Impact of subsampling and pruning on random forests.

Roxane Duroux
Sorbonne Universités, UPMC Univ Paris 06, F-75005, Paris, France
roxane.duroux@upmc.fr

Erwan Scornet
Sorbonne Universités, UPMC Univ Paris 06, F-75005, Paris, France
erwan.scornet@upmc.fr

Abstract

Random forests are ensemble learning methods introduced by Breiman (2001) that operate by averaging several decision trees built on a randomly selected subspace of the data set. Despite their widespread use in practice, the respective roles of the different mechanisms at work in Breiman's forests are not yet fully understood, neither is the tuning of the corresponding parameters. In this paper, we study the influence of two parameters, namely the subsampling rate and the tree depth, on Breiman's forests performance. More precisely, we show that fully developed subsampled forests and pruned (without subsampling) forests have similar performances, as long as respective parameters are well chosen. Moreover, experiments show that a proper tuning of subsampling or pruning lead in most cases to an improvement of Breiman's original forests errors.

Index Terms — Random forests, randomization, parameter tuning, subsampling, tree depth.

1 Introduction

Random forests are a class of learning algorithms used to solve pattern recognition problems. As ensemble methods, they grow many base learners (*i.e.*, decision trees) and aggregate them to predict. Building several different trees from a single data set requires to randomize the tree building process by, for example, sampling the data set. Thus, there exists a large variety of random forests, depending on how trees are designed and how the randomization is introduced in the whole procedure.

One of the most popular random forests is that of Breiman (2001) which grows trees based on CART procedure (Classification And Regression Trees, Breiman et al., 1984) and randomizes both the training set and the splitting directions. Breiman's (2001) random forests have been under active investigation during the last decade mainly because of their good practical performance and their ability to handle high-dimensional data sets. They are acknowledged to be state-of-the-art methods in fields such as genomics (Qi, 2012) and pattern recognition (Rogez et al., 2008), just to name a few.

The ease of the implementation of random forests algorithms is one of their key strengths and has greatly contributed to their widespread use. A proper tuning of the different parameters of the algorithm is not mandatory to obtain a plausible prediction, making random forests a turn-key solution to deal with large, heterogeneous data sets.

Several authors studied the influence of the parameters on random forests accuracy. For example, the number M of trees in the forests has been thoroughly investigated by Díaz-Uriarte and de Andrés (2006) and Genuer et al. (2010). It is easy to see that the computational cost for inducing a forest increases linearly with M, so a good choice for M results from a trade-off between computational complexity and accuracy (M must be large enough for predictions to be stable). Díaz-Uriarte and de Andrés (2006) argued that in micro-array classification problems, the particular value of M is irrelevant, assuming that M is large enough (typically over 500). Several recent studies provided theoretical guarantees for choosing M. Scornet (2014) proposed a way to tune M so that the error of the forest is minimal. Mentch and Hooker (2014) and Wager (2014) gave a more in-depth analysis by establishing a central limit theorem for random forests prediction and providing a method to estimate their variance. All in all, the role of M on the forest prediction is broadly understood.

Besides the number of trees, forests depend on three parameters: the number a_n of data points selected to build each tree, the number m_{try} of preselected variables along which the best split is chosen, and the minimum number `nodesize` of data points in each cell of each tree. The effect of m_{try} was thoroughly investigated in Díaz-Uriarte and de Andrés (2006) and Genuer et al. (2010) who claimed that the default value is either optimal or too small, therefore leading to no global understanding of this parameter. The story is roughly the same regarding the parameter `nodesize` for which the default value has been reported as a good choice by Díaz-Uriarte and de Andrés (2006). Furthermore, there is no theoretical guarantee to support the default values of parameters or any of the data-driven tuning proposed in the literature.

Our objective in this paper is two-folds: (i) to provide a theoretical framework to analyse the influence of the number of points a_n used to build each tree, and the tree depth (corresponding to parameters `nodesize` or `maxnode`) on the random forest performances; (ii) to implement several experiments to test our theoretical findings. The paper is organized as follows. Section 2 is devoted to notations and presents Breiman's random forests algorithm. To carry out a theoretical analysis of the subsample size and the tree depth, we study in Section 3 a particular random forest called median forest. We establish an upper bound for the risk of median forests and by doing so, we highlight the fact that subsampling and pruning have similar influence on median forest predictions. The numerous experiments on Breiman forests are presented in Section 4. Proofs are postponed to Section 6.

2 First definitions

2.1 General framework

In this paper, we consider a training sample $\mathcal{D}_n = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$ of $[0,1]^d \times$ $\mathbb{R}$-valued independent and identically distributed observations of a random pair $(\mathbf{X}, Y)$, where $\mathbb{E}[Y^2] < \infty$. The variable $\mathbf{X}$ denotes the predictor variable and Y the response variable. We wish to estimate the regression function $m(\mathbf{x}) = \mathbb{E}\left[Y|\mathbf{X} = \mathbf{x}\right]$. In this context, we use random forests to build an estimate $m_n : [0, 1]^d \to \mathbb{R}$ of m, based on the data set $\mathcal{D}_n$.

Random forests are classification and regression methods based on a collection of M randomized trees. We denote by $m_n(\mathbf{x}, \Theta_j, \mathcal{D}_n)$ the predicted value at point $\mathbf{x}$ given by the j-th tree, where $\Theta_1, \ldots, \Theta_M$ are independent random variables, distributed as a generic random variable Θ, independent of the sample $\mathcal{D}_n$. In practice, the variable Θ can be used to sample the data set or to select the candidate directions or positions for splitting. The predictions of the M randomized trees are then averaged to obtain the random forest prediction

$$m_{M,n}(\mathbf{x}, \Theta_1, \ldots, \Theta_M, \mathcal{D}_n) = \frac{1}{M} \sum_{m=1}^{M} m_n(\mathbf{x}, \Theta_m, \mathcal{D}_n). \tag{1}$$

By the law of large numbers, for any fixed $\mathbf{x}$, conditionally on $\mathcal{D}_n$, the finite forest estimate tends to the infinite forest estimate

$$m_{\infty,n}(\mathbf{x}, \mathcal{D}_n) = \mathbb{E}_{\Theta}\left[m_n(\mathbf{x}, \Theta)\right].$$

For the sake of simplicity, we denote $m_{\infty,n}(\mathbf{x}, \mathcal{D}_n)$ by $m_{\infty,n}(\mathbf{x})$. Since we carry out our analysis within the $\mathbb{L}^2$ regression estimation framework, we say that $m_{\infty,n}$ is $\mathbb{L}^2$ consistent if its risk, $\mathbb{E}[m_{\infty,n}(\mathbf{X}) - m(\mathbf{X})]^2$, tends to zero, as n goes to infinity.

2.2 Breiman's forests

Breiman's (2001) forest is one of the most used random forest algorithms. In Breiman's forests, each node of a single tree is associated with a hyper-rectangular cell included in $[0, 1]^d$. The root of the tree is $[0, 1]^d$ itself and, at each step of the tree construction, a node (or equivalently its corresponding cell) is split in two parts. The terminal nodes (or leaves), taken together, form a partition of $[0, 1]^d$. In details, the algorithm works as follows:

1. Grow M trees as follows:

   (a) Prior to the j-th tree construction, select uniformly with replacement, a_n data points among $\mathcal{D}_n$. Only these a_n observations are used in the tree construction.

   (b) Consider the cell $[0, 1]^d$.

   (c) Select uniformly without replacement m_{try} coordinates among $\{1, \ldots, d\}$.

   (d) Select the split minimizing the CART-split criterion (see Breiman et al., 1984, for details) along the pre-selected m_{try} directions.

   (e) Cut the cell at the selected split.

   (f) Repeat $(c) - (e)$ for the two resulting cells until each cell of the tree contains less than `nodesize` observations.

(g) For a query point $\mathbf{x}$, the j-th tree outputs the average of the Y_i falling into the same cell as $\mathbf{x}$.

2. For a query point $\mathbf{x}$, Breiman's forest outputs the average of the predictions given by the M trees.

The whole procedure depends on four parameters: the number M of trees, the number a_n of sampled data points in each tree, the number m_{try} of pre-selected directions for splitting, and the maximum number `nodesize` of observations in each leaf. By default in the `R` package `randomForest`, M is set to 500, $a_n = n$ (bootstrap samples are used to build each tree), $m_{\text{try}} = d/3$ and `nodesize`$= 5$.

Note that selecting the split that minimizes the CART-split criterion is equivalent to select the split such that the two resulting cells have a minimal (empirical) variance (regarding the Y_i falling into each of the two cells).

3 Theoretical results

The numerous mechanisms at work in Breiman's forests, such as the subsampling step, the CART-criterion and the trees aggregation, make the whole procedure difficult to theoretically analyse. Most attempts to understand the random forest algorithms (see e.g., Biau et al., 2008; Ishwaran and Kogalur, 2010; Denil et al., 2013) have focused on simplified procedures, ignoring the subsampling step and/or replacing the CART-split criterion by a data independent procedure more amenable to analysis. On the other hand, recent studies try to dissect the original Breiman's algorithm in order to prove its asymptotic normality (Mentch and Hooker, 2014; Wager, 2014) or its consistency (Scornet et al., 2015). When studying the original algorithm, one faces the complexity of the algorithm thus requiring high-level mathematics to prove insightful—but rough— results.

In order to provide theoretical guarantees on the parameters default values in random forests, we focus in this section on a simplified random forest called median forest (see, for example, Biau and Devroye, 2014, for details on median tree).

3.1 Median Forests

To take one step further into the understanding of Breiman's (2001) forest behavior, we study the median random forest, which satisfies the X-property (Devroye et al., 1996). Indeed, its construction depends only on the X_i's which is a good trade off between the complexity of Breiman's (2001) forests and the simplicity of totally non adaptive forests, whose construction is independent of the data set. Besides, median forests can be tuned such that each leaf of each tree contains exactly one point. In this way, they are closer to Breiman's forests than totally non adaptive forests (whose cells cannot contain a pre-specified number of points) and thus provide a good understanding on Breiman's forests performance even when there is exactly one data point in each leaf.

