ringnorm	1.1±0.3	1.2±0.5	2.8±0.7	1.1±0.3	1.2±0.4	3.3±1.2
Sonar	1.4±1.2	1.9±1.7	8.1±3.9	1.3±1.4	1.4±1.6	7.0±4.1
Spam	1.0±0.3	0.9±0.3	0.8±0.3	1.0±0.3	1.0±0.3	0.8±0.3
Spiral	1.0±0.1	1.0±0.1	1.8±0.5	1.0±0.1	1.0±0.1	1.9±0.5
Tic-tac-toe	0.9±0.5	0.8±0.5	1.3±0.8	0.9±0.5	0.8±0.6	0.6±0.5
Twonorm	1.0±0.3	1.1±0.4	2.2±0.8	1.1±0.4	1.2±0.4	2.6±0.8
Votes	0.8±0.8	0.8±0.9	0.7±0.9	1.1±0.8	1.0±1.0	1.0±0.8
Whitewine	1.0±0.3	1.0±0.3	2.6±0.7	1.1±0.2	1.0±0.3	2.6±0.6

instances for which the assigned class label changes. The results presented are averages (± standard deviation) over the different realizations of the training and test data.

Table 2 displays for each problem and ensemble method (bagging and RF) the average disagreement rates between the finite and the asymptotic (infinite-size) ensemble predictions. The standard deviations of the disagreement rates are given after the ± symbol. In this table the suffix *OOB* indicates that the value of $T^\star$ is estimated using the out-of-bag data. The suffix *Test* indicates that the value of $T^\star$ is estimated using the test set (note that the class labels of these examples are not needed for this estimation). Finally, the suffix *BAN* indicates that the value of T has been estimated using the method proposed in [7].

These results show that the disagreement rates of the optimal ensembles are close to the $1 - \alpha = 1\%$ threshold level set in the experiments. By contrast, the disagreement rates of ensembles whose size is estimated with the method of [7] are rather disperse: The differences are small in some problems (e.g. 0.6% in Breast for RF, 0.6% in Tic-tac-toe, for bagging, and 0.4% in Musk, for both ensemble methods) and much larger in others (e.g. above 5% for German, Pima and Liver, for both RF and bagging).

Tables 3 and 4 display for each problem and for bagging and RF the asymptotic ensemble test error (Bag∞ and RF∞) and the average test error of the estimated ensembles (Bag-Test, Bag-OOB, Bag-BAN, RF-Test, RF-OOB, and RF-BAN), averaged over the 100 realizations of the classification problems considered. The standard deviations of these values are given after the $\pm$ symbol. As in the previous table, the procedure used for the estimation of the size of the ensemble is indicated by a suffix attached to the ensemble method (*OOB* for out-of-bag, *Test* for the method that uses the test set and *BAN* for the method proposed by [7]). The median of the number of trees used in these ensembles for the different realizations of the classification problems are also displayed. The interquartile interval is shown between parentheses. These measures are used instead of the mean and the standard deviation because they are more robust estimates of the center and dispersion of the ensemble sizes, respectively. To determine whether the differences in error rate are statistically significant we perform a Wilcoxon rank test [34]. Error rates that are significantly larger than the corresponding asymptotic ensemble level (a p-value below 5% is obtained in the Wilcoxon test) are highlighted in boldface.

Regarding the generalization performance, these results confirm that RF typically obtains lower error rates than bagging [3]. More relevant to this study, the lowest errors correspond to the asymptotic ensembles, as expected. The error rates of the ensembles estimated with the proposed method (Bag-OOB, Bag-Test, RF-OOB and RF-Test) are only slightly higher. In many problems the differences are not statistically significant. In all cases the increases in the error rate are much lower than the upper bound given by $1 - \alpha$, the fraction of instances whose class label prediction is expected to change from the finite to the infinite ensemble. This means that the changes in the predicted class-label occur in approximately equal numbers of correctly and of incorrectly classified instances. Ensembles whose size is estimated by the method of [7] typically have larger error rates than the proposed method regardless of whether the out-of-bag or the test data are used in the estimations of (15).

Tables 3 and 4 show that the values obtained for the optimum number of classifiers for the ensemble, $T^\star(\alpha)$, when $f(\pi_1)$ is estimated using out-of-bag data or using the test set are very similar. In addition, different classification problems require ensembles of very different sizes. Some classification tasks need ensembles of less than 50 trees to reach a stable prediction with a confidence level $\alpha = 99\%$ (e.g. *Votes*, *Breast* and *Musk*). For others, the appropriate number of trees to combine is in the thousands (e.g. *German*, *Pima*, *Sonar* and *Liver*). Therefore, contrary to the common practice in most of the literature on ensembles, one should not use the same number of classifiers irrespective of the problem considered. Furthermore, the ensembles used in previous studies are rarely above 100-200 classifiers, which is probably too small for some problems.

The sizes of the ensembles obtained using the recommendation of [7] (RF-BAN and Bag-BAN) are typically too small in classification problems with small numbers of training instances, such as *Labor*, *Sonar Heart* or *Hepatitis*. For these problems, RF-BAN and Bag-BAN typically yield ensembles with lower accuracy and higher disagreement rates than the method proposed in the current research. This behavior can be explained by the rather large variability of the out-of-bag estimates, even when they are smoothed by averaging. One empirically finds that this variability can lead to spurious variations in accuracy solely because of sample fluctuations. The premature convergence of this algorithm for small datasets is also discussed in [7] and is the main reason why their analysis focuses on large classification problems with several thousands of training in-

Table 3: Average and standard deviation of the test errors for the infinite-size and for the estimated RF ensembles. Median and interquartile interval (between parentheses) of the number of trees for the estimated RF ensembles.

