Bernoulli Random Forests: Closing the Gap between Theoretical Consistency and Empirical Soundness

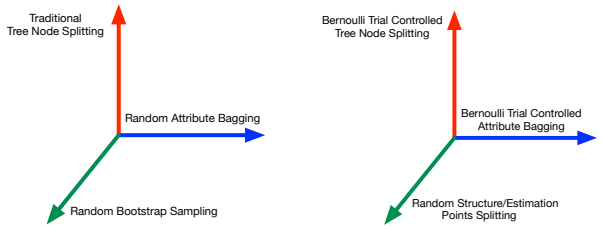
Yisen Wang[†,‡], **Qingtao Tang**[†,‡], **Shu-Tao Xia**[†,‡], **Jia Wu** [⋆], **Xingquan Zhu** [◇]
[†] Dept. of Computer Science and Technology, Tsinghua University, China
[‡] Graduate School at Shenzhen, Tsinghua University, China
[⋆] Quantum Computation & Intelligent Systems Centre, University of Technology Sydney, Australia
[◇] Dept. of Computer & Electrical Engineering and Computer Science, Florida Atlantic University, USA
{wangys14, tqt15}@mails.tsinghua.edu.cn; xiast@sz.tsinghua.edu.cn; jia.wu@uts.edu.au; xzhu3@fau.edu

Abstract

Random forests are one type of the most effective ensemble learning methods. In spite of their sound empirical performance, the study on their theoretical properties has been left far behind. Recently, several random forests variants with nice theoretical basis have been proposed, but they all suffer from poor empirical performance. In this paper, we propose a Bernoulli random forests model (BRF), which intends to close the gap between the theoretical consistency and the empirical soundness of random forests classification. Compared to Breiman's original random forests, BRF makes two simplifications in tree construction by using two independent Bernoulli distributions. The first Bernoulli distribution is used to control the selection of candidate attributes for each node of the tree, and the second one controls the splitting point used by each node. As a result, BRF enjoys proved theoretical consistency, so its accuracy will converge to optimum (*i.e.*, the Bayes risk) as the training data grow infinitely large. Empirically, BRF demonstrates the best performance among all theoretical random forests, and is very comparable to Breiman's original random forests (which do not have the proved consistency yet). The theoretical and experimental studies advance the research one step further towards closing the gap between the theory and the practical performance of random forests classification.

1 Introduction

Random forests (RF) represent a class of ensemble learning methods that construct a large number of randomized decision trees and combine results from all trees for classification or regression. The training of the random forests, at least for the original Breiman version [Breiman, 2001], is extremely easy and efficient. Because predictions are derived from a large number of trees, random forests are very robust, and have demonstrated superb empirical performance for many real-world learning tasks (*e.g.*, [Svetnik *et al.*, 2003; Prasad *et al.*, 2006; Cutler *et al.*, 2007; Shotton *et al.*, 2011; Xiong *et al.*, 2012]).



(a) Breiman RF (b) BRF

Figure 1: Comparisons between Breiman RF (left panel) *vs.* the proposed BRF (right panel). The tree node splitting of Breiman RF is deterministic, so the final trees are highly data-dependent. Instead, BRF employs two Bernoulli distributions to control the tree construction. BRF ensures a certain degree of randomness in the trees, so the final trees are less data-dependent but still maintain a high classification accuracy.

Different from their well recognized empirical reputation across numerous domain applications, the theoretical properties of random forests have yet been fully established and are still under active research investigation. Particularly, one of the most fundamental theoretical properties of a learning algorithm is the *consistency*, which guarantees that the algorithm will converge to optimum (*i.e.* the Bayes risk) as the data grow infinitely large. Due to the inherent bootstrap randomization as well as attribute bagging process employed by random forests, and the highly data-dependent tree structure, it is very difficult to prove the theoretical consistency of random forests. As shown in Fig. 1(a), Breiman's original random forests have two random processes, bootstrap instance sampling and attribute bagging, which intend to make the tree less data-dependent. However, when constructing each internal node of the tree, the attribute selected for splitting and the splitting point used by the attribute are controlled by a deterministic criterion (such as Gini index [Breiman *et al.*, 1984]). As a result, the constructed trees eventually become data-dependent, making theoretical analysis difficult.

Notice the difficulty of proving the consistency of random forests, several random forests variants [Breiman, 2004; Biau *et al.*, 2008; Genuer, 2010; 2012; Biau, 2012; Denil *et*

al., 2014] relax or simplify the deterministic tree construction process by employing two randomized approaches: (1) replacing the attribute bagging by randomly sampling one attribute; or (2) splitting the tree node with a more elementary split protocol in place of the complicated impurity criterion. For both approaches, the objective is to make the tree less data-dependent, such that the consistency analysis can be carried out. Unfortunately, although the properties of such approaches can be theoretically analyzed, they all suffer from poor empirical performance, and are mostly incomparable to Breiman's original random forests in terms of the classification accuracies. The dilemma of the theoretical consistency *vs.* empirical soundness persists and continuously motivates the research to close the gap.