We now describe the construction of median forest. In the spirit of Breiman's (2001) algorithm, before growing each tree, data are subsampled, that is a_n

points ($a_n < n$) are selected, without replacement. Then, each split is performed on an empirical median along a coordinate, chosen uniformly at random among the d coordinates. Recall that the median of n real valued random variables $X_1, \ldots, X_n$ is defined as the only $X_{(\ell)}$ satisfying $F_n(X_{(\ell-1)}) \leq 1/2 < F_n(X_{(\ell)})$, where the $X_{(i)}$'s are ordered increasingly and F_n is the empirical distribution function of X. Note that data points on which splits are performed are not sent down to the resulting cells. This is done to ensure that data points are uniformly distributed on the resulting cells (otherwise, there would be at least one data point on the edge of a resulting cell, and thus the data points distribution would not be uniform on this cell). Finally, the algorithm stops when each cell has been cut exactly k_n times, *i.e.*, `nodesize`$= a_n 2^{-k_n}$. The parameter k_n, also known as the level of the tree, is assumed to verify $a_n 2^{-k_n} \geq 4$. The overall construction process is recalled below.

1. Grow M trees as follows:
   (a) Prior to the j-th tree construction, select uniformly without replacement, a_n data points among $\mathcal{D}_n$. Only these a_n observations are used in the tree construction.
   (b) Consider the cell $[0, 1]^d$.
   (c) Select uniformly one coordinate among $\{1, \ldots, d\}$ without replacement.
   (d) Cut the cell at the empirical median of the X_i falling into the cell along the preselected direction.
   (e) Repeat $(c) - (d)$ for the two resulting cells until each cell has been cut exactly k_n times.
   (f) For a query point $\mathbf{x}$, the j-th tree outputs the average of the Y_i falling into the same cell as $\mathbf{x}$.
2. For a query point $\mathbf{x}$, median forest outputs the average of the predictions given by the M trees.

3.2 Main theorem

Let us first make some regularity assumptions on the regression model.

(H) *One has*

$$Y = m(\mathbf{X}) + \varepsilon,$$

where ε is a centred noise such that $\mathbb{V}[\varepsilon | \mathbf{X} = \mathbf{x}] \leq \sigma^2$, where $\sigma^2 < \infty$ is a constant. Moreover, $\mathbf{X}$ is uniformly distributed on $[0, 1]^d$ and m is L-Lipschitz continuous.

Theorem 3.1 presents an upper bound for the $\mathbb{L}^2$ risk of $m_{\infty,n}$.

Theorem 3.1. *Assume that* **(H)** *is satisfied. Then, for all n, for all $\mathbf{x} \in [0, 1]^d$,*

$$\mathbb{E}\big[m_{\infty,n}(\mathbf{x}) - m(\mathbf{x})\big]^2 \leq 2\sigma^2 \frac{2^k}{n} + dL^2 C_1 \left(1 - \frac{3}{4d}\right)^k. \tag{2}$$

In addition, let $\beta = 1 - 3/4d$. The right-hand side is minimal for

$$k_n = \frac{1}{\ln 2 - \ln \beta}\Big[\ln(n) + C_3\Big], \tag{3}$$

under the condition that $a_n \geq C_4 n^{\frac{\ln 2}{\ln 2 - \ln \beta}}$. For these choices of k_n and a_n, we have

$$\mathbb{E}\big[m_{\infty,n}(\mathbf{x}) - m(\mathbf{x})\big]^2 \leq C n^{\frac{\ln \beta}{\ln 2 - \ln \beta}}. \tag{4}$$

Equation (2) stems from a standard decomposition of the estimation/approximation error of median forests. Indeed, the first term in equation (2) corresponds to the estimation error of the forest as in Biau (2012) or Arlot and Genuer (2014) whereas the second term is the approximation error of the forest, which decreases exponentially in k. Note that this decomposition is consistent with the existing literature on random forests. Two common assumptions to prove consistency of simplified random forests are $n/2^k \to \infty$ and $k \to \infty$, which respectively controls the estimation and approximation of the forest. According to Theorem 3.1, making these assumptions for median forests results in proving their consistency.

Note that the estimation error of a single tree grown with a_n observations is of order $2^k/a_n$. Thus, because of the subsampling step (*i.e.*, since $a_n < n$), the estimation error of median forests $2^k/n$ is smaller than that of a single tree. The variance reduction of random forests is a well-known property, already noticed by Genuer (2012) for a totally non adaptive forest, and by Scornet (2014) in the case of median forests. In our case, we exhibit an explicit bound on the forest variance, which allows us to precisely compare it to the individual tree variance therefore highlighting a first benefit of forests over singular trees.

As for the first term in inequality (2), the second term could be expected. Indeed, in the levels close to the root, a split is very close to the center of a side of a cell (since $\mathbf{X}$ is uniformly distributed over $[0,1]^d$). Thus, for all k small enough, the approximation error of median forests should be close to that of centred forests studied by Biau (2012). Surprisingly, the rate of consistency of median forests is faster than that of centred forest established in Biau (2012), which is equal to

$$\mathbb{E}\big[m_{\infty,n}^{cc}(\mathbf{X}) - m(\mathbf{X})\big]^2 \leq C n^{\frac{-3}{4d\ln 2 + 3}}, \tag{5}$$

where $m_{\infty,n}^{cc}$ stands for the centred forest estimate. A close inspection of the proof of Proposition 2.2 in Biau (2012) shows that it can be easily adapted to match the (more optimal) upper bound in Theorem 3.1.

Noteworthy, the fact that the upper bound (4) is sharper than (5) appears to be important in the case where $d = 1$. In that case, according to Theorem 3.1, for all n, for all $\mathbf{x} \in [0,1]^d$,

$$\mathbb{E}\big[m_{\infty,n}(\mathbf{x}) - m(\mathbf{x})\big]^2 \leq C n^{-2/3},$$

which is the minimax rate over the class of Lipschitz functions (see, e.g., Stone, 1980, 1982). This was to be expected since, in dimension one, median random forests are simply a median tree which is known to reach minimax rate (Devroye et al., 1996). Unfortunately, for $d = 1$, the centred forest bound (5) turns out to be suboptimal since it results in

$$\mathbb{E}\big[m_{\infty,n}^{cc}(\mathbf{X}) - m(\mathbf{X})\big]^2 \le Cn^{\frac{-3}{4\ln 2+3}}. \tag{6}$$

Theorem 3.1 allows us to derive rates of consistency for two particular forests: the pruned median forest, where no subsampling is performed prior to build each tree, and the fully developed median forest, where each leaf contains a small number of points. Corollary 1 deals with the pruned forests.

Corollary 1 (Pruned median forests)**.** *Let $\beta = 1 - 3/4d$. Assume that* **(H)** *is satisfied. Consider a median forest without subsampling (i.e., $a_n = n$) and such that the parameter k_n satisfies (3). Then, for all n, for all $\mathbf{x} \in [0,1]^d$,*

$$\mathbb{E}\big[m_{\infty,n}(\mathbf{x}) - m(\mathbf{x})\big]^2 \le Cn^{\frac{\ln\beta}{\ln 2-\ln\beta}}.$$

Up to an approximation, Corollary 1 is the counterpart of Theorem 2.2 in Biau (2012) but tailored for median forests. Indeed, up to a small modification of the proof of Theorem 2.2, the rate of consistency provided in Theorem 2.2 for centred forests and that of Corollary 1 for median forests are identical. Note that, for both forests, the optimal depth k_n of each tree is the same.

Corollary 2 handles the case of fully grown median forests, that is forests which contain a small number of points in each leaf. Indeed, note that since $k_n = \log_2(a_n) - 2$, the number of observations in each leaf varies between 4 and 8.

Corollary 2 (Fully grown median forest)**.** *Let $\beta = 1 - 3/4d$. Assume that* **(H)** *is satisfied. Consider a fully grown median forest whose parameters k_n and a_n satisfy $k_n = \log_2(a_n) - 2$. The optimal choice for a_n (that minimizes the $\mathbb{L}^2$ error in (2)) is then given by (3), that is*

$$a_n = C_4 n^{\frac{\ln 2}{\ln 2-\ln\beta}}.$$

In that case, for all n, for all $\mathbf{x} \in [0,1]^d$,

$$\mathbb{E}\big[m_{\infty,n}(\mathbf{x}) - m(\mathbf{x})\big]^2 \le Cn^{\frac{\ln\beta}{\ln 2-\ln\beta}}.$$

Whereas each individual tree in the fully developed median forest is inconsistent (since each leaf contains a small number of points), the whole forest is consistent and its rate of consistency is provided by Corollary 2. Besides, Corollary 2 provides us with the optimal subsampling size for fully developed median forests.

Provided a proper parameter tuning, pruned median forests without subsampling and fully grown median forests (with subsampling) have similar performance. A close look at Theorem 3.1 shows that the subsampling size has no effect on the performance, provided it is large enough. The parameter of real importance is the tree depth k_n. Thus, fixing k_n as in equation (3), and by varying the subsampling rate a_n/n one can obtain random forests that are more-or-less pruned, all satisfying the optimal bound in Theorem 3.1. In that way, Corollary 1 and 2 are simply two particular examples of such forests.

Although our analysis sheds some light on the role of subsampling and tree depth, the statistical performances of median forests does not allow us to choose between pruned and subsampled forests. Interestingly, note that these two types of random forests can be used in two different contexts. If one wants to obtain fast predictions, then subsampled forests, as described in Corollary 2, are to be preferred since their computational time is lower than pruned random forests (described in Corollary 1). However, if one wants to build more accurate predictions, pruned random forests have to be chosen since the recursive random forest procedure allows to build several forests of different tree depths in one run, therefore allowing to select the best model among these forests.

4 Experiments

In the light of Section 3, we carry out some simulations to investigate (i) how pruned and subsampled forests compare with Breiman's forests and (ii) the influence of subsampling size and tree depth on Breiman's procedure. To do so, we start by defining various regression models on which the several experiments are based. Throughout this section, we assess the forest performances by computing their empirical $\mathbb{L}^2$ error.