Problem	RF∞	RF-Test	RF-OOB	RF-BAN	# Tree RF-Test	# Tree RF-OOB	# Tree RF-BAN
Abalone	16.67±0.68	16.72±0.69	**16.73±0.71**	**16.88±0.73**	391 (318, 474)	397 (363, 442)	92 (66, 120)
Australian	13.13±1.90	13.08±2.02	13.20±2.06	13.24±1.89	257 (192, 427)	238 (189, 318)	58 (43, 78)
Banana	10.77±0.61	**10.82±0.60**	10.81±0.61	**10.86±0.61**	108 (91, 133)	111 (99, 127)	65 (43, 94)
Breast	3.20±0.89	**3.55±1.00**	**3.57±1.02**	**3.40±0.94**	19 (15, 34)	23 (17, 28)	57 (36, 76)
Circle	5.30±1.14	**5.41±1.10**	**5.42±1.20**	**5.54±1.11**	64 (46, 87)	57 (35, 87)	41 (23, 61)
Echo	9.16±3.41	**9.59±3.50**	9.20±3.53	9.52±3.50	57 (24, 131)	88 (62, 117)	35 (18, 46)
German	24.16±1.77	24.21±1.65	24.19±1.74	**24.45±1.92**	1570 (1216, 2280)	1616 (1422, 2130)	78 (54, 102)
Heart	17.20±3.42	17.10±3.35	17.22±3.40	**17.90±3.63**	529 (320, 1079)	618 (404, 1088)	47 (32, 74)
Hepatitis	15.44±4.68	15.63±4.53	15.27±4.56	15.73±5.07	313 (178, 767)	532 (288, 768)	30 (20, 61)
Horse	14.07±2.83	14.26±2.90	14.22±2.90	**14.67±2.99**	191 (126, 350)	241 (164, 368)	73 (49, 110)
Ionosphere	6.72±1.97	6.78±1.93	**6.95±2.03**	**7.26±2.16**	66 (39, 100)	71 (53, 96)	41 (29, 61)
Labor	8.42±5.39	**9.53±5.43**	8.74±5.90	**9.89±7.42**	64 (37, 117)	78 (53, 175)	21 (14, 37)
Liver	28.16±4.05	28.17±3.86	28.37±3.98	**29.37±4.23**	2224 (1312, 4062)	2440 (1526, 3631)	54 (33, 81)
Magic	12.07±0.35	**12.14±0.34**	**12.13±0.33**	**12.18±0.36**	247 (226, 276)	257 (243, 270)	144 (109, 175)
Musk	2.46±0.32	**2.78±0.36**	**2.72±0.34**	**2.51±0.31**	17 (15, 19)	17 (17, 19)	84 (66, 107)
Phoneme	9.60±0.72	9.63±0.70	9.63±0.69	**9.77±0.66**	246 (206, 287)	267 (233, 297)	96 (76, 122)
Pima	24.05±2.10	24.07±2.06	24.05±2.00	**24.41±2.28**	1194 (798, 1904)	1258 (1000, 1598)	56 (36, 89)
Ringnorm	6.17±1.14	**6.29±1.09**	**6.26±1.17**	**6.86±1.15**	563 (429, 703)	443 (346, 638)	83 (64, 111)
Sonar	18.30±5.16	18.36±5.28	18.41±5.44	**19.38±5.05**	1975 (954, 3877)	2070 (1198, 3146)	58 (37, 85)
Spam	5.00±0.56	**5.08±0.61**	5.03±0.53	**5.09±0.53**	63 (53, 72)	64 (58, 73)	90 (70, 114)
Spiral	16.18±0.40	**16.23±0.40**	**16.22±0.42**	**16.30±0.40**	234 (214, 262)	238 (212, 265)	77 (58, 107)
Tic-tac-toe	2.01±0.85	**2.37±0.88**	**2.23±0.93**	**2.49±0.98**	143 (97, 195)	185 (148, 216)	116 (86, 141)
Twonorm	3.82±0.66	**3.96±0.64**	**3.98±0.71**	**4.55±0.78**	365 (286, 428)	315 (225, 454)	96 (62, 117)
Votes	3.82±1.52	**4.01±1.52**	**4.04±1.52**	3.93±1.55	20 (13, 36)	29 (19, 41)	44 (30, 61)
Whitewine	16.93±0.87	**17.01±0.88**	16.97±0.91	**17.12±0.86**	714 (570, 842)	716 (644, 788)	100 (78, 127)

Table 4: Average and standard deviation of the test errors for the infinite-size and for the estimated bagging ensembles. Median and interquartile interval (between parentheses) of the estimated sized for the bagging ensembles.