Motivated by the above observations, in this paper, we propose a novel Bernoulli random forests model (BRF) with proved theoretical consistency and comparable performance to Breiman's original random forests. As shown in Fig. 1(b), our key is to introduce two Bernoulli distributions to drive/control the tree construction process, so we can ensure a certain degree of randomness in the trees, but still keep the quality of the tree for classification. Because each Bernoulli trial involves a probability value controlled random process, BRF can ensure that the tree construction is random with respect to a probability value or being deterministic, otherwise. As a result, the trees constructed by BRF are much less data-dependent, compared to Breiman's original random forests, yet still have much better performance compared to all existing theoretically consistent random forests.

The main contribution of the paper is threefold: (1) BRF has the least simplification changes, compared to Breiman's original random forests, and its theoretical consistency is fully proved; (2) The two independent Bernoulli distributions controlled attribute and splitting point selection process provides a solution to resolve the dilemma of theoretical consistency *vs.* empirical soundness; and (3) empirically, a series of experimental comparisons demonstrate that BRF achieves the best performance among all theoretical random forests.

2 Bernoulli Random Forests

Compared to Breiman's original random forests, the proposed BRF makes three alterations, as shown in Fig. 1, to close the gap between theoretical consistency and empirical soundness.

2.1 Data point partitioning

Given a data set $\mathcal{D}_n$ with n instances, for each instance $(\boldsymbol{X}, Y)$, we have $\boldsymbol{X} \in \mathbb{R}^D$ with D being the number of attributes and $Y \in \{1, 2, \ldots, C\}$ with C being the number of classes. Before the construction of each individual tree, we randomly partition the entire data points into two parts, *i.e.,* **Structure points** and **Estimation points**. The two parts perform different roles in the individual tree construction, which helps establish the consistency property of the proposed BRF.

Structure points are used to construct the tree. They are only used to determine the attributes and splitting points in each internal node of the tree, but are not allowed to be used for estimating class labels in tree leaves.

Estimation points are only used to fit leaf predictors (estimating class labels in tree leaves). Note that these points are also split obeying the rules created by structure points along the construction of the tree, but they have no effect on the structure of the tree.

For each tree, the points are partitioned randomly and independently. The ratio of the two parts is parameterized by $Ratio = |\text{Structure points}|/|\text{Entire points}|$.

2.2 Tree construction

In the proposed BRF, firstly, as stated above, the data point partitioning process replaces the bootstrap technique in training instances. Secondly, two independent Bernoulli distributions are introduced into the strategies of selecting attributes and splitting points.

The first novel alteration in BRF is to choose candidate attributes with a probability satisfying a Bernoulli distribution. Let $B_1 \in \{0, 1\}$ be a binary random variable with "success" probability of p_1, then B_1 has a Bernoulli distribution which takes 1 in a probability of p_1. We define $B_1 = 1$ if 1 candidate attribute is chosen and $B_1 = 0$ if $\sqrt{D}$ candidate attributes are chosen. To be specific, for each internal node, we choose 1 or $\sqrt{D}$ candidate attributes in a probability of p_1 or $1 - p_1$ respectively.

The second novel alteration is to choose splitting points using two different methods with a probability satisfying another Bernoulli distribution. Similar to B_1, we assume $B_2 \in \{0, 1\}$ satisfies another Bernoulli distribution which takes 1 in a probability of p_2. We define $B_2 = 1$ if the random sampling method is used and $B_2 = 0$ if the optimizing impurity criterion method is used. Specifically, we choose the splitting point through random sampling or optimizing impurity criterion in a probability of p_2 or $1 - p_2$ respectively in each candidate attribute.

The impurity criterion is denoted by:

$$I(v) = T(\mathcal{D}^S) - \frac{|\mathcal{D}'^S|}{|\mathcal{D}^S|} T(\mathcal{D}'^S) - \frac{|\mathcal{D}''^S|}{|\mathcal{D}^S|} T(\mathcal{D}''^S). \quad (1)$$

Here v is the splitting point which is chosen by maximizing $I(v)$. $\mathcal{D}$ is the cell belonging to the node to be split, which contains structure points $\mathcal{D}^S$ and estimation points $\mathcal{D}^E$. $\mathcal{D}'$ and $\mathcal{D}''$ are two children that would be created if $\mathcal{D}$ is split at v. The function $T(\mathcal{D}^S)$ is the impurity criterion, *e.g.* Shannon entropy or Gini index, which computes over the labels of the structure points $\mathcal{D}^S$. In BRF, the impurity criterion is Gini index and it is certain that Shannon entropy is also acceptable.

