IMPROVING THE ACCURACY-MEMORY TRADE-OFF OF RANDOM FORESTS VIA LEAF-REFINEMENT

Sebastian Buschjäger ¹  Katharina Morik ¹

ABSTRACT

Random Forests (RF) are among the state-of-the-art in many machine learning applications. With the ongoing integration of ML models into everyday life, the deployment and continuous application of models becomes more and more an important issue. Hence, small models which offer good predictive performance but use small amounts of memory are required. Ensemble pruning is a standard technique to remove unnecessary classifiers from an ensemble to reduce the overall resource consumption and sometimes even improve the performance of the original ensemble. In this paper, we revisit ensemble pruning in the context of ‘modernly’ trained Random Forests where trees are very large. We show that the improvement effects of pruning diminishes for ensembles of large trees but that pruning has an overall better accuracy-memory trade-off than RF. However, pruning does not offer fine-grained control over this trade-off because it removes entire trees from the ensemble. To further improve the accuracy-memory trade-off we present a simple, yet surprisingly effective algorithm that refines the predictions in the leaf nodes in the forest via stochastic gradient descent. We evaluate our method against 7 state-of-the-art pruning methods and show that our method outperforms the other methods on 11 of 16 datasets with a statistically significant better accuracy-memory trade-off compared to most methods. We conclude our experimental evaluation with a case study showing that our method can be applied in a real-world setting.

1 I NTRODUCTION

Ensemble algorithms offer state-of-the-art performance in many applications and often outperform single classifiers by a large margin. With the ongoing integration of embedded systems and machine learning models into our everyday life, e.g. in the form of the Internet of Things, the hardware platforms which execute ensembles must also be taken into account when training ensembles. From a hardware perspective, a small ensemble with minimal execution time and a small memory footprint is desired. Similar, learning theory indicates that ensembles of small models should generalize better which would make them ideal candidates for small, resource constraint devices (Koltchinskii et al., 2002; Cortes et al., 2014). Practical problems, on the other hand, often require ensembles of complex base learners to achieve good results. For some ensembling techniques such as Random Forest it is even desired that individual trees are as large as possible leading to overall large ensembles (Breiman, 2000; Biau, 2012; Denil et al., 2014; Biau & Scornet, 2016). Ensemble pruning is a standard technique for implementing ensembles on small devices (Tsoumakas et al., 2009; Zhang et al., 2006) by removing unnecessary classifiers from the ensemble. Remarkably, this removal can sometimes lead to a better predictive performance (Margineantu & Dietterich, 1997; Martínez-Muñoz & Suárez, 2006; Li et al., 2012). In this paper, we revisit ensemble pruning and show that this improvement effect does not carry over to the modern-style of training individual trees as large as possible in Random Forests. Maybe even more frustrating, ensemble pruning does not seem to be necessary anymore to achieve the best accuracy if the original forest has a sufficient amount of large trees. If, however, one also considers the memory requirements of the individual trees the situation changes. We argue that, from a hardware perspective, the trade-off between memory and accuracy is what really matters. Although a Random Forest might produce a good model it might not be possible to deploy it onto a small device due to its memory requirements. As shown later, the best-performing RF models are often larger than $5 - 10$ MB (see e.g. Fig. 2 and Fig. 3) while most available microcontroller units (MCU) only offer a few KB to a few MB of memory as depicted in Table 1. Hence, to deploy RF onto these small devices we require a good algorithm which gives accurate models for a variety of different memory constraints.

We directly optimize the accuracy-memory trade-off by introducing a technique called leaf-refinement. Leaf-Refinement is a simple, but surprisingly effective method, which, instead of removing trees from the ensemble, further refines the predictions of small ensembles using gradient-descent. This way, we can refine any given tree-ensemble

¹Chair for Artificial Intelligence, TU Dortmund University, Germany. Correspondence to: Sebastian Buschjäger <sebastian.buschjaeger@tu-dortmund.de>.
Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

| MCU          | Flash | (S)RAM | Power |
|--------------|-------|--------|-------|
| Arduino Uno  | 32KB  | 2KB    | 12mA  |
| Arduino Mega | 256KB | 8KB    | 6mA   |
| Arduino Nano | 26–32KB | 1–2KB | 6mA   |
| STM32L0      | 192KB | 20KB   | 7mA   |
| Arduino MKR1000 | 256KB | 32KB   | 4mA   |
| Arduino Due  | 512KB | 96KB   | 50mA  |
| STM32F2      | 1MB   | 128KB  | 21mA  |
| STM32F4      | 2MB   | 384KB  | 50mA  |

Table 1. Available memory on different microcontroller units. Excerpt from (Branco et al., 2019).

to optimize its accuracy thereby maximizing the accuracy-memory trade-off. Our contributions are as follows:

- **Revisiting ensemble pruning:** We revisit ensemble pruning in the context of modernly trained Random Forests in which individual trees are typically large. We show that pruning a Random Forest can improve the accuracy if individual trees are small, but this effect becomes negligible for larger trees. Moreover, if we are only interested in the most accurate models where memory is no constraint we can simply train unpruned Random Forests which yields comparable results without the need for pruning.

