Incorporating prior knowledge in dynamical ensemble pruning strategies

Anonymous Author(s)
Affiliation
Address
email

Abstract

The goal of dynamical ensemble pruning is to determine the minimum number of classifiers whose predictions need to be known to achieve a stable class prediction. This number can be very different depending on the particular instance that is being classified. In ensembles of independent classifiers, whose individual predictions are combined via majority voting, it is possible to determine when the partial tally of votes provides sufficient information about the final prediction by the complete ensemble. Starting from a prior that reflects the initial uncertainty on the final ensemble prediction, Bayes' theorem is used to update this uncertainty as the individual classifier votes become known. Dirichlet and, specially, mixtures of Dirichlet distributions are flexible models that can be used to approximate these priors in a wide range of classification problems. Incorporating this problem-specific knowledge in the statistical analysis of majority voting leads to higher pruning rates, and, therefore, to faster classification.

1 Introduction

Ensemble learning consists in combining the predictions of complementary classifiers to achieve improvements in generalization [1, 2, 3, 4, 5, 6]. In this work, we focus on homogeneous ensembles whose elements are generated in independent applications of a randomized learning algorithm on a fixed training dataset. These include bagging [7], random forests [8] or rotation forests [9]. In these types of ensembles, the base learners are independent conditioned on the training data. For instance each individual classifier in bagging is generated by applying a fixed learning algorithm (e.g. CART, C4.5, neural networks trained with backpropagation, etc.) on an independent bootstrap sample of the original training data. Therefore, the base learners generated are independent random functions, and their predictions on a given test example are independent random variables.

One of the drawbacks of ensembles is their large memory requirements and the computational cost of their predictions . These shortcomings can be alleviated using ensemble pruning algorithms. There are two types of pruning techniques for ensembles. In static pruning one selects a subset of classifiers to build a smaller ensemble [10, 11]. In dynamic pruning, all the classifiers are stored in memory. However, the number of classifiers that are actually queried to produce a prediction depends on the test example considered. For parallel ensembles that use majority voting [1], dynamical pruning can be implemented by stopping the voting process when it is unlikely that the remaining (unknown) votes change the class prediction [12, 13, 14]. This is the basis of *Statistical Instance-Based Pruning* (SIBP) [14]. In SIBP the classifiers are queried sequentially. At each step, the votes recorded are used to estimate the probability that the majority class predicted by the classifiers queried up to that moment and by the full ensemble coincide. If this probability exceeds a specified confidence level α, the classification process is halted. To compute this estimate, the probabilities that an individual ensemble classifier predicts a given class label for the particular instance considered is modeled as a random variable. Starting from a uniform prior, Bayes' theorem is used to update the distribution of

this variable with the information provided by the actual votes, as they become known. In most of the problems analyzed in [14], the assumption that the prior is uniform leads to conservative estimates of the confidence on the stability of the predictions when only a fraction of the classifiers have been queried. Analyzing the results of those experiments, it is apparent that the actual disagreement rates between the dynamically pruned ensembles and the complete ensembles are significantly lower that the specified target α. As a consequence, more queries are made that are actually needed.

The distribution of the class-prediction probabilities is in general not uniform. It depends on the classification task considered and on the learning algorithm used to generate the ensemble classifiers. In this work, we propose to approximate this distribution using a Dirichlet or Dirichlet mixture models. We also carry out a statistical analysis of majority voting that takes into accounts these non-uniform priors. The parameters of the Dirichlet and for the Dirichlet mixture models can be estimated for the corresponding classification problem using a validation set or by cross-validation. For random forests, the ensembles considered in this work, the out-of-bag set provides a more data-effective way to carry out this estimation. The use of these problem-specific knowledge leads to more accurate estimations of the disagreement rates between the dynamically pruned ensemble and the complete ensemble, which are closer to the specified target α. In this manner higher pruning rates (i.e. faster classification) are achieved.

The article is organized as follows: section 2 provides a description of the dynamical ensemble pruning algorithm analyzed (statistical instance-based pruning). Sections 3 and 4 describe how the distribution on the probabilities of class predictions can be modeled using a Dirichlet model or mixtures of Dirichlet models respectively. In Section 5 the validity of the proposed framework and the improvements in the dynamical pruning algorithm are illustrated in a suite of benchmark problems. Finally, section 6 summarizes the conclusions of the research.

2 Statistical instance-based pruning

Let $\{h_i(\cdot)\}_{i=1}^{T}$ be an ensemble of T classifiers whose final prediction is computed by majority voting

$$\underset{y}{arg\,max} \sum_{i=1}^{T} \mathbb{I}\left(h_i(\mathbf{x}) = y\right) \, , y \in \mathcal{Y}. \tag{1}$$

where $h_i(\mathbf{x})$ is the class prediction of the ith member of the ensemble, $\mathbb{I}$ is an indicator function and $\mathcal{Y} = \{y_1, \ldots, y_l\}$ is the set of possible class labels.