Through the above two steps, for each internal node of the tree, one attribute and its corresponding splitting point is selected to split the data and grow the tree. Note that only the structure points are involved in the tree construction. The process recursively repeats until the stopping condition is met.

Similar to Breiman's original random forests, BRF's stopping condition is also based on the minimum leaf size. But, we only restrict the estimation points rather than the entire data points. In other words, in each leaf, we require that the number of estimation points must be larger than k_n which depends on the number of training instances, *i.e.*, $k_n \to \infty$ and $k_n/n \to 0$ when $n \to \infty$.

2.3 Prediction

Once the trees are trained by structure points and the leaf predictors are fitted by estimation points, BRF can be used to predict labels of new unlabeled instances.

When making predictions, each individual tree will predict independently. We denote a base decision tree classifier created by our algorithm as g. Assuming the unlabeled instance is $\boldsymbol{x}$, the probability of each class $c \in \{1, 2, \ldots, C\}$ is

$$\eta^{(c)}(\boldsymbol{x}) = \frac{1}{N(A^E(\boldsymbol{x}))} \sum_{(\boldsymbol{X},Y)\in A^E(\boldsymbol{x})} \mathbb{I}\{Y = c\}, \quad (2)$$

and the prediction of the tree is the class that maximizes $\eta^c(\boldsymbol{x})$:

$$\hat{y} = g(\boldsymbol{x}) = \arg\max_c \{\eta^{(c)}(\boldsymbol{x})\}, \quad (3)$$

where $N(A^E(\boldsymbol{x}))$ denotes the number of estimation points in the leaf containing the instance $\boldsymbol{x}$. $\mathbb{I}(e)$ is the indicator function which takes 1 if e is true and takes 0 for other cases.

Second, the prediction of the forests is the class which receives the most votes from individual trees.

$$\overline{\hat{y}} = \overline{g^{(M)}(\boldsymbol{x})} = \arg\max_c \sum_{i=1}^{M} \mathbb{I}\left\{g^{(i)}(\boldsymbol{x}) = c\right\}, \quad (4)$$

where M is the number of individual trees in the random forests.

3 Consistency

In this section, we first give an outline of the lemmas for establishing the consistency of random forests. Then we prove the consistency of the proposed BRF. We use a variable Z to denote the randomness involved in the construction of the tree, including the selection of attributes and splitting points.

3.1 Preliminaries

Definition 1. *Given the data set $\mathcal{D}_n$, a sequence of classifiers $\{g\}$ are consistent for a certain distribution of $(\boldsymbol{X}, Y)$ if the error probability L satisfies*

$$\mathbb{E}[L] = \mathbb{P}(g(\boldsymbol{X}, Z, \mathcal{D}_n) \neq Y) \to L^* \quad as \quad n \to \infty, \quad (5)$$

where L^* is the Bayes risk that is the minimum achievable risk of any classifier for the distribution of $(\boldsymbol{X}, Y)$.

The consistency of random forests is implied by the consistency of the trees they are comprised of, which will be shown in the following two lemmas.

Lemma 1. *Suppose a sequence of classifiers $\{g\}$ are consistent, then the voting classifier $\overline{g^{(M)}}$ obtained by taking the majority vote over M copies of g with different randomizing variables is also consistent.*

Lemma 2. *Suppose each class posterior estimation is $\eta^{(c)}(\boldsymbol{x}) = \mathbb{P}(Y = c \,|\, \boldsymbol{X} = \boldsymbol{x})$, and that these estimations are each consistent. The classifier*

$$g(\boldsymbol{x}) = \arg\max_c \{\eta^{(c)}(\boldsymbol{x})\} \quad (6)$$

is consistent for the corresponding multi-class classification problem.

Lemma 1 shows that the consistency of random forests is determined by the individual trees [Biau *et al.*, 2008]. Lemma 2 allows us to reduce the consistency of multi-class classifiers to the consistency of posterior estimates for each class [Denil *et al.*, 2013].

Lemma 3. *Suppose a sequence of classifiers $\{g\}$ are conditionally consistent for a specified distribution on $(\boldsymbol{X}, Y)$, i.e.*

$$\mathbb{P}(g(\boldsymbol{X}, Z, I) \neq Y \,|\, I) \to L^*, \quad (7)$$

where I represents the randomness in the data point partitioning. If the random partitioning produces acceptable structure and estimation parts with probability 1, then $\{g\}$ are unconditionally consistent, i.e.

$$\mathbb{P}(g(\boldsymbol{X}, Z, I) \neq Y) \to L^*. \quad (8)$$

Lemma 3 shows that data point partitioning procedure of the tree construction almost would not affect the consistency of the base decision tree [Denil *et al.*, 2013].