- **Random Forest with Leaf Refinement:** We show that pruning exhibits a better accuracy-memory trade-off than RF does. To further optimize this trade-off we present a simple, yet surprisingly effective gradient-descent based algorithm called leaf-refinement (RF-LR) which refines the predictions of a pre-trained Random Forest.

- **Experiments:** We show the performance of our algorithm on 16 datasets and compare it against 7 state-of-the-art pruning methods. We show that RF-LR outperforms the other methods on 11 of 16 datasets with a statistically significant better accuracy-memory trade-off compared to most methods. We conclude our experimental evaluation with a case study showing that our method can be applied on a real-world setting.

The paper is organized as the following. Section 2 presents our notation and related work. In Section 3 we revisit ensemble pruning in the context of 'modern' Random Forests, whereas section 4 discusses how to improve the accuracy-memory trade-off without ensemble pruning. In section 5 we experimentally evaluate our method and in section 6 we conclude the paper.

## 2 Background and Notation

We consider a supervised learning setting, in which we assume that training and test points are drawn i.i.d. according to some distribution \( \mathcal{D} \) over the input space \( \mathcal{X} \) and labels \( \mathcal{Y} \).

We assume that we have given a trained ensemble with \( M \) classifiers \( h_i \in \mathcal{H} \) of the following form:

\[
f(x) = \frac{1}{M} \sum_{i=1}^{M} h_i(x)
\]

Additionally, we have given a labeled pruning sample \( \mathcal{S} = \{ (x_i, y_i) | i = 1, \ldots, N \} \) where \( x_i \in \mathcal{X} \subseteq \mathbb{R}^d \) is a \( d \)-dimensional feature-vector and \( y_i \in \mathcal{Y} \subseteq \mathbb{R}^C \) is the corresponding target vector. This sample can either be the original training data used to train \( f \) or another pruning set not related to the training or test data. For classification problems with \( C \geq 2 \) classes we encode each label as a one-hot vector \( y = (0, \ldots, 0, 1, 0, \ldots, 0) \) which contains a '1' at coordinate \( c \) for label \( c \in \{0, \ldots, C-1\} \); for regression problems we have \( C = 1 \) and \( \mathcal{Y} = \mathbb{R} \). In this paper, we will focus on classification problems, but note that our approach is directly applicable for regression tasks, as well. Moreover we will focus on tree ensembles and specifically Random Forests, but note that most of our discussion directly translates to other tree ensembles such as Bagging (Breiman, 1996), ExtraTrees (Geurts et al., 2006), Random Subspaces (Ho, 1998) or Random Patches (Louppe & Geurts, 2012).

The goal of ensemble pruning is to select a subset of \( K \) classifier from \( f \) which forms a small and accurate sub-ensemble. Formally, each classifier \( h_i \) receives a corresponding pruning weight \( w_i \in \{0, 1\} \). Let

\[
L(w) = \frac{1}{N} \sum_{(x, y) \in \mathcal{S}} \ell \left( \sum_{i=1}^{M} w_i h_i(x), y \right)
\]

be a loss function and let \( \|w\|_0 = \sum_{i=1}^{M} 1 \{w_i > 0\} \) be the \( l_0 \) norm which counts the number of nonzero entries in the weight vector \( w = (w_1, w_2, \ldots, w_M) \). Then the ensemble pruning problem is defined as:

\[
\arg \min_{w \in \{0, 1\}^M} L(w) \text{ st. } \|w\|_0 = K
\]