Problem	Bag∞	Bag-Test	Bag-OOB	Bag-BAN	# Tree Bag-Test	# Tree Bag-OOB	# Tree Bag-BAN
Abalone	17.13±0.69	17.14±0.74	17.10±0.71	**17.21±0.71**	367 (304, 439)	399 (352, 454)	88 (66, 127)
Australian	13.21±1.72	13.18±1.84	13.26±1.80	**13.50±1.91**	181 (123, 263)	189 (140, 268)	54 (30, 70)
Banana	11.16±0.69	**11.22±0.66**	11.19±0.71	**11.26±0.74**	99 (90, 120)	107 (97, 117)	62 (38, 83)
Breast	3.99±1.11	**4.17±1.11**	**4.20±1.12**	**4.17±1.10**	21 (17, 29)	23 (18, 33)	42 (28, 63)
Circle	6.00±1.39	**6.11±1.33**	**6.10±1.37**	**6.26±1.41**	45 (33, 68)	48 (36, 69)	35 (20, 56)
Echo	9.70±4.07	**10.18±4.22**	9.84±4.06	10.16±3.65	37 (15, 80)	60 (33, 89)	23 (13, 41)
German	24.22±1.98	24.20±2.06	24.23±2.05	**24.83±2.06**	1645 (1196, 2296)	1605 (1264, 2242)	68 (42, 91)
Heart	19.14±3.68	19.31±3.61	19.19±3.57	**20.28±3.85**	422 (208, 886)	481 (307, 718)	44 (23, 61)
Hepatitis	17.29±5.08	17.25±5.20	17.13±5.11	17.75±5.02	259 (142, 640)	442 (271, 698)	27 (15, 46)
Horse	14.93±3.23	**15.16±3.28**	14.98±3.42	**15.74±3.41**	182 (100, 300)	193 (144, 312)	59 (41, 73)
Ionosphere	7.88±2.14	7.93±2.37	8.05±2.35	8.09±2.14	81 (43, 123)	72 (55, 100)	41 (25, 61)
Labor	11.95±7.84	12.32±7.64	12.00±7.52	12.58±6.56	53 (29, 96)	78 (50, 146)	21 (12, 28)
Liver	29.32±3.77	29.56±3.81	29.34±3.93	**30.50±3.66**	1292 (761, 1944)	1643 (1228, 2437)	46 (31, 69)
Magic	12.36±0.36	**12.42±0.36**	**12.42±0.34**	**12.46±0.34**	230 (208, 250)	225 (215, 239)	114 (93, 156)
Musk	2.66±0.34	**2.89±0.37**	**2.87±0.37**	**2.71±0.34**	17 (15, 21)	19 (17, 21)	78 (54, 110)
Phoneme	9.91±0.76	**10.00±0.70**	**9.99±0.74**	**10.08±0.76**	213 (183, 248)	224 (197, 253)	93 (67, 121)
Pima	24.41±2.20	24.41±2.16	24.36±2.24	24.74±2.24	1041 (670, 1555)	970 (701, 1408)	52 (32, 71)
Ringnorm	8.93±1.98	**9.01±1.97**	**9.04±2.01**	**9.81±2.06**	722 (547, 916)	607 (379, 774)	74 (53, 98)
Sonar	21.58±5.67	21.59±5.75	21.71±5.84	22.10±5.71	997 (476, 2004)	1298 (743, 1871)	36 (22, 66)
Spam	5.93±0.60	**6.03±0.59**	**6.01±0.60**	**6.01±0.60**	47 (41, 54)	47 (41, 55)	78 (59, 102)
Spiral	16.65±0.41	**16.71±0.41**	**16.69±0.44**	**16.80±0.40**	239 (221, 270)	243 (223, 282)	73 (56, 108)
Tic-tac-toe	1.95±0.85	**2.37±0.90**	**2.27±0.96**	**2.05±0.86**	35 (25, 62)	45 (37, 58)	77 (62, 101)
Twonorm	6.17±1.41	**6.29±1.44**	**6.30±1.47**	**6.77±1.57**	514 (388, 618)	445 (346, 670)	78 (63, 108)
Votes	4.69±1.71	4.85±1.60	**4.86±1.69**	4.80±1.72	21 (12, 48)	26 (17, 37)	25 (16, 43)
Whitewine	17.35±0.91	**17.41±0.92**	17.37±0.91	**17.51±0.82**	682 (532, 793)	663 (585, 744)	107 (82, 140)

stances. In datasets with a larger number of training instances, the estimated ensemble sizes for RF-BAN and Bag-BAN tend to be larger, *e.g. Abalone*, *Magic* or *Whitewine*. In these datasets RF-BAN and Bag-BAN identify ensembles with a prediction performance similar to the asymptotic one.

The predictive accuracy of the different ensembles generated is compared using the statistical framework introduced in [34], for both RF and bagging. This framework allows to compare different classification systems on a collection of classification problems. To perform the comparison, the different methods are ranked according to their accuracy in each of the problems considered. Then, the average of the ranks obtained by each method in each of the problems is computed. Finally, a non-parametric statistical test is applied to determine whether the differences among the average ranks of the methods considered are statistically significant. In these tests, RF and bagging ensembles are analyzed separately. A Friedman test based on these average ranks rejects (with a p-value $< 5\%$) the null-hypothesis that there are no significant differences in accuracy among the different methods evaluated, for both bagging and RF. Finally, a Nemenyi post-hoc test is applied to determine whether the differences in average rank are statistically significant. If the differences between average ranks are above a critical distance (CD), which depends on the level of significance specified for the test, they are considered statistically significant.

Figure 5 displays the results of the Nemenyi post-hoc test for both RF (left) and bagging (right). Methods whose differences in average rank are not statistically significant at a level $\alpha = 5\%$ are connected with a horizontal solid line. The critical distance (CD) that marks whether the differences in average rank are statistically significant is displayed on the top of the plots. From these results we observe that the proxies of the infinite-size ensembles, RF-∞ and Bag-∞, have a significantly better average rank than any of the ensembles of finite size. There are no statistically significant differences between the average ranks of the proposed method when either out-of-bag or unlabeled test data are used to determine the optimal ensemble size. In contrast, the differences in accuracy between the method proposed in [7] (RF-BAN and Bag-BAN) and the one proposed in this work are statistically significant when out-of-bag data are employed (*i.e.* with respect to RF-OOB and Bag-OOB respectively). When using unlabeled test data for the estimations (*i.e.* RF-Test and BF-Test), the differences in average ranks with respect to Banfield's method are statistically significant for Random Forest, but not for bagging ensembles.

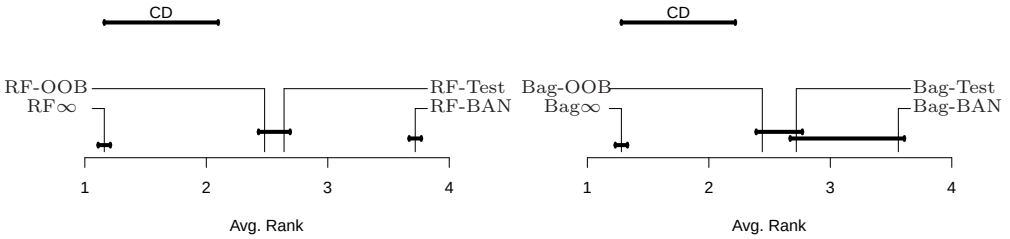

Figure 5: Average rank of each method displayed in Table 3 (left) and Table 4 (right) over the 25 classification problems investigated. The critical distance (CD) between average ranks, as estimated by a Nemenyi post-hoc test, is displayed on each figure for a p-value = 5%. The methods for which the differences in accuracy are not statistically significant are linked with a solid segment.