To prove the consistency of the base decision tree, we employ a general consistency lemma used for decision rules as follows:

Lemma 4. *Consider a classification rule which builds a prediction by averaging the labels in each leaf node, if the labels of the voting data do not influence the structure of the classification rule then*

$$\mathbb{E}[L] \to L^* \quad as \quad n \to \infty \quad (9)$$

provided that

1. $\mathrm{diam}(A(\boldsymbol{X})) \to 0$ *in probability,*
2. $N(A^E(\boldsymbol{X})) \to \infty$ *in probability,*

where $A(\boldsymbol{X})$ denotes the leaf containing $\boldsymbol{X}$ and $N(A^E(\boldsymbol{X}))$ denotes the number of estimation points in $A(\boldsymbol{X})$.

Generally, the construction of decision trees can be viewed as a partitioning of the original instance space. Thus, each node of the tree corresponds to a rectangular subset/cell of $\mathbb{R}^D$ and the tree root corresponds to all of $\mathbb{R}^D$. Therefore, $\mathrm{diam}(A(\boldsymbol{X})) \to 0$ is equal to that the size of hypercube $A(\boldsymbol{X})$ is close to 0.

Lemma 4 shows that the consistency of the tree construction can be proved on condition that the hypercubes/cells belonging to leaves are sufficiently small but contain infinite number of estimation points [Devroye *et al.*, 2013].

3.2 Consistency theorem

With these preliminary results in hand, we are equipped to prove the main consistency theorem.

Theorem 1. *Suppose that $\boldsymbol{X}$ is supported on $[0, 1]^D$ and has non-zero density almost everywhere. Moreover, the cumulative distribution function (CDF) of the splitting points is right-continuous at 0 and left-continuous at 1. Then BRF is consistent provided that $k_n \to \infty$ and $k_n/n \to 0$ as $n \to \infty$.*

According to Subsection 3.1, we know that the consistency of random forests is determined by the consistency of the base decision tree (Lemma 1 and 2), which is further dependent on the consistency of the tree construction (Lemma 3 and 4). More specifically, the proof of Theorem 1 is to prove the two conditions of Lemma 4, *i.e.,* $\mathrm{diam}(A(\boldsymbol{X})) \to 0$ and $N(A^E(\boldsymbol{X})) \to \infty$ in probability.

Proof. Firstly, since BRF requires $N(A^E(\boldsymbol{X})) \geq k_n$, $N(A^E(\boldsymbol{X})) \to \infty$ is trivial when $n \to \infty$.

Secondly, we prove $\mathrm{diam}(A(\boldsymbol{X})) \to 0$ in probability. Let $V(d)$ be the size of the d-th attribute of $A(\boldsymbol{X})$. It suffices to show that $\mathbb{E}[V(d)] \to 0$ for all $d \in \{1, 2, \ldots, D\}$.

For any given d, the largest size of the child node for the d-th attribute is denoted by $V^*(d)$. Recalling that the splitting point is chosen either by randomly sampling in $[0, 1]$ with a probability p_2 or by optimizing the impurity criterion with a probability $1 - p_2$, we have

$$\begin{aligned}\mathbb{E}[V^*(d)] &\leq (1-p_2) \times 1 + p_2 \times \mathbb{E}[\max(U, 1-U)] \\ &= (1-p_2) \times 1 + p_2 \times \frac{3}{4} = 1 - \frac{1}{4}p_2, \end{aligned} \tag{10}$$

where U is the splitting point for random sampling method and $U \sim Uniform[0, 1]$.

Revisiting that 1 or $\sqrt{D}$ candidate attributes are selected with a probability p_1 or $1 - p_1$ respectively, we define the following events:

$$\begin{aligned} E_1 &= \{\text{One candidate attribute is split}\} \\ E_2 &= \{\text{The splitting attribute is exactly the } d\text{-th one}\} \end{aligned}$$

Denoting the size of the child node for the d-th attribute by $V'(d)$, then

$$\begin{aligned}\mathbb{E}[V'(d)] &= \mathbb{P}(E_1)\,\mathbb{E}[V'(d)|E_1] + \mathbb{P}(\bar{E_1})\,\mathbb{E}[V'(d)|\bar{E_1}] \\ &\leq p_1 \times \mathbb{E}[V'(d)|E_1] + (1-p_1) \times 1 \\ &= p_1 \times \big(\mathbb{P}(E_2|E_1)\,\mathbb{E}[V'(d)|E_1, E_2] \\ &\quad + \mathbb{P}(\bar{E_2}|E_1)\,\mathbb{E}[V'(d)|E_1, \bar{E_2}]\big) + (1-p_1) \\ &\leq p_1 \times \big(\frac{1}{D}\mathbb{E}[V^*(d)] + 1 - \frac{1}{D}\big) + (1-p_1) \\ &\leq 1 - \frac{p_1 p_2}{4D}. \end{aligned} \tag{11}$$