Model 1: $n = 800, d = 50, Y = \tilde{X}_1^2 + \exp(-\tilde{X}_2^2)$

Model 2: $n = 600, d = 100, Y = \tilde{X}_1\tilde{X}_2 + \tilde{X}_3^2 - \tilde{X}_4\tilde{X}_7 + \tilde{X}_8\tilde{X}_{10} - \tilde{X}_6^2 + \mathcal{N}(0, 0.5)$

Model 3: $n = 600, d = 100, Y = -\sin(2\tilde{X}_1) + \tilde{X}_2^2 + \tilde{X}_3 - \exp(-\tilde{X}_4) + \mathcal{N}(0, 0.5)$

Model 4: $n = 600, d = 100, Y = \tilde{X}_1 + (2\tilde{X}_2 - 1)^2 + \sin(2\pi\tilde{X}_3)/(2 - \sin(2\pi\tilde{X}_3)) + \sin(2\pi\tilde{X}_4) + 2\cos(2\pi\tilde{X}_4) + 3\sin^2(2\pi\tilde{X}_4) + 4\cos^2(2\pi\tilde{X}_4) + \mathcal{N}(0, 0.5)$

Model 5: $n = 700, d = 20, Y = \mathbb{1}_{\tilde{X}_1 > 0} + \tilde{X}_2^3 + \mathbb{1}_{\tilde{X}_4 + \tilde{X}_6 - \tilde{X}_8 - \tilde{X}_9 > 1 + \tilde{X}_{10}} + \exp(-\tilde{X}_2^2) + \mathcal{N}(0, 0.5)$

Model 6: $n = 500, d = 30, Y = \sum_{k=1}^{10} \mathbb{1}_{\tilde{X}_k^3 < 0} - \mathbb{1}_{\mathcal{N}(0,1) > 1.25}$

Model 7: $n = 600, d = 300, Y = \tilde{X}_1^2 + \tilde{X}_2^2\tilde{X}_3 \exp(-|\tilde{X}_4|) + \tilde{X}_6 - \tilde{X}_8 + \mathcal{N}(0, 0.5)$

Model 8: $n = 500, d = 1000, Y = \tilde{X}_1 + 3\tilde{X}_3^2 - 2\exp(-\tilde{X}_5) + \tilde{X}_6$

For all regression frameworks, we consider covariates $\mathbf{X} = (X_1, \ldots, X_d)$ that are uniformly distributed over $[0, 1]^d$. We also let $\tilde{X}_i = 2(X_i - 0.5)$ for $1 \leq i \leq d$. Some of these models are toy models (**Model 1, 5-8**). **Model 2** can be found in van der Laan et al. (2007) and **Models 3-4** are presented in Meier et al. (2009). All numerical implementations have been performed using the free R software. For each experiment, the data set is divided into a training set (80% of the data set) and a test set (the remaining 20%). Then, the empirical risk ($\mathbb{L}^2$ error) is evaluated on the test set.

4.1 Pruning

We start by studying Breiman's original forests and pruned Breiman's forests. Breiman's forests are the standard procedure implemented in the `R` package `randomForest`, with the parameters default values, as described in Section 2. Pruned Breiman's forests are similar to Breiman's forests except that the tree depth is controlled via the parameter `maxnodes` (which corresponds to the number of leaves in each tree) and that the whole sample $\mathcal{D}_n$ is used to build each tree.

In Figure 1, we present, for the **Models 1-8** introduced previously, the evolution of the empirical risk of pruned forests for different numbers of terminal nodes. We add the representation of the empirical risk of Breiman's original forest in order to compare all forests errors at a glance. Every sub-figure of Figure 1 presents forests built with 500 trees. The printed errors are obtained by averaging the risks of 50 forests. Because of the estimation/approximation compromise, we expect the empirical risk of pruned forests to be decreasing and then increasing, as the number of leaves grows. In most of the models, it seems that the estimation error is too low to be detected, this is why several risks in Figure 1 are only decreasing.

For every model, we can notice that pruned forests performance is comparable with the one of standard Breiman's forest, as long as the pruning parameter (the number of leaves) is well chosen. For example, for the **Model 1**, a pruned forest with approximately 110 leaves for each tree has the same empirical risk as the standard Breiman's forest. In the original algorithm of Breiman's forest, the construction of each tree uses a bootstrap sample of the data. For the pruned forests, the whole data set is used for each tree, and then the randomness comes only from the pre-selected directions for splitting. The performances of bootstrapped and pruned forests are very alike. Thus, bootstrap seems not to be the cornerstone of the Breiman's forest practical superiority to other regression algorithms. As it is shown in Corollary 1 and the simulations, pruning and sampling of the data set (here bootstrap) are equivalent.

In order to study the optimal pruning value (`maxnodes` parameter in the R algorithm), we draw the same curves as in Figure 1, for different learning data set sizes ($n = 100$, 200, 300 and 400). We also copy in an other graph the optimal values that we found for each size of the learning set. The optimal pruning value $m^\star$ is defined as

$$m^\star = \min\{m : |\hat{L}_m - \min_r \hat{L}_r| < 0.05 \times (\max_r \hat{L}_r - \min_r \hat{L}_r)\}$$

where $\hat{L}_r$ is the risk of the forest built with the parameter `maxnodes`$= r$. The results can be seen in Figure 2. According to the last sub-figure in Figure 2, the optimal pruning value seems to be proportional to the sample size. For **Model 1**, the optimal value $m^\star$ seems to verify $0.25n < m^\star < 0.3n$. The other models show a similar behaviour, as it can be seen in Figure 3.

We also present the $\mathbb{L}^2$ errors of pruned Breiman's forests for different pruning percentages (10%, 30%, 63%, 80% and 100%), when the sample size is fixed, for **Models 1-8**. The results can be found in Figure 4 in the form of box-plots. We can notice that the forests with a 30% pruning (*i.e.*, such that `maxnodes`$= 0.3n$)

give similar (**Model 5**) or best (**Model 6**) performances than the standard Breiman's forest.

4.2 Subsampling

In this section, we study the influence of subsampling on Breiman's forests by comparing the original Breiman's procedure with subsampled Breiman's forests. Subsampled Breiman's forests are nothing but Breiman's forests where the subsampling step consists in choosing a_n observations without replacement (instead of choosing n observations among n with replacement), where a_n is the subsample size. Comparison of Breiman's forests and subsampled Breiman's forests is presented in Figure 5 for the **Models 1-8** introduced previously. More precisely, we can see the evolution of the empirical risk of subsampled forests with different subsampling values, and the empirical risk of the Breiman's forest as a reference. Every sub-figure of Figure 5 presents forests built with 500 trees. The printed errors are obtained by averaging the risks of 50 forests.

For every model, we can notice that subsampled forests performance is comparable with the one of standard Breiman's forest, as long as the subsampling parameter is well chosen. For example, a forest with a subsampling rate of 50% has the same empirical risk as the standard Breiman's forest, for **Model 2**. Once again, the similarity between bootstrapped and subsampled Breiman's forests moves aside bootstrap as a performance criteria. As it is shown in Corollary 2 and the simulations, subsampling and bootstrap of the data set are equivalent.

We want of course to study the optimal subsampling size (`samplesize` parameter in the R algorithm). For this, we draw the curves of Figure 5 for different learning data set sizes, the same as in Figure 2. We also copy in an other graph the optimal subsample size $a_n^\star$ that we found for each size of the learning set. The optimal subsampling size $a_n^\star$ is defined as

$$a_n^\star = \min\{a : |\hat{L}_a - \min_s \hat{L}_s| < 0.05 \times (\max_s \hat{L}_s - \min_s \hat{L}_s)\}$$

where $\hat{L}_s$ is the risk of the forest with parameter `sampsize` $= s$. The results can be seen in Figure 6. The optimal subsampling size seems, once again, to be proportional to the sample size, as illustrated in the last sub-figure of Figure 6. For **Model 1**, the optimal value $a_n^\star$ seems to be close to $0.8n$. The other models show a similar behaviour, as it can be seen in Figure 7.

Then we present, in Figure 8, the $\mathbb{L}^2$ errors of subsampled Breiman's forests for different subsampling sizes ($0.4n$, $0.5n$, $0.63n$ and $0.9n$), when the sample size is fixed, for **Models 1-8**. We can notice that the forests with a subsampling size of $0.63n$ give similar performances than the standard Breiman's forests. This is not surprising. Indeed, a bootstrap sample contains around 63% of distinct observations. Moreover the high subsampling sizes, around $0.9n$, lead to small $\mathbb{L}^2$ errors. It may arise from the probably high signal/noise rate. In each model, when the noise is increasing, the results, exemplified in Figure 9, are less obvious. That is why we can lawfully use the subsampling size as an optimization parameter for the Breiman's forest performance.

5 Discussion

In this paper, we studied the role of subsampling step and tree depth in Breiman's forest procedure. By analysing a simple version of random forests, we show that the performance of fully grown subsampled forests and that of pruned forests with no subsampling step are similar, provided a proper tuning of the parameter of interest (subsample size and tree depth respectively).

The extended experiments have shown similar results: Breiman's forests can be outperformed by either subsampled or pruned Breiman's forests by properly tuning parameters. Noteworthy, tuning tree depth can be done at almost no additional cost while running Breiman's forests (due to the intrinsic recursive nature of forests). However if one is interested in a faster procedure, subsampled Breiman's forests are to be preferred to pruned forests.

As a by-product, our analysis also shows that there is no particular interest in bootstrapping data instead of subsampling: in our experiments, bootstrap is comparable (or worse) than subsampling. This sheds some light on several previous theoretical analysis where the bootstrap step was replaced by subsampling, which is more amenable to analyse. Similarly, proving theoretical results on fully grown Breiman's forests turned out to be extremely difficult. Our analysis shows that there is no theoretical background for considering Breiman's forests with default parameters values instead of pruned or subsampled Breiman's forests, which reveal themselves to be easier to examine.