4.3. Dependence of the Ensemble Size on the Confidence Level

Additional experiments have been carried out to investigate the dependence of the ensemble size on the confidence level for the prediction α. Figure 6 plots the classification error and the median of the ensemble size as a function of α in five representative classification problems *Twonorm*, *German*, *Breast*, *Ionosphere* and *Musk*, for bagging and random forest ensembles. $T^\star(\alpha)$ is estimated using out-of-bag data. Similar curves are obtained for the other classification problems investigated. These curves illustrate the trade-off between the desired level of confidence in the predictions (α) and the number of classifiers that are required for prediction. The larger the value of α, the larger the number of classifiers that are needed to reach this confidence level. Specifically, when α approaches 100% there is a sharp increase in the number of classifiers required to achieve that level of confidence in the predictions. By contrast, the average test error decreases more slowly with increasing α, and only very small gains are obtained when the confidence level approaches 100%. The value of $\alpha = 99\%$ used in the previous experiments provides a good balance between the decrease in the average test error and the number of classifiers required to reach that confidence level. The results of this section agree with the dependence of $T^\star(\alpha)$ on α, as described by Eq. (16).

4.4. Comparison with Dynamic Ensemble Pruning Techniques

A final batch of experiments is carried out to determine whether the two dynamic ensemble pruning techniques described in [16, 17] are effective in reducing the number of queries for ensembles whose size is determined using the methods introduced in this work. These pruning techniques assume that an initial ensemble of appropriate size has been generated. The goal of dynamic pruning is to reduce the number of classifiers of a given initial ensemble that need to be queried to output a decision, without significant deterioration of the generalization performance.

For this comparison, we have carried out experiments in the classification problems that require large ensemble sizes, according to the results displayed in Tables 3 and 4. These datasets are *German*, *Heart*, *Liver*, *Pima*, *Ringnorm*, *Sonar*, *Twonorm* and *Whitewine*. The performance of dynamic pruning in ensembles whose size is determined by means the methods introduced in this work, using out-of-bag (OOB) data, is gauged against an ensemble composed of 101 classifiers. This particular value (101 classifiers) has been selected because it is a common choice for the ensemble size in the literature for bagging and RF [16, 17, 3, 4, 35, 21]. All the ensembles considered are then dynamically pruned using the two different methods described in [16, 17]. The confidence level for both pruning methods is set to 99%. This confidence level is with respect to the complete ensemble for [16] and with respect to the asymptotic (infinite) ensemble prediction for [17]. The average number of queried trees is also recorded. The computational cost of determining when to stop querying is negligible, provided that some computations are made before the prediction process starts [16, 17]. Therefore, the reduction in the number of trees queried directly translates into a reduction in the time required for classification. It is worth noting that for the dynamic pruning method introduced in [17], as discussed in Section 3, the prediction of some test instances at the desired confidence level may not be possible even after querying all the available classifiers. When this occurs, we simply return the prediction of the complete ensemble for those instances.

The results of the pruning experiments for RF are shown in Tables 5 for [16] and 6 for [17]. The results for bagging ensembles are very similar. The columns in the first