Assuming K is the distance from the tree root to a leaf and iterating (11) after K splits, we have:

$$\mathbb{E}[V(d)] \leq \Big(1 - \frac{p_1 p_2}{4D}\Big)^K. \tag{12}$$

The consistency of BRF suffices to have $K \to \infty$ in probability, which will be shown in Lemma 5. □

Lemma 5. *For sufficiently large n, every node of the tree will be split infinite times in probability, on the condition that the CDF of the splitting points is right-continuous at 0 and left-continuous at 1.*

Proof. Revisit that the splitting point is obtained either by random sampling method with a probability p_2 or by optimizing the impurity criterion with a probability $1 - p_2$. Thus, the final selected splitting point through the two methods can be viewed as a random variable $W_i (i \in \{1, 2, \ldots, K\})$ with CDF F_{W_i} from the tree root to a leaf.

For any given K and a constant $\delta > 0$, the smallest child node of the root has the size $M_1 = \min(W_1, 1 - W_1)$ at least $\delta^{1/K}$ with the probability:

$$\begin{aligned}\mathbb{P}\left(M_1 \geq \delta^{1/K}\right) &= \mathbb{P}\left(\delta^{1/K} \leq W_1 \leq 1 - \delta^{1/K}\right) \\ &= F_{W_1}(1 - \delta^{1/K}) - F_{W_1}(\delta^{1/K}). \end{aligned} \tag{13}$$

Without loss of generality, we scale the values of attributes to the range $[0, 1]$ for each node, then after K splits, the smallest child at the K-th level have the size at least δ with the probability at least

$$\prod_{i=1}^{K}(F_{W_i}(1 - \delta^{1/K}) - F_{W_i}(\delta^{1/K})), \tag{14}$$

which is derived by assuming that the same attribute is split at each level of the tree. If different attributes are split at different levels, the bound (14) also holds. Since F_{W_i} is right-continuous at 0 and left-continuous at 1, $F_{W_i}(1 - \delta^{1/K}) - F_{W_i}(\delta^{1/K}) \to 1$ as $\delta \to 0$. Thus, $\forall \epsilon_1 > 0$, $\exists \delta_1 > 0$, such that

$$\prod_{i=1}^{K}(F_{W_i}(1 - \delta_1^{1/K}) - F_{W_i}(\delta_1^{1/K})) > (1 - \epsilon_1)^K. \tag{15}$$

Besides, $\forall \epsilon > 0$, $\exists \epsilon_1 > 0$, such that

$$(1 - \epsilon_1)^K > 1 - \epsilon. \tag{16}$$

The above (15) and (16) show that each node at K-th level of the tree has the size of δ with probability at least $1 - \epsilon$.

Because the distribution of $\boldsymbol{X}$ has a non-zero density, each of these nodes has a positive measure with respect to $\mu_{\boldsymbol{X}}$. Defining

$$p = \min_{l\text{: a leaf at } K\text{-th level}} \mu_{\boldsymbol{X}}(l), \tag{17}$$

we know $p > 0$ since the minimum is over finitely many leaves and each leaf contains a set of positive measure.

In a data set of size n, the number of data points falling in the leaf A is $\mathrm{Binomial}(n, p)$. Then, the number of estimation points in A is $np/2$ assuming the $Ratio$ is 0.5 without loss of generality. According to Chebyshev's inequality, we can bound $N(A^E)$ as follows:

$$\begin{aligned}\mathbb{P}\left(N(A^E) < k_n\right) &= \mathbb{P}\left(N(A^E) - \frac{np}{2} < k_n - \frac{np}{2}\right) \\ &\overset{(a)}{=} \frac{1}{2}\mathbb{P}\left(|N(A^E) - \frac{np}{2}| > |k_n - \frac{np}{2}|\right) \\ &\leq \frac{1}{|k_n - \frac{np}{2}|^2}, \end{aligned} \tag{18}$$

where the equation (a) is due to the fact that $k_n - \frac{np}{2}$ is negative as $n \to \infty$. The RHS of (18) goes to zero as $n \to \infty$, so the leaf contains at least k_n estimation points in a high probability. According to the stopping condition, if the number of estimation points in the node is larger than k_n, the tree will continue to grow. So, $K \to \infty$ in probability. □

4 Discussion

In this section we discuss the proposed BRF with three variants of random forests which have the proved consistency, *i.e.,* [Biau *et al.*, 2008], [Biau, 2012] and [Denil *et al.*, 2014] denoted by $Biau08$, $Biau12$ and $Denil14$ respectively. Besides, we also discuss these models with Breiman's original random forests [Breiman, 2001] denoted by $Breiman$.