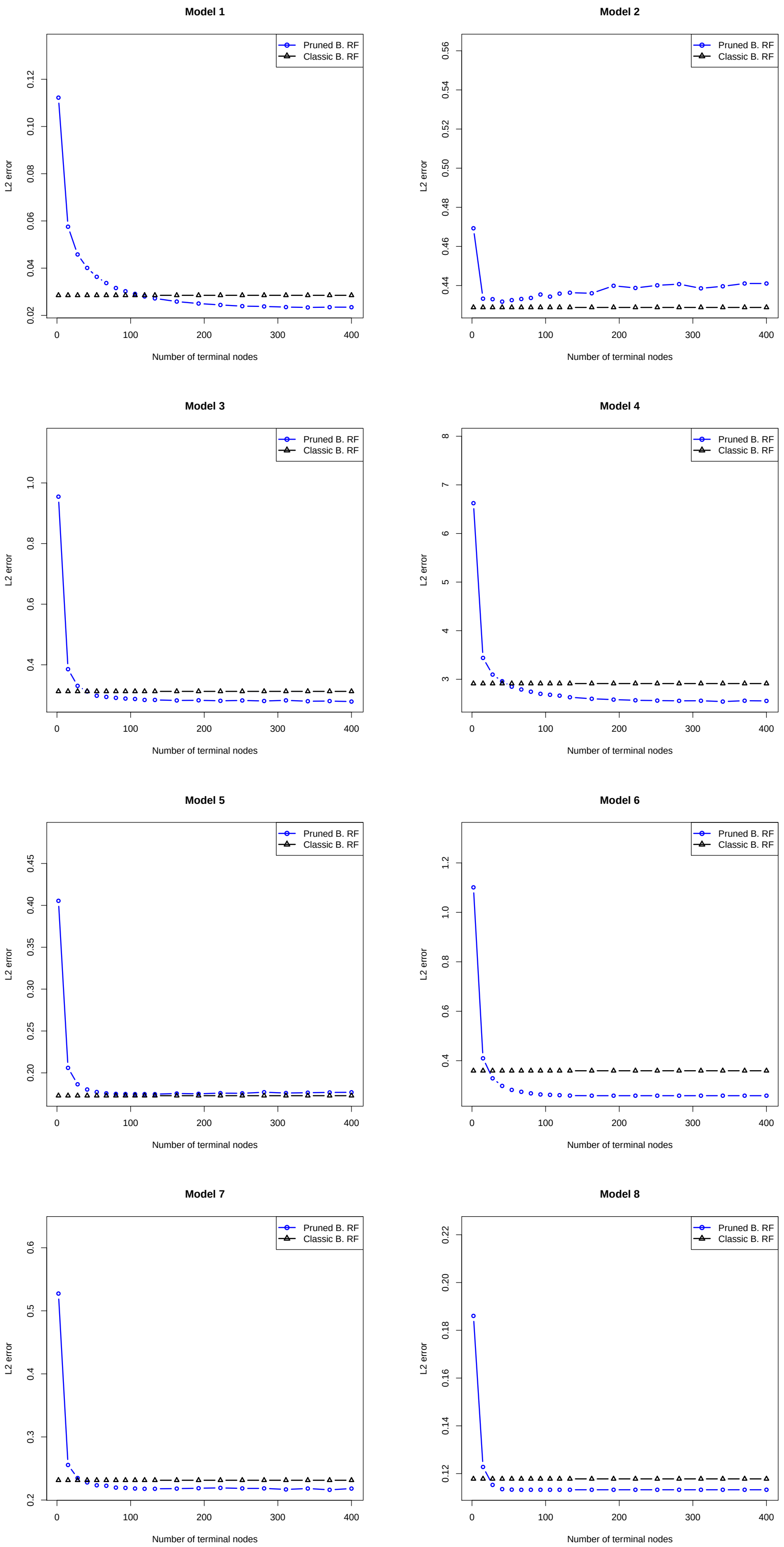

Figure 1: Comparison of standard Breiman's forests (B. RF) against pruned Breiman's forests in terms of $\mathbb{L}^2$ error.

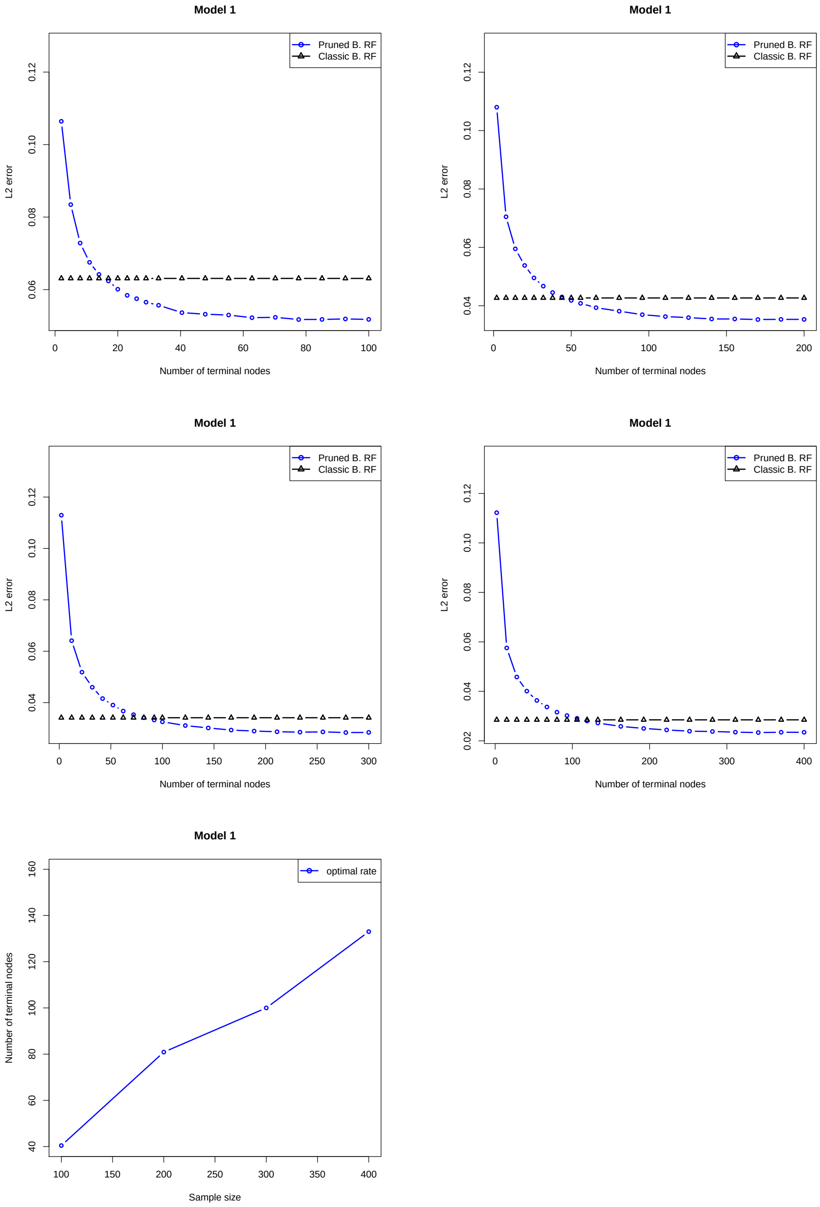

Figure 2: Tuning of pruning parameter (**Model 1**).

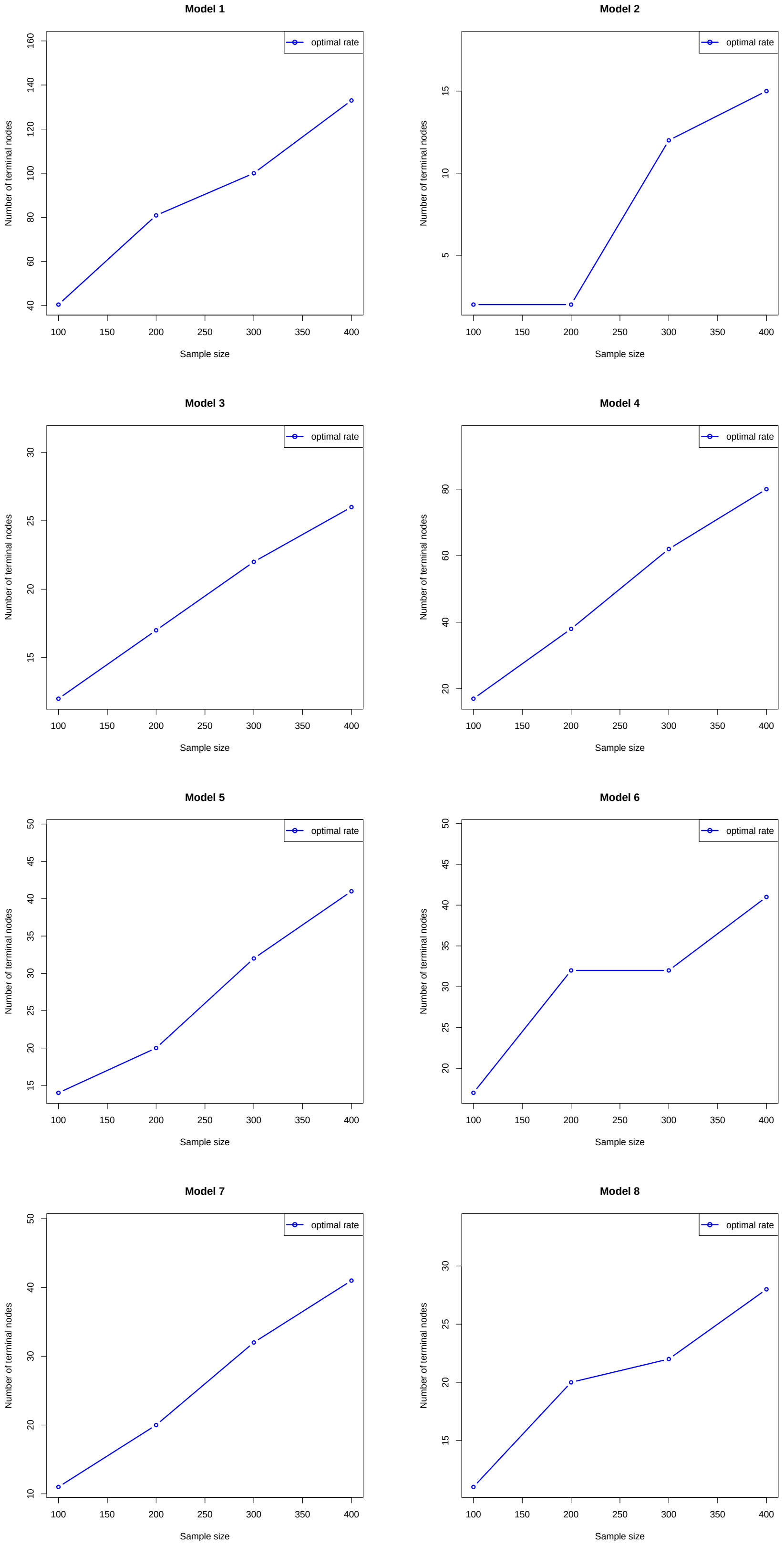

Figure 3: Optimal values of pruning parameter for **Models 1-8**.