We consider ensembles of classifiers that are generated in independent applications of a randomized algorithm on the training data. The classification of instance $\mathbf{x}$ by majority voting in an ensemble of size t is a sequence of t independent trials. The outcome of each trial is in the set $\mathcal{Y}$. The probabilities of each of these outcomes can be collected the probability vector

$$\mathbf{p}(\mathbf{x}) = \{p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_l(\mathbf{x})\}, \quad \sum_{i=1}^{l} p_i(\mathbf{x}) = 1, \tag{2}$$

where $p_i(\mathbf{x}) \geq 0$ is the probability that instance $\mathbf{x}$ is assigned class label y_i by an individual ensemble classifier. The values of these probabilities, which are initially unknown, depend on the base learning algorithm, on the particular classification problem considered and on the instance that is being classified $\mathbf{x}$.

Given $\mathbf{p}(\mathbf{x})$, the probability distribution of the results of these t experiments is the multinomial distribution

$$\mathcal{P}(\mathbf{t}|\mathbf{p}) = \frac{t!}{t_1!t_2!\ldots t_l!}\, p_1^{t_1} p_2^{t_2} \ldots p_l^{t_l}\,, \tag{3}$$

where $\mathbf{t} = \{t_1, t_2, \ldots, t_l; \sum_{i=1}^{l} t_i = t\}$,. The elements of this vector, t_i, are the votes for class y_i votes for $\mathbf{x}$, the instance that is being classified. To simplify the notation the dependence on $\mathbf{x}$ of $\mathbf{p}$ has been omitted.

In the original formulation of the dynamical pruning algorithm, it was assumed that no information on the distribution of values of the probability vector in (3) is initially available. Consequently, the initial estimate of $\mathcal{P}(\mathbf{p})$ was taken to be uniform in the probability simplex. Bayes' theorem was then

used to sequentially update this distribution using the evidence from the individual predictions of the ensemble classifiers, as these became known. Thus, the voting process is used not only to determine the majority class, but also to improve our estimate of the distribution of (2). This estimate can be used to halt the querying process when the confidence on the stability of the partial ensemble prediction is above a specified confidence level α.

Starting from the uniform estimate for $\mathcal{P}(\mathbf{p})$, it is possible to show that the probability that the class labels predicted by the subensemble of size $t < T$ and by the complete ensemble of size T coincide is

$$\mathcal{P}^*(\mathbf{t}, T) = \sum_{\mathbf{T} \in \mathcal{T}_\mathbf{t}} \frac{(T-t)!}{\prod_{i=1}^{l}(T_i - t_i)!} \frac{\prod_{i=1}^{l}(t_i+1)_{T_i - t_i}}{(t+l)_{T-t}} \tag{4}$$

where $k_\mathbf{t} = \arg\max_i t_i$ is the majority class derived from the vector of votes $\mathbf{t}$ and $\mathcal{T}_\mathbf{t}$ is the set of vectors of votes for the complete ensemble $\mathbf{T} = \{T_1, T_2 \dots T_l\}$ such that the class predicted by the subensemble of size t and the class predicted by the complete ensemble coincide: $k_\mathbf{t} = k_\mathbf{T}$, $T_i \geq t_i$ and $\sum_{i=1}^{l} T_i = T$. Dynamical pruning consists in halting the voting process when t^* classifiers have been queried, where $\mathbf{t}^*$ is such that $\mathcal{P}^*(\mathbf{t}^*, T) \geq \alpha$.

In this article we propose to use a more sophisticated initial estimate for $\mathcal{P}(\mathbf{p})$ that accurately reflects the information on the distribution of (3) available *before* the voting process starts. This initial estimate can be made using, for example, a validation set, cross-validation or, in bagging and in random forests, out-of-bag data. Specifically, suppose that such a collection of labeled examples $\{(\mathbf{x}_j, y_j)\}_{j=1}^{J}$ is available. The initial estimate of $\mathcal{P}(\mathbf{p})$ is assumed to be of the Dirichlet type. This form is convenient because it is conjugate to a multinomial likelihood and leads to updates of the same form. However, the Dirichlet family is not sufficiently flexible to capture the actual patterns observed in the data. Therefore, we consider Dirichlet mixtures, and assume that the examples in each class can be modeled by a separate component in the mixture. The parameters of these models are determined using the sample estimates of $p_i(\mathbf{x}_j) \approx t_i^j/T$, where t_i^j is the number of votes assigned to class i by the original ensemble of size T for instance $\mathbf{x}_j$.

The experiments performed in Section 5 show that by incorporating this information in the initial estimate $\mathcal{P}(\mathbf{p})$ the disagreement rates approach the established target α. Consequently the classification speed significantly improves, specially if the actual distribution of $\mathcal{P}(\mathbf{p})$ is very different from the assumed uniform distribution. In addition, the test error rates are still close to the error rates achieved by the original ensemble.