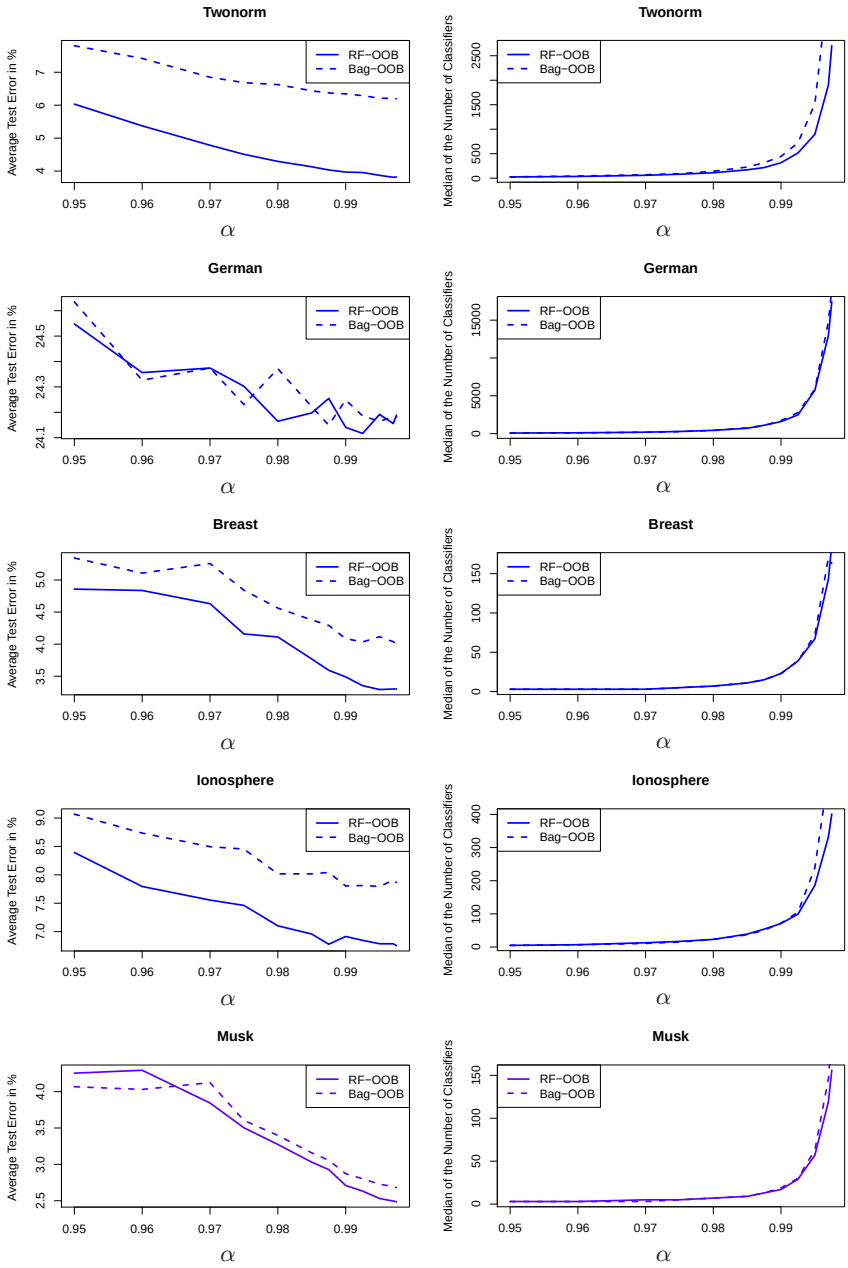

Figure 6: Average test error and median of the number of classifiers in the ensemble as a function of the confidence level α. The results are displayed for the classification problems *Twonorm*, *German*, *Breast*, *Ionosphere* and *Musk* for both bagging and RF. The out-of-bag data are used to estimate the ensemble size $T^\star(\alpha)$.

block of these tables display the average test errors of the ensembles of size 101 (RF-101) and of optimal size (RF-OOB), respectively. The columns in the second block show the test error of the dynamically pruned ensembles (DP-RF-101 and DP-RF-OOB). For each block, the error rates that are significantly smaller than the corresponding counterpart using a Wilcoxon rank test [34] (with p-value $< 5\%$) are highlighted in boldface. The median and the interquartile range of the average number of trees queried for each test instance when pruning is applied is also reported in these tables. Finally, the average speed-up factor –measured as the number of classifiers available for querying divided by the average number of classifiers actually used– is shown in the last column.

The results displayed in both tables are very similar. RF-OOB ensembles are typically more accurate than RF-101 ensembles either with or without pruning. RF-OOB outperforms RF-101 in 5 of the 8 classification problems analyzed. After applying the dynamic pruning technique, this number is reduced to 3. In addition, the average number of queries in RF-OOB ensembles is significantly reduced by the dynamic pruning strategies. In most cases, the speed-up factor is above 10, as can be seen from the values reported in the last column in tables 5 and 6. The reduction of the average number of classifiers that need to be queried for accurate prediction is greater for larger initial ensembles.

The different dynamic pruning methods are compared using the statistical framework introduced in [34]. Specifically, the four different methods (RF-101, RF-OOB, DP-RF-101 and DP-RF-OOB) are ranked according to their predictive accuracy in each of the classification tasks considered. Figure 7 displays the results of the comparison among average ranks for the dynamic pruning method described in [16] (left) and for the dynamic pruning method described in [17] (right). Similar conclusions are obtained in both cases. The comparison shows that there are significant differences between the average ranks of RF-101 and RF-OOB. In contrast, there are no statistically significant differences between the average ranks of RF-OOB and RF-101 and the corresponding dynamically pruned counterparts. In the set of problems analyzed RF-OOB is better than RF-101, both without and with pruning. Note, however, that the RF-OOB ensembles considered are larger than the corresponding RF-101 ensembles

In summary, the results of these experiments show that the dynamical pruning techniques introduced in [16, 17] are effective in reducing the number of required queries also for ensembles whose size is determined with the methods introduced in this work. The relative improvements of classification speed are more significant for larger ensembles. This means that the overhead in cost of classification introduced by having larger initial ensembles can be significantly reduced by applying dynamic pruning methods.

Table 5: Average and standard deviation of the test errors for the RF ensembles of 101 classifiers and for the estimated RF ensembles using out-of-bag data. We also show results when the dynamic pruning technique described in [16] is used in classification time. The median and interquartile range of the average number of trees queried for each test instance and the average improvement in the classification time are also reported.

	Not Pruned		**Dynamically Pruned**		**Median # of Trees**				**Speed-up**
Problem	**RF-101**	**RF-OOB**	**RF-101**	**RF-OOB**	**RF-101**		**RF-OOB**		**RF-OOB**
German	24.29±2.00	24.19±1.74	24.34±2.08	24.25±1.79	29	(27, 30)	110	(98, 127)	15.73±3.93
Heart	17.34±3.57	17.22±3.40	17.44±3.62	17.34±3.40	23	(21, 25)	55	(44, 66)	13.76±7.41
Liver	28.61±4.12	28.37±3.98	28.59±3.87	28.49±4.08	34	(33, 36)	162	(128, 214)	16.30±9.58
Pima	24.31±2.06	**24.05±2.00**	24.39±2.06	24.13±2.05	26	(24, 27)	84	(71, 96)	15.63±5.08
Ringnorm	6.73±1.16	**6.26±1.17**	6.87±1.17	**6.38±1.20**	23	(22, 24)	42	(37, 49)	11.38±3.85
Sonar	19.03±5.24	**18.41±5.44**	19.03±5.15	18.72±5.48	32	(30, 34)	148	(119, 203)	14.39±7.23
Twonorm	4.40±0.80	**3.98±0.71**	4.51±0.80	**4.14±0.75**	21	(20, 22)	32	(28, 36)	10.62±3.87
Whitewine	17.14±0.90	**16.97±0.91**	17.19±0.89	**17.00±0.90**	21	(21, 22)	50	(48, 54)	14.35±1.87

Table 6: Average and standard deviation of the test errors for the RF ensembles of 101 classifiers and for the estimated RF ensembles using out-of-bag data. We also show results when the dynamic pruning technique described in [17] is used in classification time. The median and interquartile range of the average number of trees queried for each test instance and the average improvement in the classification time are also reported.