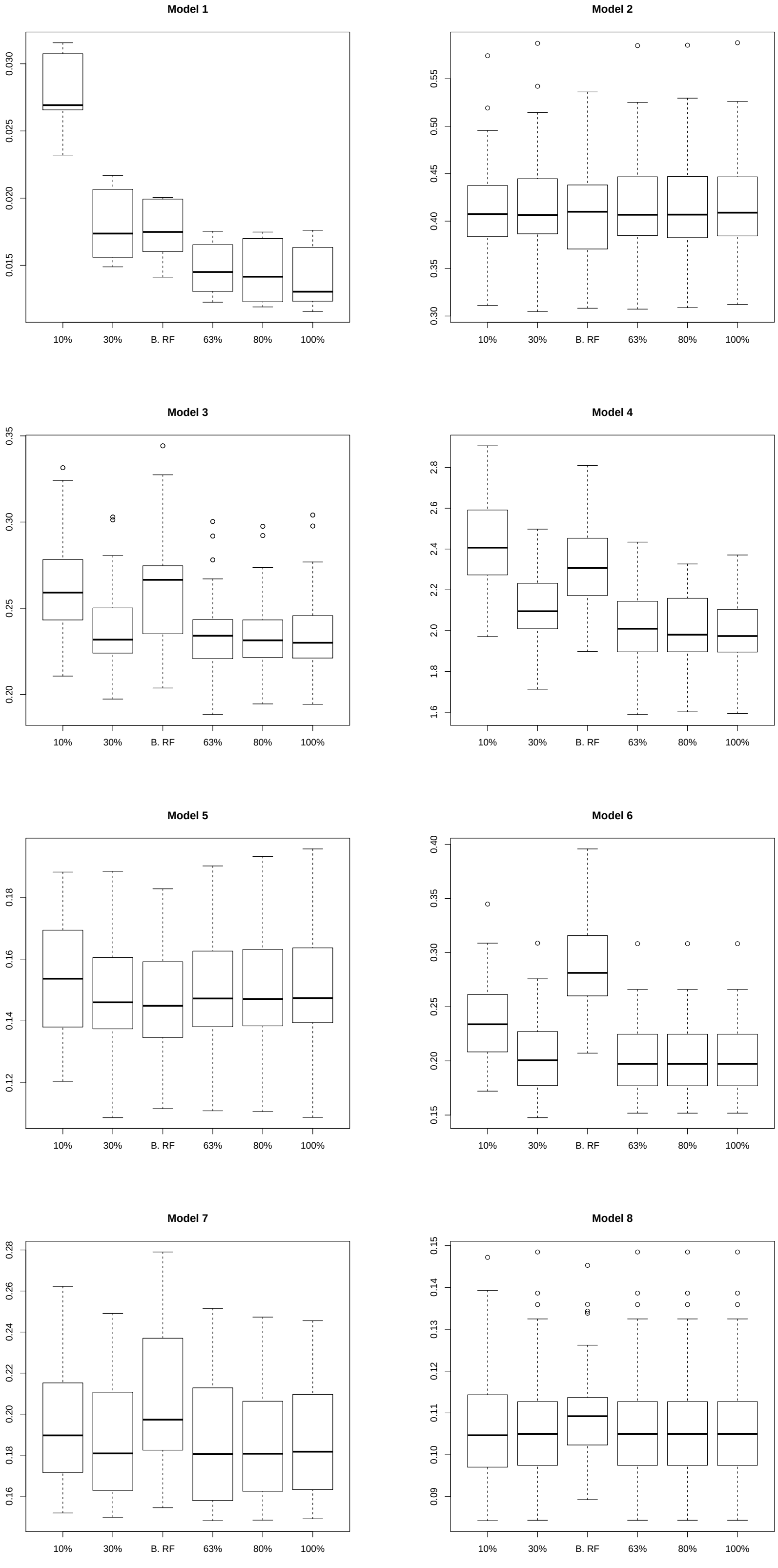

Figure 4: Comparison of standard Breiman's forests against several pruned Breiman forests in terms of $\mathbb{L}^2$ error.

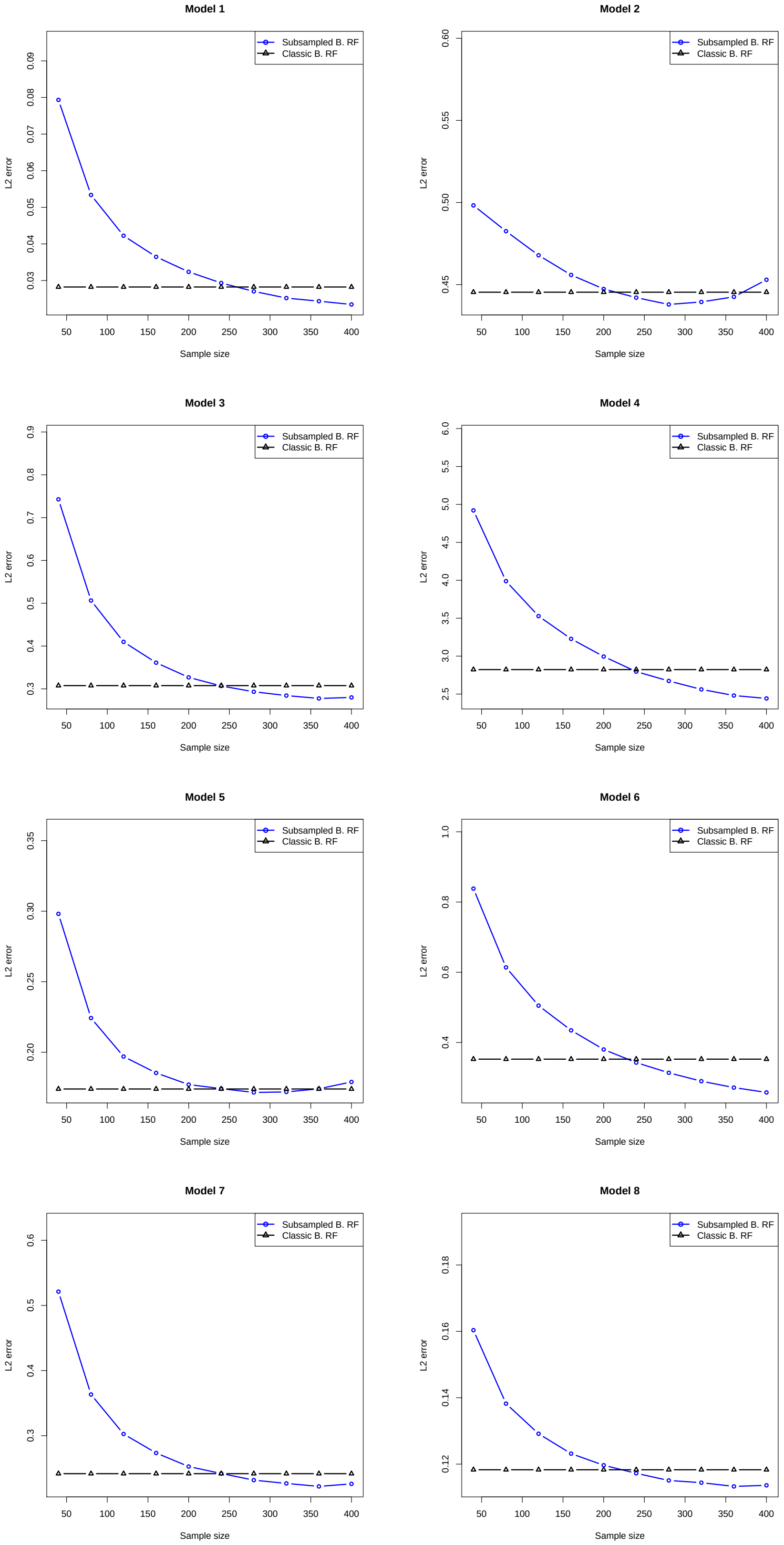

Figure 5: Standard Breiman Forests versus Subsampled Breiman Forests.

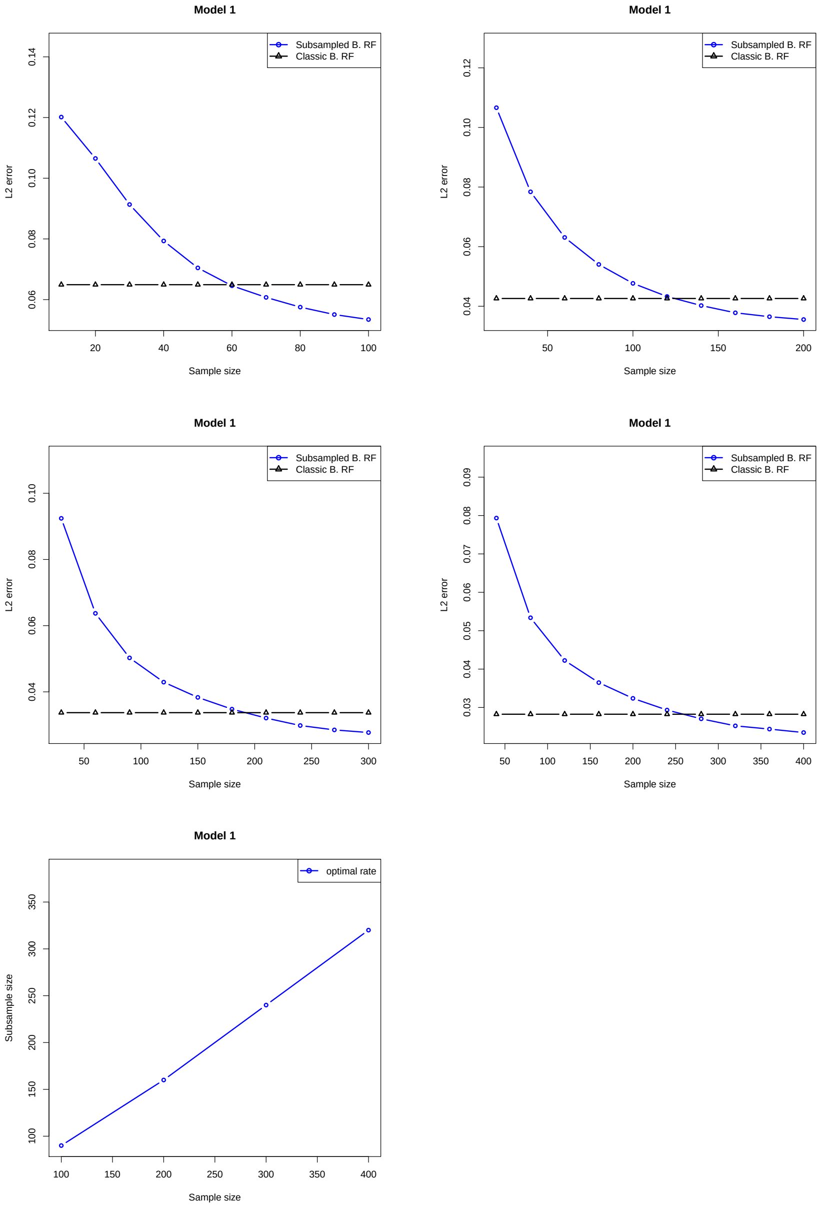

Figure 6: Tuning of subsampling rate (model 1).