3 Majority voting with Dirichlet priors

As discussed in the previous section, the dynamical pruning described in [14] can be improved by using an estimate of $\mathcal{P}(\mathbf{p})$ that reflects the actual distribution of the class-prediction probabilities in the classification problem considered. Let's assume that, instead of uniform, this distribution is of the Dirichlet form [15]

$$\mathcal{P}(\mathbf{p}; \boldsymbol{\beta}) = \frac{\Gamma(\beta)}{\Gamma(\beta_1) \dots \Gamma(\beta_l)} \prod_{i=1}^{l} p_i^{\beta_i - 1}, \tag{5}$$

where $\boldsymbol{\beta}$ is the vector $\boldsymbol{\beta} = (\beta_1, \dots, \beta_l)$ such that $\beta_i > 0$, $\beta = \sum_{i=1}^{l} \beta_i$ and $l \geq 2$. In binary classification problems the Dirichlet distribution is a beta distribution. The support of the Dirichlet distribution is the l-dimensional probability simplex $\sum_{i=1}^{l} p_i = 1$ and $0 < p_i < 1 \; \forall i$. This particular form of $\mathcal{P}(\mathbf{p})$ is convenient because it is conjugate to the multinomial likelihood $\mathcal{P}(\mathbf{t}|\mathbf{p})$ [15]. The posterior distribution $\mathcal{P}(\mathbf{p}|\mathbf{t})$ (that is, the distribution of $\mathbf{p}$ given that the predictions of t classifiers are known) is also a Dirichlet distribution. Therefore, the estimation of the distribution of $\mathbf{p}$ is refined by updating the parameters of the Dirichlet distribution as the classifier predictions become known.

As in the original SIBP, our goal is to compute $\mathcal{P}^*(\mathbf{t}, T)$, the probability that the ensemble of size T and the sub-ensemble of size t predict the same class label for the instance considered. Before, an expression for this probability is derived, we need some intermediate quantities.

Proposition 1. *Assuming that* $\mathbf{p}$ *follows a Dirichlet distribution with parameters* $\boldsymbol{\beta} = \{\beta_1, \dots, \beta_l\}$ *and that the likelihood* $\mathcal{P}(\mathbf{t}|\mathbf{p})$ *is multinomial , the conditional distribution of* $\mathbf{T} = \{T_1, \dots, T_l\}$

given $\mathbf{t} = \{t_1, \ldots, t_l\}$ *obtained after querying* t *classifiers is*

$$\mathcal{P}(\mathbf{T}|\mathbf{t}) = \frac{(T-t)!}{\prod_{i=1}^{l}(T_i - t_i)!} \frac{\prod_{i=1}^{l}(t_i+\beta_i)_{T_i-ti}}{(t+\beta)_{T-t}}. \tag{6}$$

Proof. Given a probability vector $\mathbf{p}$, the probability distribution of the vector of independent votes $\mathbf{t} = \{t_1, \ldots, t_l\}$ follows a multinomial distribution [15]

$$\mathcal{P}(\mathbf{t}|\mathbf{p}) = \frac{t!}{t_1!t_2!\ldots t_l!} \prod_{i=1}^{l} p_i^{t_i}. \tag{7}$$

The class predicted by the subensemble t is given by $k_{\mathbf{t}} = \arg\max_i t_i$. The posterior probability $\mathcal{P}(\mathbf{p}|\mathbf{t})$ can be obtained by applying Bayes' theorem and considering that the prior $\mathcal{P}(\mathbf{p})$ is of the form (5).

$$\mathcal{P}(\mathbf{p}|\mathbf{t}) \quad = \quad \frac{\mathcal{P}(\mathbf{t}|\mathbf{p})\mathcal{P}(\mathbf{p})}{\mathcal{P}(\mathbf{t})} = \frac{\Gamma(t+\beta)}{\prod_{i=1}^{l}\Gamma(t_i+\beta_i)} \prod_{i=1}^{l} p_i^{t_i+\beta_i-1}, \tag{8}$$

where $\mathcal{P}(\mathbf{t})$, the normalization constant is

$$\begin{aligned}
\mathcal{P}(\mathbf{t}) &= \int_{\mathcal{D}} \mathcal{P}(\mathbf{t}|\mathbf{p})\mathcal{P}(\mathbf{p})d\mathbf{p} = \frac{\Gamma(\beta)}{\Gamma(\beta_1)\ldots\Gamma(\beta_l)} \frac{t!}{t_1!t_2!\ldots t_l!} \int_{\mathcal{D}} \prod_{i=1}^{l} p_i^{t_i+\beta_i-1} d\mathbf{p} \\
&= \frac{\Gamma(\beta)}{\Gamma(\beta_1)\ldots\Gamma(\beta_l)} \frac{t!}{t_1!t_2!\ldots t_l!} \frac{\prod_{i=1}^{l}\Gamma(t_i+\beta_i)}{\Gamma(t+\beta)}.
\end{aligned}$$

Thus, the posterior distribution of $\mathbf{p}$ given the observed vector of votes $\mathbf{t}$ is a Dirichlet distribution of order l and parameters $(t_1+\beta_1, t_2+\beta_2, \ldots, t_l+\beta_l)$ [15].