	Not Pruned		**Dynamically Pruned**		**Median # of Trees**				**Speed-up**
Problem	**RF-101**	**RF-OOB**	**RF-101**	**RF-OOB**	**RF-101**		**RF-OOB**		**RF-OOB**
German	24.29±2.00	24.19±1.74	24.29±2.03	24.25±1.79	38	(36, 39)	143	(124, 165)	12.23±3.13
Heart	17.34±3.57	17.22±3.40	17.36±3.60	17.27±3.44	30	(28, 32)	70	(54, 84)	11.06±6.36
Liver	28.61±4.12	28.37±3.98	28.58±3.98	28.54±4.15	46	(44, 48)	207	(164, 265)	12.97±8.86
Pima	24.31±2.06	**24.05±2.00**	24.34±2.07	24.12±2.04	34	(32, 35)	108	(91, 124)	12.27±4.08
Ringnorm	6.73±1.16	**6.26±1.17**	6.80±1.18	**6.35±1.17**	30	(28, 31)	53	(47, 62)	9.07±3.18
Sonar	19.03±5.24	**18.41±5.44**	18.96±5.25	18.67±5.38	42	(40, 46)	200	(155, 260)	11.24±5.68
Twonorm	4.40±0.80	**3.98±0.71**	4.46±0.80	**4.10±0.73**	27	(26, 28)	41	(36, 45)	8.43±3.12
Whitewine	17.14±0.90	**16.97±0.91**	17.19±0.90	**17.00±0.90**	27	(27, 28)	64	(60, 68)	11.36±1.55

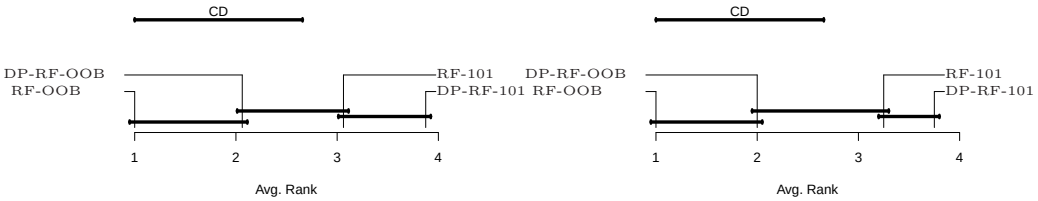

Figure 7: Average rank of each method displayed in Table 5 (left) and Table 6 (right) over the 8 classification problems investigated. The critical distance (CD) between average ranks, as estimate by a Nemyi post-hoc test, is displayed on each figure for a p-value = 5%. The methods for which the differences in accuracy are statistically significant are linked with a solid segment. When a dynamic pruning method has been used during classification, the prefix DP has been added.

5. Conclusions

In this research we have addressed the question of how to determine the size of parallel classification ensembles. The method proposed consists in estimating the number of classifiers that are necessary to reach a prediction that, on average, coincides with a hypothetical ensemble of infinite size with high probability $\alpha \approx 1$. In contrast to previous proposals found in the literature this procedure is not based on estimating the generalization error. Instead, it relies on the analysis of the convergence of the prediction of parallel classification ensembles as a function of ensemble size in the asymptotic regime, when the number of classifiers in the ensemble tends to infinity. The framework is valid for any classification problem and any parallel ensemble provided that the individual classifiers are built in independent applications of a randomized learning algorithm on the training data and that their predictions are combined by majority voting. The analysis performed shows that, while most of the instances require only a few classifiers to reach the infinite ensemble prediction with a high confidence, the predictions of a small but not negligible fraction of instances require extremely large numbers of queries to converge. We demonstrate and provide empirical evidence that this observation leads to universal behavior that emerges when large ensembles are used for prediction. In particular, the fraction of instances whose predicted class label differs from the asymptotic prediction is proportional to $T^{-1/2}$. The proportionality constant is determined by the probability density of instances with $\pi_1 \approx 1/2$. In consequence, the behavior of sufficiently large ensembles is determined by the fraction of data instances whose prediction by the ensemble is uncertain (i.e. instances with $\pi_1 \approx 1/2$).

The validity of the probabilistic framework developed is illustrated using two representative parallel ensemble learning algorithms (bagging and RF) for a wide range of classification problems. Given the generality of this analysis, similar behavior should be obtained for any type of parallel randomized ensemble, any type of base learner and for any classification problem considered. From the results of the empirical study one observes that the predictions of the finite classification ensembles constructed agree with the asymptotic ones with a probability close to α, the target confidence level used to determine the ensemble size. Because the differences in error are bound from above by $1 - \alpha$, the prediction accuracy of the optimal ensembles is only slightly lower than

the corresponding infinite-size ensembles. The value $\alpha = 99\%$ provides a good balance between accuracy and ensemble size.

The method proposed has been evaluated in experiments in a wide range of classification problems. The results of this empirical evaluation show that ensembles whose size is determined by requiring stability of the class prediction are, in the cases analyzed, more accurate than ensembles whose size is determined by requiring that the prediction error be stable, as in [7]. Even though the improvements are statistically significant when $\alpha = 99\%$, the differences are small in absolute value. Furthermore, the resulting ensembles are larger than in [7]. Nonetheless, if there are strict memory restrictions, one can consider smaller values of α (*e.g.* 97% or 98%) to decrease the size of the resulting ensembles at the expense of lower classification accuracy. Based on the results of the experiments carried out, using the stability of class predictions to determine the size of the ensemble leads to more consistent performance than using estimates of the generalization error. Another important conclusion of our study, which has also been pointed out by [7], is the need to adapt the ensemble size to the particular classification problem considered. Some problems require ensembles of only tens of classifiers to converge to the infinite ensemble prediction. In others, very large sizes are required for the ensemble predictions to stabilize. Finally, when the required ensemble size for a given classification problem is large, the dynamic pruning methods described in [16, 17] can be used to reduce the number of classifiers that have to be queried for classification.