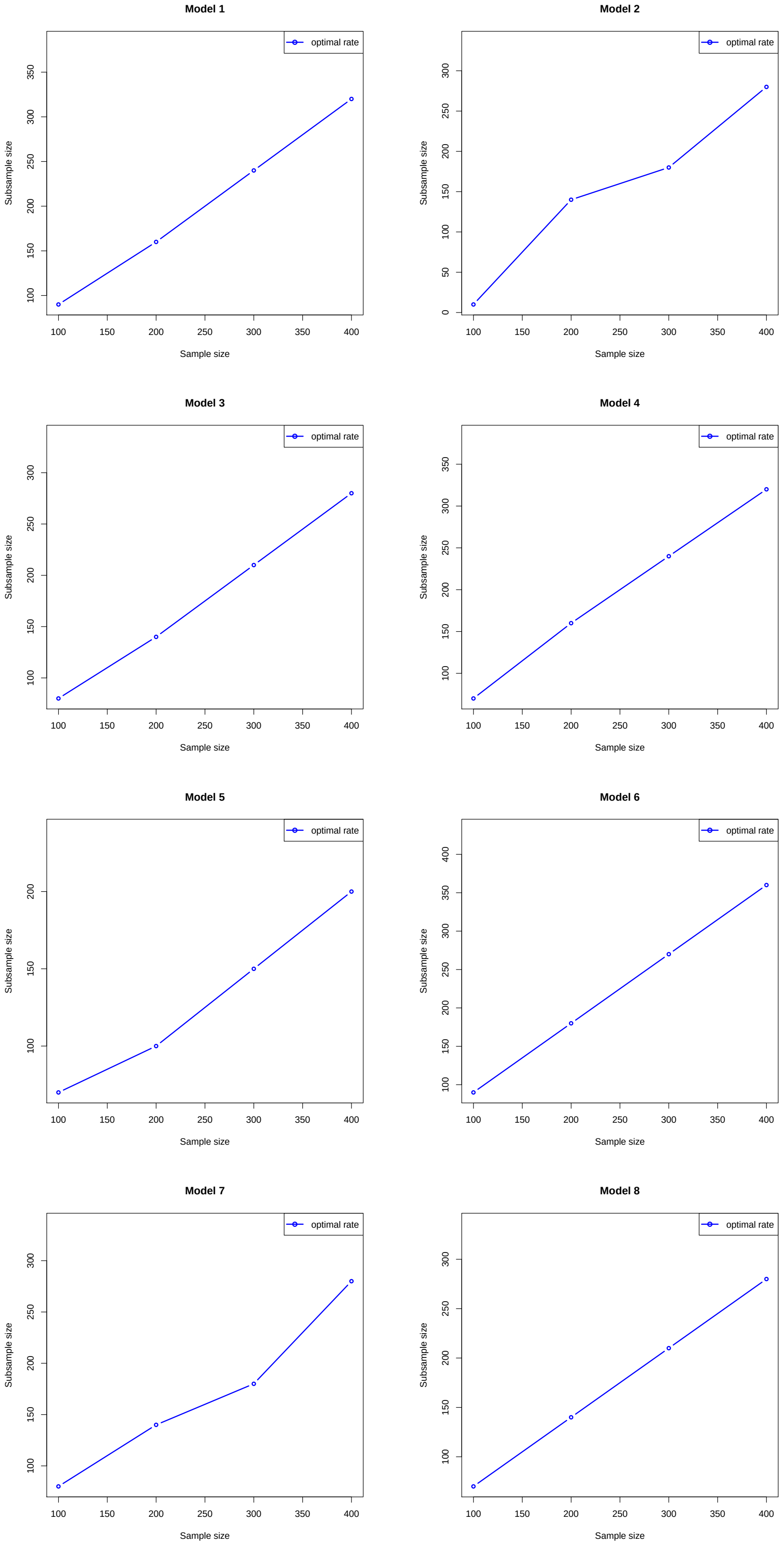

Figure 7: Optimal values of pruning parameter.

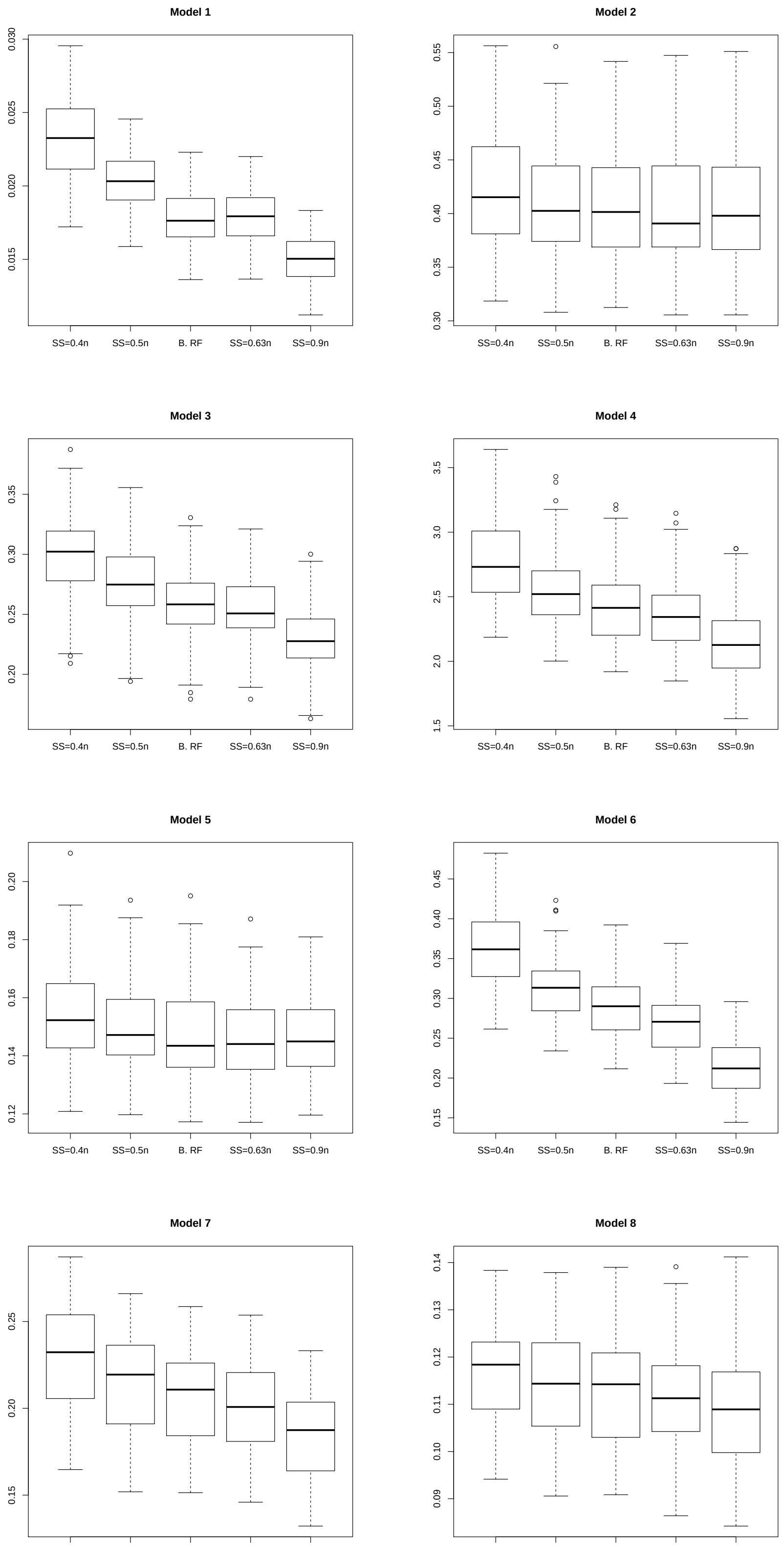

Figure 8: Standard Breiman forests versus several pruned Breiman forests.

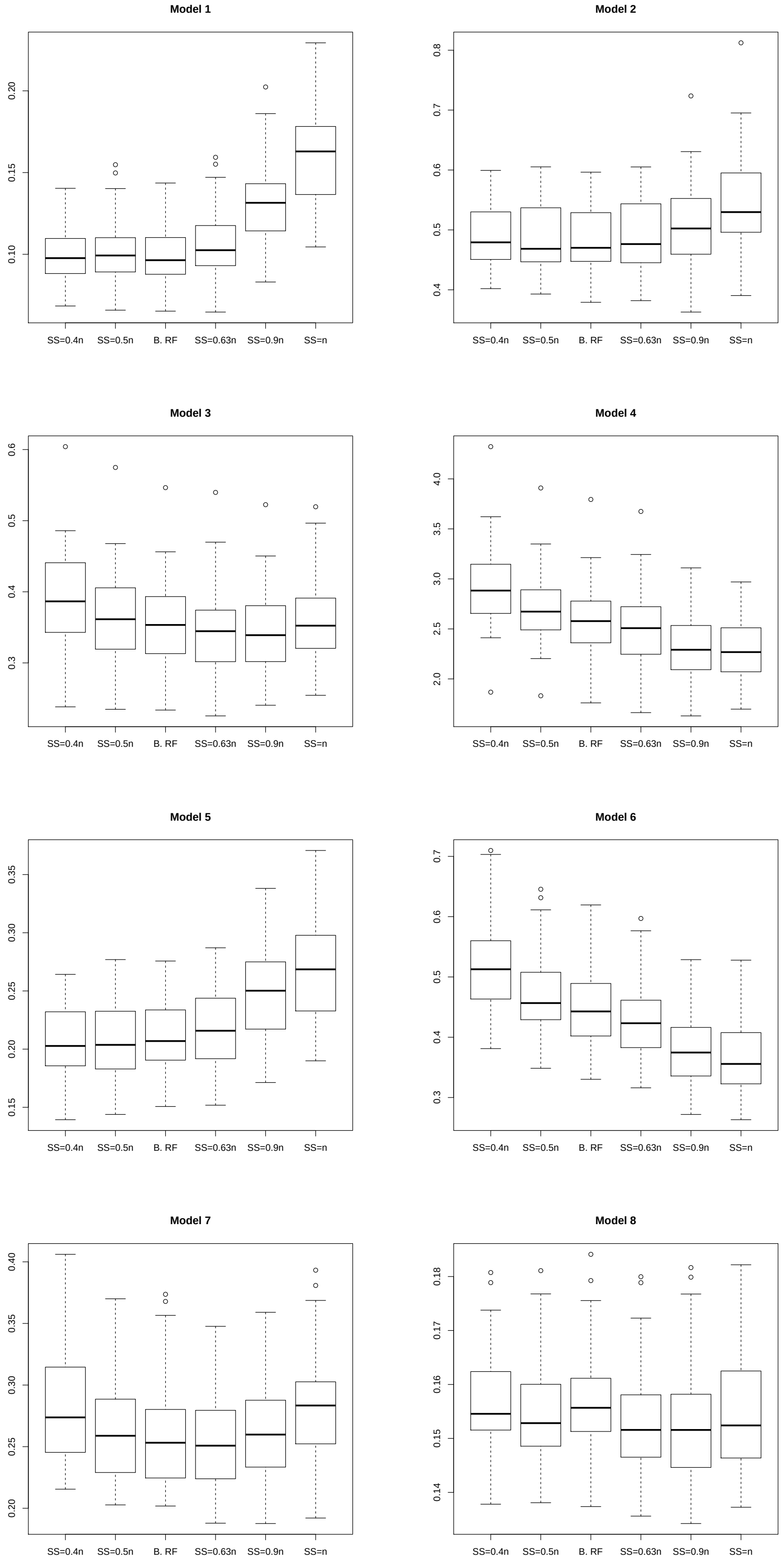

Figure 9: Standard Breiman forests versus several pruned Breiman forests (noisy models).