Since the individual classifiers are independent, given the training data,

$$\mathcal{P}(\mathbf{T}|\mathbf{t}) = \mathcal{P}(\mathbf{T}-\mathbf{t}|\mathbf{t}) \qquad \mathcal{P}(\mathbf{T}-\mathbf{t}|\mathbf{p},\mathbf{t}) = \mathcal{P}(\mathbf{T}-\mathbf{t}|\mathbf{p}).$$

Using these equalities,

$$\begin{aligned}
\mathcal{P}(\mathbf{T}|\mathbf{t}) &= \mathcal{P}(\mathbf{T}-\mathbf{t}|\mathbf{t}) = \int_{\mathcal{D}} \mathcal{P}(\mathbf{T}-\mathbf{t}|\mathbf{p},\mathbf{t})\mathcal{P}(\mathbf{p}|\mathbf{t})d\mathbf{p} = \int_{\mathcal{D}} \mathcal{P}(\mathbf{T}-\mathbf{t}|\mathbf{p})\mathcal{P}(\mathbf{p}|\mathbf{t})d\mathbf{p} \\
&= \frac{(T-t)!}{\prod_{i=1}^{l}(T_i-t_i)!} \frac{\Gamma(t+\beta)}{\prod_{i=1}^{l}\Gamma(t_i+\beta_i)} \int_{\mathcal{D}} \prod_{i=1}^{l} p_i^{T_i+\beta_i-1} d\mathbf{p} \\
&= \frac{(T-t)!}{\prod_{i=1}^{l}(T_i-t_i)!} \frac{\prod_{i=1}^{l}(t_i+\beta_i)_{T_i-t_i}}{(t+\beta)_{T-t}},
\end{aligned}$$

where $(t)_n = t(t+1)\cdots(t+n-1)$ is the Pochhammer symbol [16], t is a non-negative real number and n is a non-negative integer. □

Finally, the probability that the classes predicted by the subensemble of size t and the full ensemble of size T coincide is

$$\mathcal{P}^*(\mathbf{t}, T) = \sum_{\mathbf{T}\in\mathcal{T}_{\mathbf{t}}} \mathcal{P}(\mathbf{T}|\mathbf{t}) = \frac{(T-t)!}{(t+\beta)_{T-t}} \sum_{\mathbf{T}\in\mathcal{T}_{\mathbf{t}}} \frac{\prod_{i=1}^{l}(t_i+\beta_i)_{T_i-t_i}}{\prod_{i=1}^{l}(T_i-t_i)!}. \tag{9}$$

where $\mathcal{T}_{\mathbf{t}}$ is the set of vectors $\mathbf{T}$ such that $k_{\mathbf{T}} = k_{\mathbf{t}}$, $T_i \geq t_i$ and $\sum_{i=1}^{l} T_i = T$. Notice that if the parameters corresponding to the uniform distribution $\beta_i = 1\ \forall i$ and $\beta = l$ are plugged into (9), one recovers (4).

4 Majority voting with mixtures of Dirichlet priors

Using a Dirichlet distribution to model the prior does not always provide an accurate fit. In the datasets investigated $\mathcal{P}(\mathbf{p})$ often has two or more modes which cannot be well approximated with a single Dirichlet distribution. Usually each mode corresponds to the distribution of instances in each class. Therefore, we assume that $\mathcal{P}(\mathbf{p})$ can be well approximated by a mixture of Dirichlet distributions, one per class label

$$\mathcal{P}(\mathbf{p}) = \sum_{j=1}^{l} \mathcal{P}(c_j)\mathcal{P}(\mathbf{p}|c_j) = \sum_{j=1}^{l} w_j \frac{\Gamma(\beta^j)}{\Gamma(\beta_1^j)\ldots\Gamma(\beta_l^j)} \prod_{i=1}^{l} p_i^{\beta_i^j - 1}, \tag{10}$$

where each Dirichlet is weighted by w_j, the class prior. These weights are the frequency of each class $w_j = \frac{n_j}{N}$, where $N = \sum_{j=1}^{l} n_j$. The parameters of each of the components is estimated using only examples of the corresponding class.

Proposition 2. *Assuming that the distribution of* $\mathbf{p}$ *is a Dirichlet mixture* (10)*, and that the likelihood* $\mathcal{P}(\mathbf{t}|\mathbf{p})$ *is multinomial, the conditional distribution of* $\mathbf{T} = \{T_1, \ldots, T_l\}$ *given* $\mathbf{t} = \{t_1, \ldots, t_l\}$ *obtained after querying* t *classifiers is*

$$\mathcal{P}(\mathbf{T}|\mathbf{t}) = \frac{(T-t)!}{\prod_{j=1}^{l}(T_j - t_j)!} \frac{\sum_{i=1}^{l} \frac{w_i}{(\beta^i)_T} \prod_{j=1}^{l} (\beta_j^i)_{T_j}}{\sum_{j=1}^{l} \frac{w_j}{(\beta^j)_t} \prod_{i=1}^{l} (\beta_i^j)_{t_i}}. \tag{11}$$