In terms of the data point partitioning, as long as the trees in the forests are constructed using the labels of data points, it is a must to partition the data points to ensure consistency, because Lemma 4 requires the labels of the voting data do not influence the structure of the tree. Thus, BRF, $Biau12$ and $Denil14$ all require data partitioning, whereas $Biau08$ and $Breiman$ do not need it.

For candidate attribute selection, according to Lemma 4, to ensure consistency, every attribute of the data must be split in probability as $n \to \infty$. For $Biau08$, it chooses a single attribute uniformly at random. $Biau12$ chooses a fixed number of random candidate attributes with replacement. $Denil14$ chooses $\min(1 + Poisson(\lambda), D)$ candidate attributes without replacement. BRF chooses 1 or $\sqrt{D}$ attributes with a probability satisfying a Bernoulli distribution B_1 without replacement. At last, $Breiman$ chooses a fixed number of random candidate attributes without replacement.

Thirdly, as for the selection of splitting points, according to Lemma 4, each candidate splitting point should be selected to split in probability so as to guarantee the consistency. $Biau08$ selects a point uniformly at random to split. $Biau12$ selects the midpoint in each attribute as the splitting point. $Denil14$ selects a few structure points at random and searches for the optimal splitting point over the range defined by previously selected points. BRF implements two strategies for splitting point selection in another Bernoulli distribution B_2. The two strategies either select a point uniformly at random to split or search for the splitting point that gives the largest impurity decrease. Lastly, $Breiman$ checks all the candidate splitting points and chooses the one with the largest impurity decrease.

According to the above discussion, we can find that the proposed BRF closely resembles to Breiman's original random forests. The key difference is that two independent Bernoulli distributions are introduced into the selection of splitting attributes and points to ensure consistency. Another noticeable difference is the data point partitioning procedure. In fact, all strategies implemented in BRF are to ensure the theoretical consistency as well as maintain the empirical performance.

5 Experiments

5.1 Data sets

Table 1 reports the 9 benchmark data sets [Lichman, 2013] ranked by the number of attributes. The benchmark data sets have different number of instances (from small to large), and also include low, moderate, and high dimensional attributes for binary and multi-class classification. Therefore, they are sufficiently representative to demonstrate the ability of random forests to handle different types of data.

Table 1: Detailed information of the benchmark data sets

DATA SET	INSTANCES	ATTRIBUTES	CLASSES
WINE	178	13	3
VEHICLE	946	18	4
IMAGE	2310	19	7
CHESS	3196	36	2
LAND-COVER	675	148	9
MADELON	2600	500	2
INDOORLOC	21048	529	4
ADS	3279	1558	2
GISETTE	13500	5000	2

5.2 Experimental settings

Since the algorithms are each parameterized slightly differently, it is not possible to use the same parameters for all methods. $Breiman$, $Denil14$ and BRF specify a minimum leaf size, which is set to 5 as suggested in [Breiman, 2001]. $Biau08$ and $Biau12$ are parameterized in terms of a target number of leaves which we set to be $n/5$, meaning the trees are approximately the same size as those parameterized by the minimum leaf size. For the forest size M, we set $M = 100$. As for the ratio of structure points to entire points $(Ratio)$, we all set $Ratio = 0.5$ for $Biau12$, $Denil14$ and BRF.

Moreover, $Denil14$ needs to choose m structure points for determining the search range of the splitting point which we set $m = 100$ according to [Denil *et al.*, 2014]. For BRF, we set the probability $p_1 = p_2 = 0.05$ in the two Bernoulli distributions respectively. Besides, for each data set, a 10 times 10-fold cross-validation is performed to reduce the influence of randomness.

5.3 Comparisons of different random forests

Table 2 reports the accuracies of different algorithms. The highest accuracy of the consistent random forests algorithms on each data set is in boldfaced. Besides, $\Delta theory$ shows the improvement of the proposed BRF compared to the state-of-the-art consistent random forests; $\Delta practice$ exhibits the gap between BRF and Breiman's original random forests.

As expected, among all consistent random forests algorithms, BRF achieves the highest accuracy. Compared to the state-of-the-art consistent random forests, *i.e.,* $Denil14$, the accuracy improvement is remarkable, especially on INDOORLOC data set where the improvement is up to 65.58%. The reason behind the huge improvement is that the INDOORLOC data set has some attributes with numerous values. Thus, in $Denil14$, the preselected m structure points may only possess a few values of the attribute, which have a great effect on selecting the splitting point and further influence the tree structure as well as performance. The classification accuracy improvement over $Denil14$ is also supported by Wilcoxon signed ranked test [Demšar, 2006] which assures the statistical significance on almost all data sets, as shown in Table 2. Combining the comparisons with other consistent algorithms, we conclude that the improvement of BRF is mainly attributed to the two Bernoulli distributions in the strategies of selecting splitting attributes and points.