Regarding future work, the current analysis can be extended to consider sequential ensembles, such as boosting [36], or combination schemes different from majority voting, such as weighted majority voting, or averages of the estimated probabilities of observing the different class labels at a given input location [27].

Appendix A.

In this appendix we derive an approximation of the asymptotic behavior of $\mathcal{P}(T^\star(\alpha, \pi_1) > T)$ as $T \to \infty$. Let $F(\pi_1)$ be the cumulative distribution function of π_1 (see Figure 1). In terms of this distribution, $\mathcal{P}(T^\star(\alpha, \pi_1) > T)$ can be estimated as the fraction of the instances whose value of π_1 is in the interval

$$\left[I^{-1}_{1-\alpha}\left(\lfloor \frac{T}{2} \rfloor + 1, T - \lfloor \frac{T}{2} \rfloor \right), I^{-1}_{\alpha}\left(\lfloor \frac{T}{2} \rfloor + 1, T - \lfloor \frac{T}{2} \rfloor \right) \right], \tag{A.1}$$

where $\pi_1 = I^{-1}_{1-\alpha}\left(\lfloor \frac{T}{2} \rfloor + 1, T - \lfloor \frac{T}{2} \rfloor \right)$ is the inverse of the incomplete beta function described in (7). That is,

$$\begin{aligned} \mathcal{P}(T^\star(\alpha, \pi_1) > T) = & F\left(I^{-1}_{\alpha}(\lfloor \frac{T}{2} \rfloor + 1, T - \lfloor \frac{T}{2} \rfloor) \right) \\ & - F\left(I^{-1}_{1-\alpha}(\lfloor \frac{T}{2} \rfloor + 1, T - \lfloor \frac{T}{2} \rfloor) \right). \end{aligned} \tag{A.2}$$

Taking the limit $T \to \infty$

$$\begin{aligned}
\mathcal{P}(T^\star(\alpha,\pi_1) > T) &\approx f(\frac{1}{2})\left(I_\alpha^{-1}(\lfloor\frac{T}{2}\rfloor+1, T-\lfloor\frac{T}{2}\rfloor) - I_{1-\alpha}^{-1}(\lfloor\frac{T}{2}\rfloor+1, T-\lfloor\frac{T}{2}\rfloor\right) \\
&\approx f(\frac{1}{2})\left(1-2I_{1-\alpha}^{-1}(\lfloor\frac{T}{2}\rfloor+1, T-\lfloor\frac{T}{2}\rfloor)\right) \\
&\approx f(\frac{1}{2})\left(\frac{1}{\sqrt{1+T/\left(\Phi^{-1}(\alpha)\right)^2}}\right) \\
&\approx \frac{f(\frac{1}{2})\Phi^{-1}(\alpha)}{\sqrt{T}}, \qquad \text{(A.3)}
\end{aligned}$$

where we have used the same approximation of the incomplete beta function as in (11) and $f(\pi_1 = 1/2) > 0$ has been assumed.

Appendix B.

In this appendix we derive an approximation of the asymptotic behavior of $\mathcal{P}(\hat{y}^T \neq \hat{y}^\infty|T)$ as $T \to \infty$. This probability is defined as

$$\mathcal{P}(\hat{y}^T \neq \hat{y}^\infty|T) = \int_0^1 \mathcal{P}(\hat{y}^T \neq \hat{y}^\infty|\pi_1, T) f(\pi_1) d\pi_1, \qquad \text{(B.1)}$$

where $f(\cdot)$ is the probability density function that an arbitrary test instance has a fixed associated value of π_1, and $\mathcal{P}(\hat{y}^T \neq \hat{y}^\infty|\pi_1, T) = 1 - \mathcal{P}(\hat{y}^T = \hat{y}^\infty|\pi_1, T)$, with $\mathcal{P}(\hat{y}^T = \hat{y}^\infty|\pi_1, T)$ defined in (7). For large T, $\mathcal{P}(\hat{y}^T \neq \hat{y}^\infty|\pi_1, T)$ can be approximated using (11) as

$$\mathcal{P}(\hat{y}^T \neq \hat{y}^\infty|\pi_1, T) \approx \Phi\left(-\frac{\sqrt{T}\max(\pi_1 - 1/2, 1/2 - \pi_1)}{\sqrt{\pi_1(1-\pi_1)}}\right). \qquad \text{(B.2)}$$

Substituting (B.2) into (B.1) and taking the limit $T \to \infty$ gives

$$\begin{aligned}
\mathcal{P}(\hat{y}^T \neq \hat{y}^\infty|T) &\approx \int_0^1 \Phi\left(-\frac{\sqrt{T}\max(\pi_1 - 1/2, 1/2 - \pi_1)}{\sqrt{\pi_1(1-\pi_1)}}\right) f(\pi_1) d\pi_1 \\
&\approx \int_{-1/2}^{1/2} \Phi\left(-\frac{2\sqrt{T}|x|}{\sqrt{1-4x^2}}\right) f(x+1/2) dx \\
&\approx f(1/2) \int_{-C/\sqrt{T}}^{C/\sqrt{T}} \Phi\left(-2\sqrt{T}|x|\right) dx \\
&\approx \frac{f(1/2)}{2\sqrt{T}} \int_{-2C}^{2C} \Phi\left(-|z|\right) dz \\
&\approx \frac{f(1/2)}{\sqrt{T}} \int_{-\infty}^{0} \Phi\left(z\right) dz, \qquad \text{(B.3)}
\end{aligned}$$

where we have used two changes of variables $\pi_1 = x + 1/2$ and $z = 2\sqrt{T}x$, and C is a positive constant such that $\Phi(-2\omega) \approx 0, \forall \omega > C$ (e.g. $C = 10$). We also assume $f(\pi_1 = 1/2) > 0$.

References

[1] L. Breiman, Bagging predictors, Machine Learning 24 (2) (1996) 123–140.

[2] T. G. Dietterich, An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization, Machine Learning 40 (2) (2000) 139–157.