Proof. The conditional probability distribution $\mathcal{P}(\mathbf{t}|\mathbf{p})$ follows a multinomial distribution (see Eq. (7)). To apply Bayes' theorem, we first need to compute the normalization constant $\mathcal{P}(\mathbf{t})$:

$$\begin{aligned} \mathcal{P}(\mathbf{t}) &= \int_{\mathcal{D}} \mathcal{P}(\mathbf{t}|\mathbf{p})\mathcal{P}(\mathbf{p})d\mathbf{p} = \frac{t!}{t_1!\ldots t_l!} \sum_{j=1}^{l} w_j \frac{\Gamma(\beta^j)}{\Gamma(\beta_1^j)\ldots\Gamma(\beta_l^j)} \int_D \prod_{i=1}^{l} p_i^{t_i+\beta_i^j-1} d\mathbf{p} \\ &= \frac{t!}{t_1!\ldots t_l!} \sum_{j=1}^{l} w_j \frac{\Gamma(\beta^j)}{\Gamma(t+\beta^j)} \prod_{i=1}^{l} \frac{\Gamma(t_i+\beta_i^j)}{\Gamma(\beta_i^j)} = \frac{t!}{t_1!\ldots t_l!} \sum_{j=1}^{l} \frac{w_j}{(\beta^j)_t} \prod_{i=1}^{l} (\beta_i^j)_{t_i}. \end{aligned} \tag{12}$$

The posterior distribution $\mathcal{P}(\mathbf{p}|\mathbf{t})$ is computed applying Bayes' Theorem. Unlike in the previous case, the numerator and denominator do not share common factors and the resulting expression is more complex

$$\mathcal{P}(\mathbf{p}|\mathbf{t}) = \frac{\mathcal{P}(\mathbf{t}|\mathbf{p})\mathcal{P}(\mathbf{p})}{\mathcal{P}(\mathbf{t})} = \frac{\sum_{j=1}^{l} w_j \Gamma(\beta^j) \prod_{i=1}^{l} \frac{p_i^{t_i+\beta_i^j-1}}{\Gamma(\beta_i^j)}}{\sum_{j=1}^{l} \frac{w_j}{(\beta^j)_t} \prod_{i=1}^{l} (\beta_i^j)_{t_i}}. \tag{13}$$

Finally, the probability of vector $\mathbf{T}$ given vector $\mathbf{t}$, or $\mathcal{P}(\mathbf{T}|\mathbf{t})$ is

$$\begin{aligned} \mathcal{P}(\mathbf{T}|\mathbf{t}) &= \int_{\mathcal{D}} \mathcal{P}(\mathbf{T}-\mathbf{t}|\mathbf{p})\mathcal{P}(\mathbf{p}|\mathbf{t})d\mathbf{p} = \frac{(T-t)!}{\prod_{i=1}^{l}(T_i - t_i)!} \frac{\sum_{j=1}^{l} w_j \Gamma(\beta^j) \int_D \prod_{i=1}^{l} \frac{p_i^{T_i+\beta_i^j-1}}{\Gamma(\beta_i^j)} d\mathbf{p}}{\sum_{j=1}^{l} \frac{w_j}{(\beta^j)_t} \prod_{i=1}^{l} (\beta_i^j)_{t_i}} \\ &= \frac{(T-t)!}{\prod_{j=1}^{l}(T_j - t_j)!} \frac{\sum_{i=1}^{l} \frac{w_i}{(\beta^i)_T} \prod_{j=1}^{l} (\beta_j^i)_{T_j}}{\sum_{j=1}^{l} \frac{w_j}{(\beta^j)_t} \prod_{i=1}^{l} (\beta_i^j)_{t_i}}. \end{aligned}$$

□

The probability that the class predicted by the complete ensemble and the subensemble of size t is

$$\mathcal{P}^*(\mathbf{t}, T) = \sum_{\mathbf{T}\in\mathcal{T}_\mathbf{t}} \frac{(T-t)!}{\prod_{j=1}^{l}(T_j - t_j)!} \frac{\sum_{i=1}^{l} \frac{w_i}{(\beta^i)_T} \prod_{j=1}^{l} (\beta_j^i)_{T_j}}{\sum_{j=1}^{l} \frac{w_j}{(\beta^j)_t} \prod_{i=1}^{l} (\beta_i^j)_{t_i}}, \tag{14}$$

where vectors $\mathbf{T} \in \mathcal{T}_\mathbf{t}$ satisfy $k_\mathbf{T} = k_\mathbf{t}$, $T_i \geq t_i$ and $\sum_{i=1}^{l} T_i = T$.