When comparing BRF to $Breiman$, the gap still remains

Table 2: Classification accuracy ($ACC\%$) of different random forests algorithms on different data sets

DATA SET	$Biau08$ [1]	$Biau12$ [1]	$Denil14$ [1]	BRF	$Breiman$ [2]	$\Delta theory$	$\Delta practice$
WINE	40.59	41.18	96.47	**97.65**	98.27	1.18	0.62
VEHICLE	27.98	23.10	68.81	**71.67** $^\bullet$	74.70	2.86	3.03
IMAGE	12.42	13.29	95.45	**96.06**	97.71	0.61	1.65
CHESS	55.64	54.95	61.32	**97.12** $^\bullet$	98.72	35.80	1.60
LAND-COVER	16.12	15.37	78.06	**82.99** $^\bullet$	86.08	4.93	3.09
MADELON	49.27	50.31	54.81	**69.23** $^\bullet$	76.58	14.42	7.35
INDOORLOC	26.61	25.12	34.39	**99.97** $^\bullet$	100.00	65.58	0.03
ADS	86.12	86.06	85.99	**94.43** $^\bullet$	97.59	8.44	3.16
GISETTE	50.08	50.27	84.97	**94.83** $^\bullet$	97.43	9.86	2.6

[1] $Biau08$, $Biau12$, and $Denil14$ are consistent random forests algorithms.
[2] $Breiman$ is the original random forests algorithm. (It has not been proved to be consistent.)
$^\bullet$ BRF is significantly better than $Denil14$ at a level of significance 0.05.

but has been significantly narrowed down to within 3% in almost all data sets. Actually, the gap is caused by the data point partitioning procedure in BRF which reduces the instance size for constructing the trees. Particularly, on MADELON data set, the gap is still 7.35% although BRF has already tremendously outperformed $Denil14$. The main reason is that after the partitioning, the number of instances are severely insufficient for 500 attributes.

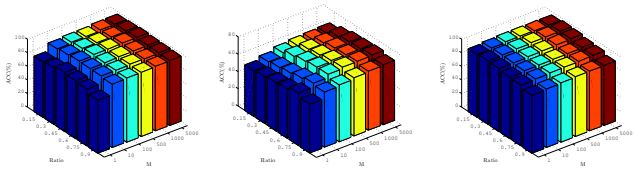
Overall, BRF is proved to be consistent. In addition, BRF not only empirically outperforms all other consistent random forests, but also achieves the closest empirical performance to $Breiman$ than previous theoretical variants including the state-of-the-art consistent random forests $Denil14$.

5.4 Cross-test for parameter settings

In this subsection, we conduct a series of cross-test experiments to evaluate the influence of parameters in BRF, *i.e.,* the ratio of structure points to entire points ($Ratio$), the number of trees M and the probabilities p_1 as well as p_2 in the Bernoulli distributions.

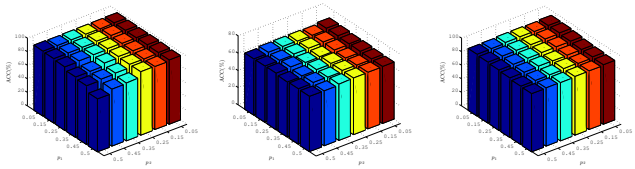
Due to page limitations, we select three representative data sets with small, middle and large instances or attributes, *i.e.*, CHESS, MADELON and ADS. We test $Ratio$ among $\{0.15, 0.3, 0.45, 0.6, 0.75, 0.9\}$, M among $\{1, 10, 100, 500, 1000, 5000\}$, p_1 and p_2 among $\{0.05, 0.15, 0.25, 0.35, 0.45, 0.5\}$.

The results in Fig. 2 show that the accuracy increases gradually and approaches to a stable value as the number of trees M increases. Besides, Fig. 2 indicates that the best $Ratio$ varies across data sets (*e.g.,* 0.45 for CHESS, 0.75 for MADELON, 0.3 for ADS). In addition, Fig. 3 demonstrates that BRF is not sensitive to p_1 and p_2 as long as they have small values (*i.e.,* $p_1, p_2 \leq 0.25$). Recalling the tree construction procedure of BRF, we know that p_1 and p_2 are two parameters to balance the consistency analysis and empirical performance. When $p_1, p_2 \to 0$, BRF is close to $Breiman$. When $p_1, p_2 \to 1$, BRF is close to $Biau08$. Therefore, p_1 and p_2 can use small values empirically. Meanwhile, on one hand, a large $Ratio$ value implies less estimation points, which results in imprecise leaf predictors. On the other hand, a small $Ratio$ value will lead to less structure points, so the structure of the tree will not be optimal. Generally, $Ratio$ value is set as 0.5 to balance the structure and estimation parts, without



(a) CHESS (b) MADELON (c) ADS

Figure 2: Classification accuracy ($ACC\%$) of BRF with different M and $Ratio$. ($p_1 = p_2 = 0.05$)



(a) CHESS (b) MADELON (c) ADS

Figure 3: Classification accuracy ($ACC\%$) of BRF with different p_1 and p_2. ($M = 100, Ratio = 0.5$)

favour of any part. Besides, the ensemble size (*i.e.*, the number of trees M) is set considering the computation time and accuracy, because the accuracy tends to be stable once the ensemble size is larger than a threshold.