6 Proofs

Proof of Theorem 3.1. Let us start by recalling that the random forest estimate m_n can be written as a local averaging estimate

$$m_{\infty,n}(\mathbf{x}) = \sum_{i=1}^{n} W_{ni}(\mathbf{x}) Y_i,$$

where

$$W_{ni}(\mathbf{x}) = \frac{\mathbb{1}_{\mathbf{X}_i \overset{\Theta}{\leftrightarrow} \mathbf{x}}}{N_n(\mathbf{x}, \Theta)}.$$

The quantity $\mathbb{1}_{\mathbf{X}_i \overset{\Theta}{\leftrightarrow} \mathbf{x}}$ indicates whether the observation $\mathbf{X}_i$ is in the cell of the tree which contains $\mathbf{x}$ or not, and $N_n(\mathbf{x}, \Theta)$ denotes the number of data points falling in the same cell as $\mathbf{x}$. The $\mathbb{L}^2$-error of the forest estimate takes then the form

$$\begin{aligned}
\mathbb{E}\big[m_{\infty,n}(\mathbf{x}) - m(\mathbf{x})\big]^2 &\leq 2\mathbb{E}\Bigg[\sum_{i=1}^{n} W_{ni}(\mathbf{x})(Y_i - m(\mathbf{X}_i))\Bigg]^2 \\
&\quad + 2\mathbb{E}\Bigg[\sum_{i=1}^{n} W_{ni}(\mathbf{x})(m(\mathbf{X}_i) - m(\mathbf{x}))\Bigg]^2 \\
&= 2I_n + 2J_n.
\end{aligned}$$

We can identify the term I_n as the estimation error and J_n as the approximation error, and then work on each term I_n and J_n separately.

Approximation error. Let $A_n(\mathbf{x}, \Theta)$ be the cell containing $\mathbf{x}$ in the tree built with the random parameter Θ. Regarding J_n, by the Cauchy Schwartz inequality,

$$\begin{aligned}
J_n &\leq \mathbb{E}\Bigg[\sum_{i=1}^{n} \sqrt{W_{ni}(\mathbf{x})}\sqrt{W_{ni}(\mathbf{x})}|m(\mathbf{X}_i) - m(\mathbf{x})|\Bigg]^2 \\
&\leq \mathbb{E}\Bigg[\sum_{i=1}^{n} W_{ni}(\mathbf{x})(m(\mathbf{X}_i) - m(\mathbf{x}))^2\Bigg] \\
&\leq \mathbb{E}\Bigg[\sum_{i=1}^{n} \frac{\mathbb{1}_{\mathbf{X}_i \overset{\Theta}{\leftrightarrow} \mathbf{x}}}{N_n(\mathbf{x}, \Theta)} \sup_{\substack{\mathbf{x},\mathbf{z},\\ |\mathbf{x}-\mathbf{z}| \leq \mathrm{diam}(A_n(\mathbf{x}))}} |m(\mathbf{x}) - m(\mathbf{z})|^2\Bigg] \\
&\leq L^2 \mathbb{E}\Bigg[\frac{1}{N_n(\mathbf{x}, \Theta)} \sum_{i=1}^{n} \mathbb{1}_{\mathbf{X}_i \overset{\Theta}{\leftrightarrow} \mathbf{x}} (\mathrm{diam}(A_n(\mathbf{x}, \Theta)))^2\Bigg] \\
&\leq L^2 \mathbb{E}\Bigg[(\mathrm{diam}(A_n(\mathbf{x}, \Theta)))^2\Bigg],
\end{aligned}$$

where the fourth inequality is due to the L-Lipschitz continuity of m. Let $V_\ell(\mathbf{x}, \Theta)$ be the length of the cell containing $\mathbf{x}$ along the ℓ-th side. Then,

$$J_n \leq L^2 \sum_{l=1}^{d} \mathbb{E}\Big[V_l(\mathbf{x}, \Theta)^2\Big].$$

According to Lemma 1 specified further, we have

$$\mathbb{E}\Big[V_l(\mathbf{x},\Theta)^2\Big] \leq C\Big(1-\frac{3}{4d}\Big)^k,$$

with $C=\exp(12/(4d-3))$. Thus, for all k, we have

$$J_n \leq dL^2C\Big(1-\frac{3}{4d}\Big)^k.$$

Estimation error. Let us now focusing on the term I_n, we have

$$\begin{aligned}
I_n &= \mathbb{E}\Big[\sum_{i=1}^n W_{ni}(\mathbf{x})(Y_i - m(\mathbf{X}_i))\Big]^2\\
&= \sum_{i=1}^n\sum_{i=1}^n \mathbb{E}\Big[W_{ni}(\mathbf{x})W_{nj}(\mathbf{x})(Y_i - m(\mathbf{X}_i))(Y_j - m(\mathbf{X}_j))\Big]\\
&= \mathbb{E}\Big[\sum_{i=1}^n W_{ni}^2(\mathbf{x})(Y_i - m(\mathbf{X}_i))^2\Big]\\
&\leq \sigma^2\mathbb{E}\Big[\max_{1\leq i\leq n} W_{ni}(\mathbf{x})\Big],
\end{aligned}$$

since, by **(H)**, the variance of ε_i is bounded above by σ^2. Recalling that a_n is the number of subsampled observations used to build the tree, we can note that

$$\begin{aligned}
\mathbb{E}\Big[\max_{1\leq i\leq n} W_{ni}(\mathbf{x})\Big] &= \mathbb{E}\Big[\max_{1\leq i\leq n} \mathbb{E}_\Theta\Big[\frac{\mathbb{1}_{\mathbf{x}\overset{\Theta}{\leftrightarrow}\mathbf{X}_i}}{N_n(\mathbf{x},\Theta)}\Big]\Big]\\
&\leq \frac{1}{\frac{a_n}{2^k}-2}\mathbb{E}\Big[\max_{1\leq i\leq n}\mathbb{P}_\Theta\Big[\mathbf{x}\overset{\Theta}{\leftrightarrow}\mathbf{X}_i\Big]\Big].
\end{aligned}$$

Observe that in the subsampling step, there are exactly $\binom{a_n-1}{n-1}$ choices to pick a fixed observation $\mathbf{X}_i$. Since $\mathbf{x}$ and $\mathbf{X}_i$ belong to the same cell only if $\mathbf{X}_i$ is selected in the subsampling step, we see that

$$\mathbb{P}_\Theta\Big[\mathbf{x}\overset{\Theta}{\leftrightarrow}\mathbf{X}_i\Big] \leq \frac{\binom{a_n-1}{n-1}}{\binom{a_n}{n}} = \frac{a_n}{n}.$$

So,

$$I_n \leq \sigma^2\frac{1}{\frac{a_n}{2^k}-2}\frac{a_n}{n} \leq 2\sigma^2\frac{2^k}{n},$$

since $a_n/2^k \geq 4$. Consequently, we obtain

$$\mathbb{E}\big[m_{\infty,n}(\mathbf{x}) - m(\mathbf{x})\big]^2 \leq I_n + J_n \leq 2\sigma^2\frac{2^k}{n} + dL^2C\Big(1-\frac{3}{4d}\Big)^k.$$

□

We set up now Lemma 1 about the length of a cell that we used to bound the approximation error.

Lemma 1. *For all $\ell \in \{1, \ldots, d\}$ and $k \in \mathbb{N}^*$, we have*

$$\mathbb{E}\Big[V_l(\mathbf{x}, \Theta)^2\Big] \leq C\Big(1 - \frac{3}{4d}\Big)^k,$$

with $C = \exp(12/(4d-3))$.

Proof of Lemma 1. Let us fix $\mathbf{x} \in [0,1]^d$ and denote by $n_0, n_1, \ldots, n_k$ the number of points in the successive cells containing $\mathbf{x}$ (for example, n_0 is the number of points in the root of the tree, that is $n_0 = a_n$). Note that $n_0, n_1, \ldots, n_k$ depends on $\mathcal{D}_n$ and Θ, but to lighten notations, we omit these dependencies. Recalling that $V_\ell(\mathbf{x}, \Theta)$ is the length of the ℓ-th side of the cell containing $\mathbf{x}$, this quantity can be written as a product of independent beta distributions:

$$V_\ell(\mathbf{x}, \Theta) \overset{\mathcal{D}}{=} \prod_{j=1}^{k} \big[B(n_j + 1, n_{j-1} - n_j)\big]^{\delta_{\ell,j}(\mathbf{x},\Theta)},$$

where $B(\alpha, \beta)$ denotes the beta distribution of parameters α and β, and the indicator $\delta_{\ell,j}(\mathbf{x}, \Theta)$ equals to 1 if the j-th split of the cell containing $\mathbf{x}$ is performed along the ℓ-th dimension (and 0 otherwise). Consequently,

$$\begin{aligned}
\mathbb{E}\big[V_\ell(\mathbf{x}, \Theta)^2\big] &= \prod_{j=1}^{k} \mathbb{E}\Big[\big[B(n_j+1, n_{j-1}-n_j)\big]^{2\delta_{\ell,j}(\mathbf{x},\Theta)}\Big] \\
&= \prod_{j=1}^{k} \mathbb{E}\Big[\mathbb{E}\Big[\big[B(n_j+1, n_{j-1}-n_j)\big]^{2\delta_{\ell,j}(\mathbf{x},\Theta)} \big| \delta_{\ell,j}(\mathbf{x},\Theta)\Big]\Big] \\
&= \prod_{j=1}^{k} \mathbb{E}\Big[\mathbb{1}_{\delta_{\ell,j}(\mathbf{x},\Theta)=0} + \mathbb{E}\big[B(n_j+1, n_{j-1}-n_j)\big]^2 \mathbb{1}_{\delta_{\ell,j}(\mathbf{x},\Theta)=1}\Big] \\
&= \prod_{j=1}^{k} \left(\frac{d-1}{d} + \frac{1}{d}\mathbb{E}\big[B(n_j+1, n_{j-1}-n_j)\big]^2\right) \\
&= \prod_{j=1}^{k} \left(\frac{d-1}{d} + \frac{1}{d}\frac{(n_j+1)(n_j+2)}{(n_{j-1}+1)(n_{j-1}+2)}\right) \\
&\leq \prod_{j=1}^{k} \left(\frac{d-1}{d} + \frac{1}{4d}\frac{(n_{j-1}+2)(n_{j-1}+4)}{(n_{j-1}+1)(n_{j-1}+2)}\right) \\
&\leq \prod_{j=1}^{k} \left(1 - \frac{1}{d} + \frac{1}{4d}\frac{n_{j-1}+4}{n_{j-1}+1}\right),
\end{aligned} \tag{7}$$

where the first inequality stems from the relation $n_j \leq n_{j-1}/2$ for all $j \in$