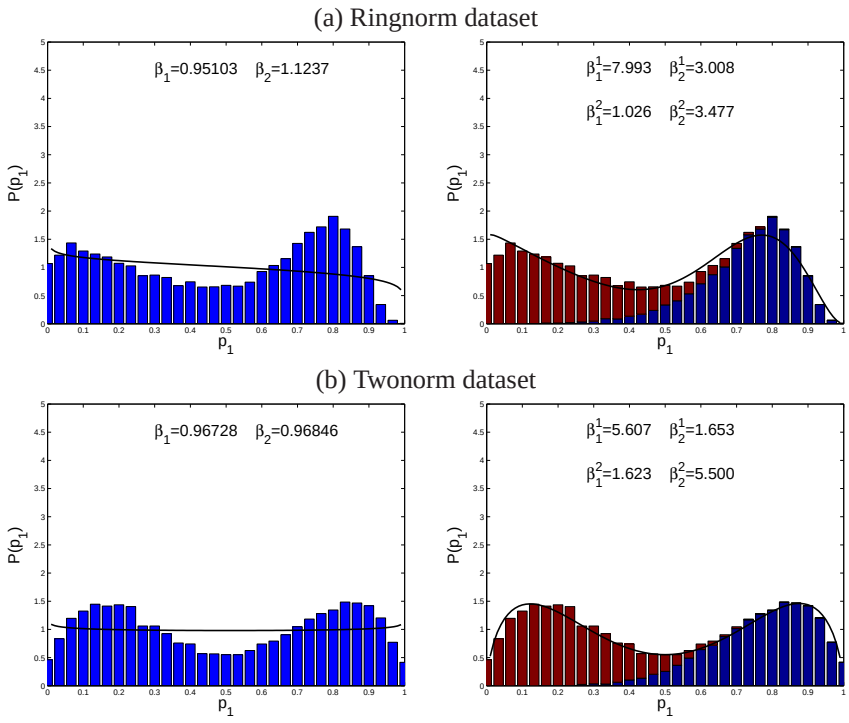

Figure 1: Prior Distributions for datasets (a) Ringnorm and (b) Twonorm. The left column displays the fitted Dirichlet prior and the left column the fitted Dirichlet mixture.

5 Experiments

In this section we present the results of dynamical pruning in random forests for a series of benchmark classification problems from the UCI Repository [17] and the synthetic problems described in [18]. These results are used to illustrate the consistency of the analysis of majority voting presented in the previous section and the improvements that are obtained using problem-specific information for the estimate of $\mathcal{P}(\mathbf{p})$. In particular, we compare the performance of the full ensemble and the dynamically pruned ensembles using uniform, Dirichlet or Dirichlet mixture priors. The parameters of the model for $\mathcal{P}(\mathbf{p})$ are determined using all of the out-of-bag data for the Dirichlet prior, or using the examples from each class in the out-of-bag data for the corresponding component in the Dirichlet mixture. The calibration method used is described in [19]. It has been implemented using the FastFit Matlab ToolBox [20]. The protocol for the experiments is as follows. For each problem, 100 partitions are created, and for each partition a random forest ensemble of size $T = 101$ is built. We then compute the mean error rate of the complete ensemble and record the mean number of trees that are queried to determine the final prediction. Note that this number need not be T: the voting process can be halted when the remaining votes (i.e. the number of classifiers whose prediction is unknown) are insufficient to alter the global ensemble decision. This is the case when the number of remaining votes is lower than the difference between the majority class and the second most voted class. SIBP is then applied to dynamically prune using the stopping rule derived from assuming different priors: uniform, Dirichlet and mixture of Dirichlet distributions. The ensembles are queried until the confidence on the ensemble decision is $\alpha = 0.99$. At that point the generalization error rate, number of queried trees and the disagreement rates between the predictions of the SIB-pruned ensembles and of the complete ensembles are computed.

Figure 1 displays the estimations of the prior distributions using a Dirichlet model (left) and a mixture of Dirichlet distributions (right). On the right plots that display the fit by a mixture of Dirichlet distributions, the fraction of examples from each class is shown in a different color. Prior distributions for the rest of the problems analyzed are included in the supplementary material. Tables 1, 2 and 3 show the average error rates, disagreement rates and number of queried trees for all the tested classification problems, respectively. The tables display the results of the complete ensem-

Problem	RF	SIBP	1DIR	NDIR
australian	**13.00±3.7**	13.09±3.7	13.14±3.7	13.14±3.7
breast	**3.22±2.1**	3.23±2.1	3.40±2.3	3.45±2.2
diabetes	24.34±4.2	24.25±4.1	24.31±4.1	**24.21±4.0**
echocardiogram	22.18±14.3	**22.05±14.7**	22.46±14.6	22.18±14.4
german	**23.43±3.5**	23.65±3.3	23.60±3.3	23.61±3.3
heart	**18.30±6.9**	18.37±7.0	18.44±7.0	18.44±7.1
horse-colic	15.47±5.6	**15.44±5.4**	15.47±5.4	15.49±5.4
ionosphere	**6.44±4.1**	**6.44±4.1**	6.50±4.0	6.55±4.0
labor	6.33±8.9	**6.17±8.8**	6.33±8.9	6.33±8.9
liver	27.10±6.7	27.09±7.0	27.04±6.9	**27.01±6.9**
mushroom	**0.00±0.0**	**0.00±0.0**	**0.00±0.0**	0.08±0.2
new-thyroid	**4.29±4.0**	4.38±4.0	4.61±4.3	4.94±4.5
ringnorm	**7.60±1.3**	7.72±1.2	7.74±1.2	7.78±1.2
sonar	**16.25±8.7**	16.45±8.7	16.40±8.7	16.50±8.7
spam	**4.59±1.5**	4.63±1.5	4.72±1.5	4.79±1.5
threenorm	**17.85±1.1**	18.04±1.1	17.94±1.1	17.95±1.1
tic-tac-toe	**1.05±1.1**	1.16±1.1	1.20±1.2	1.34±1.3
twonorm	**4.66±0.6**	4.77±0.6	4.76±0.6	4.90±0.6
votes	**4.05±2.9**	4.12±2.9	4.23±2.9	4.23±2.9
waveform	**17.30±0.9**	17.36±0.8	17.65±0.8	17.46±0.8
wine	**1.69±2.8**	1.74±2.8	2.19±3.2	2.24±3.4