Many effective ensemble pruning methods have been proposed in literature. These methods usually differ in the specific loss function used to measure the performance of a sub-ensemble and the way this loss is minimized. Tsoumakas et al. give in (Tsoumakas et al., 2009) a detailed taxonomy of pruning methods which was later expanded in (Zhou, 2012) to which we refer interested readers. Early works on ensemble pruning focus on ranking-based approaches which assign a rank to each classifier depending on their individual performance and then pick the top \( K \) classifier.
from the ranking. One of the first pruning methods in this direction was due toMarginanteau and Dietterich which proposed to use the Cohen-Kappa statistic to rate the effectiveness of each classifier in (Marginanteau & Dietterich, 1997). More recent approaches also incorporate the ensemble’s diversity into the selection such as (Lu et al., 2010; Jiang et al., 2017; Guo et al., 2018). An alternative to a simple ranking, Mixed Quadratic Integer Programming (MQIP) has also been proposed. Originally this approach was proposed by Zhang et al. in (Zhang et al., 2006) which uses the pairwise errors of each classifier to formulate an MQIP. Cavalcanti et al. expand on this idea in (Cavalcanti et al., 2016) which combines 5 different measures into the MQIP. A third branch of pruning considers the clustering of ensemble members to promote diversity. The main idea is to cluster the classifiers into (diverse) groups and then to select one representative from each group (Giacinto et al., 2000; Lazarevic & Obradovic, 2001). Last, ordering-based pruning has been proposed. Ordering-based approaches order all ensemble members according to their overall contribution to the (sub)-ensemble and then pick the top $K$ classifier from this list. This approach was also first considered in (Marginanteau & Dietterich, 1997) which proposed to greedily minimize the overall ensemble error. A series of works by Martínez-Muñoz, Suárez and others (Martínez-Munoz & Suárez, 2004; Martínez-Muñoz & Suárez, 2006; Martínez-Muñoz et al., 2008) add upon this work proposing different error measures. More recently, theoretical insights from PAC theory and the bias-variance decomposition were also transformed into greedy pruning approaches (Li et al., 2012; Jiang et al., 2017).

Looking beyond ensemble pruning there are numerous, orthogonal methods to deploy ensembles to small devices. First, ‘classic’ decision tree pruning algorithms (e.g. minimal cost complexity pruning or sample complexity pruning) already reduce the size of DTs while offering a better accuracy (c.f. (Barros et al., 2015)). Second, in the context of model compression (see e.g. (Choudhary et al., 2020) for an overview) specific models such as Bonsai (Kumar et al., 2017) or Decision Jungles (Shotton et al., 2013) aim to find smaller tree ensembles already during training. Last, the optimal implementation of tree ensembles has also been studied, e.g. by optimizing the memory layout for caching (Buschjäger et al., 2018) or changing the tree traversal to utilize SIMD instructions (Ye et al., 2018). We find that all these methods are orthogonal to our approach and that they can be freely combined with one another, e.g. we may train a decision jungle, then perform ensemble pruning or leaf-refinement on it and finally find the optimal memory layout of the trees in the jungle for the best deployment.

### Algorithm 1 Reduced Error Pruning (RE).

1: $w \leftarrow (0, \ldots, 0)$
2: $i \leftarrow \arg \min \{L(w + \epsilon_i)|i = 1, \ldots, M\}$
3: $w \leftarrow w + \epsilon_i$
4: for $j = 1, \ldots, K - 1$ do
5: $i \leftarrow \arg \min \{L(w + \epsilon_i)|i = 1, \ldots, M, w_i \neq 1\}$
6: $w \leftarrow w + \epsilon_i$
7: end for

### 3 REVISITING ENSEMBLE PRUNING

Before we discuss our method we first want to revisit Reduced Error Pruning (RE, (Marginanteau & Dietterich, 1997)) and repeat some experiments performed with it. RE pruning is arguably one of the simplest pruning algorithms but often offers competitive performance. RE is a ordering-based pruning method. It starts with an empty ensemble and iteratively adds that tree which minimizes the overall ensemble error the most until $K$ members have been selected. Algorithm 1 depicts this approach where $L$ is the $0-1$ loss and $\epsilon_i$ denotes the unit vector with a ‘1’ entry at position $i$.