6 Conclusion

In this paper, we proposed a new Bernoulli random forests model (BRF) with sound empirical performance and proved theoretical consistency. We argued that although Breiman's original random forests have very good empirical performance, their theoretical consistency has yet to be proved due to the highly sensitive data-driven tree construction procedure. On the other hand, several random forests variants have nice theoretical consistency but they all suffer from poor empirical performance. In the proposed BRF, we employ two Bernoulli distributions to help determine the attributes as well as the splitting points used by each node. For each Bernoulli trial, it determines whether to use a random process or a deterministic process to build the tree with a probability value. As a result, the trees constructed by BRF are much less data-

dependent, compared to Breiman's original random forests, yet still have much better performance compared to theoretically consistent random forests. Experiments and comparisons validate that BRF significantly outperforms all existing consistent random forests, and its performance is very close to Breiman's original random forests. BRF takes a step forward to close the gap between theoretical consistency and empirical soundness of random forests classification.

Acknowledgments

This research is supported in part by the Major State Basic Research Development Program of China (973 Program, 2012CB315803), the National Natural Science Foundation of China (61371078), and the Research Fund for the Doctoral Program of Higher Education of China (20130002110051).

References

[Biau *et al.*, 2008] Gérard Biau, Luc Devroye, and Gábor Lugosi. Consistency of random forests and other averaging classifiers. *Journal of Machine Learning Research*, 9:2015–2033, 2008.

[Biau, 2012] Gérard Biau. Analysis of a random forests model. *Journal of Machine Learning Research*, 13(1):1063–1095, 2012.

[Breiman *et al.*, 1984] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees*. CRC press, 1984.

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

[Breiman, 2004] Leo Breiman. Consistency for a simple model of random forests. Technical Report 670, Statistical Department, University of California at Berkeley, 2004.

[Cutler *et al.*, 2007] D Richard Cutler, Thomas C Edwards Jr, Karen H Beard, Adele Cutler, Kyle T Hess, Jacob Gibson, and Joshua J Lawler. Random forests for classification in ecology. *Ecology*, 88(11):2783–2792, 2007.

[Demšar, 2006] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[Denil *et al.*, 2013] Misha Denil, David Matheson, and De Freitas Nando. Consistency of online random forests. In *Proceedings of The 30th International Conference on Machine Learning (ICML 2013)*, pages 1256–1264, 2013.

[Denil *et al.*, 2014] Misha Denil, David Matheson, and Nando De Freitas. Narrowing the gap: Random forests in theory and in practice. In *Proceedings of The 31st International Conference on Machine Learning (ICML 2014)*, pages 665–673, 2014.

[Devroye *et al.*, 2013] Luc Devroye, László Györfi, and Gábor Lugosi. *A probabilistic theory of pattern recognition*, volume 31. Springer Science & Business Media, 2013.

[Genuer, 2010] Robin Genuer. Risk bounds for purely uniformly random forests. *arXiv preprint arXiv:1006.2980*, 2010.

[Genuer, 2012] Robin Genuer. Variance reduction in purely random forests. *Journal of Nonparametric Statistics*, 24(3):543–562, 2012.

[Lichman, 2013] M. Lichman. UCI machine learning repository, 2013.

[Prasad *et al.*, 2006] Anantha M Prasad, Louis R Iverson, and Andy Liaw. Newer classification and regression tree techniques: bagging and random forests for ecological prediction. *Ecosystems*, 9(2):181–199, 2006.

[Shotton *et al.*, 2011] Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-time human pose recognition in parts from single depth images. In *Proceedings of 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2011)*, pages 1297–1304, 2011.

[Svetnik *et al.*, 2003] Vladimir Svetnik, Andy Liaw, Christopher Tong, J Christopher Culberson, Robert P Sheridan, and Bradley P Feuston. Random forest: a classification and regression tool for compound classification and qsar modeling. *Journal of chemical information and computer sciences*, 43(6):1947–1958, 2003.

[Xiong *et al.*, 2012] Caiming Xiong, David Johnson, Ran Xu, and Jason J Corso. Random forests for metric learning with implicit pairwise position dependence. In *Proceedings of The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2012)*, pages 958–966, 2012.