Table 1: Error rates. For each dataset the best method is highlighted in boldface.

Problem	SIBP	1DIR	NDIR
australian	0.3±0.6	0.5±0.8	0.4±0.8
breast	0.1±0.4	0.3±0.7	0.5±0.9
diabetes	0.6±0.9	0.8±1.1	0.7±1.0
echocardiogram	0.7±3.1	0.8±3.3	0.8±3.3
german	0.8±0.8	0.9±0.9	0.8±0.8
heart	0.8±1.8	1.0±1.9	1.0±2.0
horse-colic	0.4±0.9	0.4±0.9	0.5±1.0
ionosphere	0.1±0.6	0.2±0.7	0.3±0.9
labor	0.2±1.7	0.3±2.3	0.3±2.3
liver	1.0±1.7	0.8±1.4	0.8±1.4
mushroom	0.0±0.0	0.0±0.0	0.1±0.2
new-thyroid	0.1±0.7	0.5±1.5	0.9±2.2
ringnorm	0.5±0.2	0.5±0.2	0.6±0.2
sonar	0.9±2.0	0.6±1.8	0.8±1.9
spam	0.1±0.2	0.4±0.3	0.5±0.4
threenorm	1.0±0.2	0.7±0.1	0.7±0.2
tic-tac-toe	0.1±0.4	0.2±0.4	0.3±0.6
twonorm	0.4±0.1	0.4±0.1	0.7±0.2
votes	0.1±0.4	0.4±1.1	0.5±1.2
waveform	0.6±0.1	1.7±0.4	1.0±0.2
wine	0.1±0.6	0.8±2.1	0.9±2.3

Table 2: Disagreement rates comparison of the SIBP variants

ble (RF), the dynamically pruned ensembles using the halting rule derived from assuming uniform priors (SIBP), using a Dirichlet prior (1DIR) and using a mixture of Dirichlets (NDIR). The results are displayed as *mean* $\pm$ *std. deviation* over the 100 partitions of the data. The best results for each problem are marked with bold fonts. From table 1 one can see that the mean error rates obtained by all pruning methods are very similar to the original Random Forests. Although for some problems the mean error rates are slightly worse than the rates obtained by the complete ensemble, the difference between them is never greater than 0.3 percentage points.

Problem	RF	SIBP	1DIR	NDIR
australian	62.2±1.4	16.1±2.1	**14.6±2.2**	14.7±2.1
breast	54.2±0.9	8.9±1.4	5.8±1.3	**5.1±1.1**
diabetes	68.8±1.8	24.9±3.2	**23.7±3.3**	24.5±3.3
echocardiogram	68.0±4.6	22.6±8.2	**21.5±8.1**	21.8±8.2
german	71.8±1.3	28.4±2.8	**27.3±2.8**	28.1±2.8
heart	67.2±2.5	22.5±4.2	21.7±4.3	**21.4±4.2**
horse-colic	66.2±2.1	20.2±3.5	19.6±3.5	**19.0±3.5**
ionosphere	57.9±1.5	11.9±2.3	9.9±2.3	**9.3±2.3**
labor	61.6±4.0	14.1±6.0	12.5±6.0	**12.1±6.0**
liver	74.5±2.3	**31.8±4.5**	32.6±4.5	32.5±4.5
mushroom	51.0±0.0	6.0±0.0	3.0±0.0	**1.0±0.0**
new-thyroid	55.2±1.8	10.7±2.6	6.2±2.4	**5.4±2.2**
ringnorm	68.6±0.8	22.9±1.1	22.6±1.3	**21.3±1.4**
sonar	73.9±3.0	**32.1±6.6**	32.7±6.6	**32.1±6.4**
spam	57.1±0.3	11.1±0.5	8.5±0.6	**7.9±0.7**
threenorm	76.6±0.5	**34.8±1.0**	36.6±1.1	36.3±1.1
tic-tac-toe	60.7±0.9	12.8±1.4	11.1±1.3	**9.4±1.3**
twonorm	67.2±0.2	21.0±0.5	20.9±0.5	**18.6±0.6**
votes	54.5±1.2	8.8±1.8	5.8±1.7	**5.1±1.6**
waveform	72.3±0.7	29.3±1.1	**24.9±1.1**	26.9±1.2
wine	57.3±2.1	11.4±2.7	6.7±2.4	**6.1±2.2**

Table 3: Number of queried trees. For each dataset the best method is highlighted in boldface.