[3] L. Breiman, Random forests, Machine Learning 45 (1) (2001) 5–32.

[4] P. Buhlmann, Bagging, subagging and bragging for improving some prediction algorithms, in: M. Akritas, D. Politis (Eds.), Recent Advances and Trends in Nonparametric Statistics, 2003, pp. 19–34.

[5] G. Martinez-Muñoz, A. Suárez, Switching class labels to generate classification ensembles, Pattern Recognition 38 (10) (2005) 1483–1494.

[6] J. J. Rodriguez, L. I. Kuncheva, C. J. Alonso, Rotation forest: A new classifier ensemble method, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (10) (2006) 1619–1630. doi:http://doi.ieeecomputersociety.org/10.1109/TPAMI.2006.211.

[7] R. E. Banfield, L. O. Hall, K. W. Bowyer, W. P. Kegelmeyer, A comparison of decision tree ensemble creation techniques, IEEE Transactions on Pattern Analisys and Machine Intelligence 29 (1) (2007) 173–180. doi:http://dx.doi.org/10.1109/TPAMI.2007.2.

[8] R. Schapire, Y. Freund, P. Bartlett, W. Lee, Boosting the margin: A new explanation for the effectiveness of voting methods, The Annals of Statistics 12 (5) (1998) 1651–1686.

[9] D. Opitz, R. Maclin, Popular ensemble methods: An empirical study, Journal of Artificial Intelligence Research 11 (1999) 169–198.

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

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

[12] R. Esposito, L. Saitta, Monte Carlo theory as an explanation of bagging and boosting, in: Proceeding of the Eighteenth International Joint Conference on Artificial Intelligence, Morgan Kaufmann, 2003, pp. 499–504.

[13] D. Ruta, B. Gabrys, A theoretical analysis of the limits of majority voting errors for multiple classifier systems, Pattern Analysis and Applications 5 (4) (2002) 333–350.

[14] L. Kuncheva, C. Whitaker, C. Shipp, R. Duin, Limits on the majority vote accuracy in classifier fusion, Pattern Analisys and Applications 6 (1) (2003) 22–31.

[15] A. Narasimhamurthy, Theoretical bounds of majority voting performance for a binary classification roblem, IEEE Transactions on Pattern Analisys and Machine Intelligence 27 (12) (2005) 1988–1995.

[16] D. Hernández-Lobato, G. Martínez-Muñoz, A. Suárez, Statistical instance-based pruning in ensembles of independent classifiers, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2) (2009) 364–369.

[17] D. Hernández-Lobato, G. Martńez-Muñoz, A. Suárez, Inference on the prediction of ensembles of infinite size, Pattern Recognition 44 (2011) 1426–1434.

[18] R. Bryll, R. Gutierrez-Osuna, F. Quek, Attribute bagging: Improving accuracy of classifier ensembles by using random feature subsets, Pattern Recognition 36 (6) (2003) 1291–1302.

[19] G. Martínez-Muñoz, A. Suárez, Out-of-bag estimation of the optimal sample size in bagging, Pattern Recognition 43 (1) (2010) 143 – 152.

[20] L. Breiman, Randomizing outputs to increase prediction accuracy, Machine Learning 40 (3) (2000) 229–242.

[21] P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees, Machine Learning 36 (1) (2006) 3–42.

[22] D. Hernández-Lobato, G. Martínez-Muñoz, A. Suárez, On the independence of the individual predictions in parallel randomized ensembles, to appear in the 20th European Symposium on Artificial Neural Networks (2012).

[23] M. Abramowitz, I. A. Stegun, Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, ninth dover printing, tenth gpo printing Edition, Dover, New York, 1964.

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

[25] L. Breiman, Out-of-bag estimation, Tech. rep., Statistics Department, University of California (1996).

[26] P. Latinne, O. Debeir, C. Decaestecker, Limiting the number of trees in random forests, in: MCS '01: Proceedings of the Second International Workshop on Multiple Classifier Systems, Springer-Verlag, London, UK, 2001, pp. 178–187.

[27] G. Fumera, R. Fabio, S. Alessandra, A theoretical analysis of bagging as a linear combination of

classifiers, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (7) (2008) 1293–1299.

[28] D. Hernández-Lobato, G. Martínez-Muñoz, A. Suárez, Out of bootstrap estimation of generalization error curves in bagging ensembles, in: H. Yin, P. Tiño, E. Corchado, W. Byrne, X. Yao (Eds.), Intelligent Data Engineering and Automated Learning - IDEAL 2007, 8th International Conference, Birmingham, UK, Proceedings, Vol. 4881 of Lecture Notes in Computer Science, Springer, 2007, pp. 47–56.

[29] A. Frank, A. Asuncion, UCI machine learning repository (2010).
URL `http://archive.ics.uci.edu/ml`

[30] R Development Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria, ISBN 3-900051-07-0 (2005).

[31] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework, Multiple-Valued Logic and Soft Computing 17 (2-3) (2011) 255–287.

[32] L. Breiman, J. H. Friedman, R. A. Olshen, C. J. Stone, Classification and Regression Trees, Chapman & Hall, New York, 1984.

[33] S. Bernard, L. Heutte, S. Adam, Influence of hyperparameters on random forest accuracy, in: J. A. Benediktsson, J. Kittler, F. Roli (Eds.), Proceedings of the 8th International Workshop on Multiple Classifier Systems, Vol. 5519 of Lecture Notes in Computer Science, Springer, 2009, pp. 171–180.

[34] J. Demšar, Statistical comparisons of classifiers over multiple data sets, Journal of Machine Learning Research 7 (2006) 1–30.

[35] G. Martínez-Muñoz, D. Hernández-Lobato, A. Suárez, An analysis of ensemble pruning techniques based on ordered aggregation, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2009) 245–259.

[36] Y. Freund, R. E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, in: Proc. 2nd European Conference on Computational Learning Theory, 1995, pp. 23–37.