$\{1,\ldots,k\}$. We have the following inequalities.

$$\begin{aligned}\frac{n_{j-1}+4}{n_{j-1}+1} \leq \frac{a_n+2^{j+1}}{a_n-2^{j-1}} &= \frac{a_n+2^{j+1}}{a_n(1-\frac{2^{j-1}}{a_n})}\\ &\leq \frac{a_n+2^{j+1}}{a_n}\left(1+\frac{2^{j-1}}{a_n}\frac{1}{1-\frac{2^{j-1}}{a_n}}\right)\\ &\leq \left(1+\frac{2^{j+1}}{a_n}\right)^2,\end{aligned}$$

since

$$\frac{2^{j-1}}{a_n} \leq \frac{2^{k-1}}{a_n} \leq \frac{1}{2}.$$

Going back to inequality (7), we find

$$\begin{aligned}\mathbb{E}\left[V_l(\mathbf{x},\Theta)^2\right] &\leq \prod_{j=1}^{k}\left[1-\frac{1}{d}+\frac{1}{4d}\left(1+\frac{2^{j+1}}{a_n}\right)^2\right]\\ &\leq \prod_{j=1}^{k}\left[1-\frac{3}{4d}+\frac{3}{d}\frac{2^{j-1}}{a_n}\right]\\ &\leq \prod_{j=1}^{k}\left[1-\frac{3}{4d}+\frac{3}{d}\frac{2^{k}}{a_n}2^{j-k}\right]\\ &\leq \prod_{j=0}^{k-1}\left[1-\frac{3}{4d}+\frac{3}{d}2^{-j-1}\right].\end{aligned}$$

Moreover, we can notice that

$$\begin{aligned}\ln\left(\prod_{j=0}^{k-1}\left[1-\frac{3}{4d}+\frac{3}{d}2^{-j-1}\right]\right) &= k\ln\left(1-\frac{3}{4d}\right)+\sum_{j=0}^{k-1}\ln\left(1+6\frac{2^{-j}}{4d-3}\right)\\ &\leq k\ln\left(1-\frac{3}{4d}\right)+\frac{12}{4d-3}.\end{aligned}$$

This yields to the desired upper bound

$$\mathbb{E}\left[V_l(\mathbf{x},\Theta)^2\right] \leq C\left(1-\frac{3}{4d}\right)^k,$$

with $C=\exp(12/(4d-3))$. □

We now put our interest in the proofs of the two Corollaries presented in Section 3.

Proof of Corollary 1. Regarding Theorem 3.1, we want to find the optimal value of pruning, in order to obtain the best rate of convergence for the forest estimate.

Let $C_1 = \frac{2\sigma^2}{n}$ and $C_2 = d^{\frac{3}{2}} L^2 C$ and $\beta = \left(1 - \frac{3}{4d}\right)$. Then,

$$\mathbb{E}\big[m_{\infty,n}(\mathbf{x}) - m(\mathbf{x})\big]^2 \leq C_1 2^k + C_2 \beta^k.$$

Let $f : x \mapsto C_1 e^{x \ln 2} + C_2 e^{x \ln(\beta)}$. Thus,

$$\begin{aligned} f'(x) &= C_1 \ln 2 e^{x \ln 2} + C_2 \ln(\beta) e^{x \ln(\beta)} \\ &= C_1 \ln 2 e^{x \ln 2} \left(1 + \frac{C_2 \ln(\beta)}{C_1 \ln 2} e^{x(\ln(\beta) - \ln 2)}\right). \end{aligned}$$

Since $\beta \leq 1$, $f'(x) \leq 0$ for all $x \leq x^\star$ and $f'(x) \geq 0$ for all $x \geq x^\star$, where $x^\star$ satisfies

$$\begin{aligned} & f'(x^\star) = 0 \\ \iff \quad & x^\star = \frac{1}{\ln 2 - \ln(\beta)} \ln\left(-\frac{C_2 \ln(\beta)}{C_1 \ln 2}\right) \\ \iff \quad & x^\star = \frac{1}{\ln 2 - \ln(\beta)} \left[\ln\left(\frac{1}{C_1}\right) + \ln\left(-\frac{C_2 \ln(\beta)}{\ln 2}\right)\right] \\ \iff \quad & x^\star = \frac{1}{\ln 2 - \ln\left(1 - \frac{3}{4d}\right)} \left[\ln(n) + \ln\left(-\frac{dL^2 C \ln\left(1 - \frac{3}{4d}\right)}{2\sigma^2 \ln 2}\right)\right] \\ \iff \quad & x^\star = \frac{1}{\ln 2 - \ln\beta} \Big[\ln(n) + C_3\Big], \end{aligned}$$

where $C_3 = \ln\left(-\frac{dL^2 C \ln\left(1 - \frac{3}{4d}\right)}{2\sigma^2 \ln 2}\right)$.

Consequently,

$$\begin{aligned} \mathbb{E}\big[m_{\infty,n}(\mathbf{x}) - m(\mathbf{x})\big]^2 &\leq C_1 \exp(x^\star \ln 2) + C_2 \exp(x^\star \ln\beta) \\ &\leq C_1 \exp\left(\frac{1}{\ln 2 - \ln\beta}\Big[\ln(n) + C_3\Big] \ln 2\right) \\ &\quad + C_2 \exp\left(\frac{1}{\ln 2 - \ln\beta}\Big[\ln(n) + C_3\Big] \ln\beta\right) \\ &\leq C_1 \exp\left(\frac{C_3 \ln 2}{\ln 2 - \ln\beta}\right) \exp\left(\frac{\ln 2}{\ln 2 - \ln\beta} \ln(n)\right) \\ &\quad + C_2 \exp\left(\frac{C_3 \ln\beta}{\ln 2 - \ln\beta}\right) \exp\left(\frac{\ln\beta}{\ln 2 - \ln\beta} \ln(n)\right) \\ &\leq C_5 n^{\frac{\ln 2}{\ln 2 - \ln\beta} - 1} + C_6 n^{\frac{\ln\beta}{\ln 2 - \ln\beta}} \\ &\leq \Big(C_5 + C_6\Big) n^{\frac{\ln\left(1 - \frac{3}{4d}\right)}{\ln 2 - \ln\left(1 - \frac{3}{4d}\right)}}, \end{aligned}$$

where $C_5 = 2\sigma^2 \exp\left(\frac{C_3 \ln 2}{\ln 2 - \ln\beta}\right)$ and $C_6 = C_2 \exp\left(\frac{C_3 \ln\beta}{\ln 2 - \ln\beta}\right)$. □

Proof of Corollary 2. In this Corollary, we focus on the optimal value of subsampling, always for the speed of convergence for the forest estimate. Since $k_n = \log_2(a_n) - 2$ and k_n satisfies equation (3), we have

$$a_n = 4.2^{\frac{C_3}{\ln 2 - \ln \beta}}.n^{\frac{\ln 2}{\ln 2 - \ln \beta}},$$

where, simple calculations show that

$$2^{\frac{C_3}{\ln 2 - \ln \beta}} = \left(\frac{3L^2 e^{12/(4d-3)}}{8\sigma^2 \ln 2}\right)^{\frac{\ln 2}{\ln 2 - \ln \beta}}.$$

This concludes the proof, according to Theorem 3.1. □

References

S. Arlot and R. Genuer. Analysis of purely random forests bias. arXiv:1407.3939, 2014.

G. Biau. Analysis of a random forests model. *Journal of Machine Learning Research*, 13:1063–1095, 2012.

G. Biau and L. Devroye. Cellular tree classifiers. In *Algorithmic Learning Theory*, pages 8–17. Springer, 2014.

G. Biau, L. Devroye, and G. Lugosi. Consistency of random forests and other averaging classifiers. *Journal of Machine Learning Research*, 9:2015–2033, 2008.

L. Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone. *Classification and Regression Trees*. Chapman & Hall/CRC, Boca Raton, 1984.

M. Denil, D. Matheson, and N. de Freitas. *Consistency of online random forests*, 2013. arXiv:1302.4853.

L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer, New York, 1996.

R. Díaz-Uriarte and S. Alvarez de Andrés. Gene selection and classification of microarray data using random forest. *BMC Bioinformatics*, 7:1–13, 2006.

R. Genuer. Variance reduction in purely random forests. *Journal of Nonparametric Statistics*, 24:543–562, 2012.

R. Genuer, J. Poggi, and C. Tuleau-Malot. Variable selection using random forests. *Pattern Recognition Letters*, 31:2225-2236, 2010.

H. Ishwaran and U.B. Kogalur. Consistency of random survival forests. *Statistics & Probability Letters*, 80:1056–1064, 2010.

L. Meier, S. Van de Geer, and P. Bühlmann. High-dimensional additive modeling. *The Annals of Statistics*, 37:3779–3821, 2009.

L. Mentch and G. Hooker. Ensemble trees and clts: Statistical inference for supervised learning. arXiv:1404.6473, 2014.

Y. Qi. *Ensemble Machine Learning*, chapter Random forest for bioinformatics, pages 307–323. Springer, 2012.

G. Rogez, J. Rihan, S. Ramalingam, C. Orrite, and P. H. Torr. Randomized trees for human pose detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

E. Scornet. On the asymptotics of random forests. arXiv:1409.2090, 2014.

E. Scornet, G. Biau, and J.-P. Vert. Consistency of random forests. *The Annals of Statistics*, 43:1716–1741, 2015.

C.J. Stone. Optimal rates of convergence for nonparametric estimators. *The Annals of Statistics*, 8:1348–1360, 1980.

C.J. Stone. Optimal global rates of convergence for nonparametric regression. *The Annals of Statistics*, 10:1040–1053, 1982.

M. van der Laan, E.C. Polley, and A.E. Hubbard. Super learner. *Statistical Applications in Genetics and Molecular Biology*, 6, 2007.

S. Wager. Asymptotic theory for random forests. arXiv:1405.0352, 2014.