From the results displayed in table 2, one can see the disagreement rates obtained by the Dirichlet and the Dirichlet mixture priors are closer to the specified target ($1 - \alpha = 1\%$) than the disagreement rates obtained when a uniform prior is assumed, except for *liver*, *sonar* and *threenorm*. In these problems the prior distributions over $\mathbf{p}$ (see supplemental material) are close to a uniform distribution and the disagreement for the original SIBP method is very close to the expected one. Nonetheless, the disagreement rates of the proposed methods do not divert significantly from the expected target $1 - \alpha = 1\%$. In general, the mixture of Dirichlet prior obtains better results than the single Dirichlet prior because it provides a better fit, specially when the the actual distribution is multi-modal. Correspondingly, using a more flexible model for the prior improves the pruning rates in all problems except for *liver*, *sonar* and *threenorm* where the differences are small. The best pruning rates correspond to the mixture of Dirichlet model followed by the single Dirichlet and, finally, by the uniform model (see table 3).

6 Conclusions

In dynamical ensemble pruning the number of classifier predictions that are computed is determined adaptively, depending on the instance that is being classified. In *Statistical Instance-Based Pruning* (SIPB) the classifiers of the ensemble are queried sequentially until the probability that the class predicted by majority voting up to that point will not change with the remaining predictions is above a specified confidence level α. In its original formulation the rule used to determine when the voting process should be halted was derived assuming a uniform prior for the class prediction probabilities. As a result, conservative pruning rates were obtained in most of the classification problems analyzed. Higher pruning rates can be obtained if problem-specific information is incorporated in the derivation of the halting rule. In this work we propose to use Dirichlet or mixtures of Dirichlet priors. A theoretical analysis of majority voting starting from these non-unform priors is carried out. The parameters of the prior distributions are determined using out-of-bag data. As a result of using problem-specific knowledge, the differences between the predictions dynamically pruned ensemble and the complete ensembles becoome closer to the specified target $1 - \alpha$. This results in faster classification speed without a significant deterioration of the predictive accuracy with respect to the complete ensemble. The halting rule derived using mixture of Dirichlet priors obtains the best overall results because of its capacity to approximate the actual class prediction probabilities in the problems analyzed.

References

[1] Lars Kai Hansen and Peter Salamon. Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:993–1001, 1990.

[2] Tin Kam Ho, Jonathan J. Hull, and Sargur N. Srihari. Decision combination in multiple classifier systems. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 16(1):66–75, 1994.

[3] Thomas G. Dietterich. Ensemble methods in machine learning. In *Multiple Classifier Systems: First International Workshop*, pages 1–15, 2000.

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

[5] Ludmila I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 2004.

[6] Robert E. Banfield, Lawrence O. Hall, Kevin W. Bowyer, and W. Philip Kegelmeyer. A comparison of decision tree ensemble creation techniques. *IEEE Transanctions on Pattern Analysis and Machine Intelligence*, 29(1):173–180, 2007.

[7] Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[8] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[9] Juan José Rodríguez, Ludmila I. Kuncheva, and Carlos J. Alonso. Rotation forest: A new classifier ensemble method. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(10):1619–1630, 2006.

[10] Dragos D. Margineantu and Thomas G. Dietterich. Pruning adaptive boosting. In *Proc. of the 14th International Conference on Machine Learning*, pages 211–218. Morgan Kaufmann, 1997.

[11] Yi Zhang, Samuel Burer, and W. Nick Street. Ensemble pruning via semi-definite programming. *Journal of Machine Learning Research*, 7:1315–1338, 2006.

[12] Wei Fan, Fang Chu, Haixun Wang, and Philip S. Yu. Pruning and dynamic scheduling of cost-sensitive ensembles. In *Proc. of the 18th National Conference on Artificial Intelligence*, pages 146–151. American Association for Artificial Intelligence, 2002.

[13] Haixun Wang, Wei Fan, Philip S. Yu, and Jiawei Han. Mining concept-drifting data streams using ensemble classifiers. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 226–235, New York, NY, USA, 2003. ACM Press.

[14] Daniel Hernández-Lobato, Gonzalo Martínez-Muñoz, and Alberto Suárez. Statistical instance-based pruning in ensembles of independent classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):364–369, 2009.

[15] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 1st ed. 2006. corr. 2nd printing edition, October 2007.

[16] Milton Abramowitz and Irene A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover, New York, ninth dover printing, tenth gpo printing edition, 1964.

[17] A. Asuncion and D.J. Newman. UCI machine learning repository, 2007.

[18] Leo Breiman. Bias, variance, and arcing classifiers. Technical Report 460, Statistics Department, University of California, 1996.

[19] Thomas Minka. Estimating a dirichlet distribution. Website, 2003.

[20] Thomas Minka. Fastfit matlab toolbox. Website.