We will now perform experiments in the spirit of (Martínez-Muñoz & Suárez, 2006), but adapt a more modern approach to training the base ensembles. In the original experiments, the authors show that when pruning a Bagging Ensemble of 200 pruned CART trees, that RE (among other methods) achieves a better accuracy with fewer trees compared to the original ensemble. This result has been empirically reproduced in various contexts (see e.g. (Barros et al., 2015; Zhou et al., 2002; Zhou, 2012)) and has been formalized in the Many-Could-Be-Better-Than-All-Theorem (Zhou et al., 2002). It shows that the error of an ensemble excluding the $k$-th classifier can be smaller than the error of the original ensemble if the bias $C_{k,k}$ is larger than its variance wrt. to the ensemble:

$$\sum_{i=1, i \neq k}^{M} \sum_{j=1, i \neq k}^{M} \frac{C_{i,j}}{(M-1)^2} \leq \sum_{i=1}^{M} \sum_{h=1}^{M} \frac{C_{i,j}}{M^2} \quad (4)$$

$$\Leftrightarrow -2 \sum_{i=1, i \neq k}^{M} C_{i,k} \leq C_{k,k} \quad (5)$$

where

$$C_{k,k} = \mathbb{E}_{x,y \sim D} [(h_k(x) - y)^2] \quad (6)$$

$$C_{k,i} = \mathbb{E}_{x,y \sim D} [(h_k(x) - y)(h_i(x) - y)] \quad (7)$$

Recall that the bias of a DT rapidly decreases while the variance increases wrt. to the size of the tree ( Domingos, 2000). The original experiment used pruned decision trees whereas the today’s accepted standard is to train trees as large as possible for minimal errors (see (Breiman, 2000; Biau, 2012; Denil et al., 2014; Biau & Scornet, 2016) for...
more formal arguments on this). Hence, it is conceivable that ensemble pruning does not have the same beneficial effect on ‘modern’ Random Forests compared to RF-like ensembles trained 20 years ago. We will now investigate this hypothesis experimentally. As an example we will consider the EEG dataset which has 14,980 datapoints with 14 attributes and two classes (details for each dataset can be found in the appendix). By today’s standards this dataset is small to medium size which allows us to quickly train and evaluate different configurations but it is roughly two times larger than the biggest dataset used in original experiments. We perform experiments as follows: Oshiro et al. showed in (Oshiro et al., 2012) empirically on a variety of datasets that the prediction of a RF stabilizes between 128 and 256 trees and adding more trees to the ensemble does not yield significantly better results. Hence, we train the ‘base’ Random Forests with $M = 256$ trees. To control the individual errors of trees we set the maximum number of leaf nodes $n_l$ to values between $n_l \in \{64, 128, 256, 512, 1024\}$. For ensemble pruning we use RE which is tasked to select $K \in \{2, 4, 8, 16, 32, 64, 128, 256\}$ trees from the original RF. We compare this against a smaller RF with $K \in \{2, 4, 8, 16, 32, 64, 128, 256\}$ trees, so that we recover the original RF for $K = M = 256$ on both cases. For RE we use the training data as pruning set. Experiments with a dedicated pruning set can be found in the appendix. Figure 1 shows the average accuracy over the size of the ensemble for a 5-fold cross-validation. The dashed lines depict the smaller RF and solid lines are the corresponding pruned ensemble. As expected, we find that ensemble pruning significantly improves the accuracy when smaller trees with $64 – 256$ leaf nodes are used. Moreover, the performance of the pruned forests approaches the performance of the original forests when more and more trees are added much like in the original experiments. However, the improvement in accuracy becomes negligible for trees with up to 1024 leaf nodes. Here, the accuracy of the pruned and the unpruned forest are near identical for any given number of trees. Maybe even worse, if we are only interested in the most accurate model then there is no reason to prune the ensemble as an unpruned Random Forest already seems to achieves the best performance.

We acknowledge that this experiment is one-sided because we only use Reduced Error Pruning – a nearly 25 year old method – for comparison. Maybe the problem simply lies in RE itself and not pruning in general? To verify this hypothesis we also repeated the above experiment with 6 additional pruning algorithms from the three different categories. In total we compare two ranking-based methods namely IE (Jiang et al., 2017) and IC (Lu et al., 2010); the three ordering-based methods RE (Margineantu & Dietterich, 1997), DREP (Li et al., 2012) and COMP (Martínez-Munoz & Suárez, 2004) and the two clustering-based prun-

| $n_l$ | K  | COMP | DREP | IC  | IE  | LMD | RE  | RF  |
|------|----|------|------|-----|-----|-----|-----|-----|
| 64   | 8  | 81.86| 80.79| 81.71| 81.34| 81.46| **82.22**| 80.92|
| 32   | 8  | 83.09| 82.33| 83.10| 82.22| 82.60| **83.23**| 81.98|
| 128  | 8  | 82.17| 83.28| 83.10| 82.49| 82.87| **83.19**| 82.34|
| 128  | 8  | 85.12| 84.08| 84.69| 84.81| 83.85| **85.53**| 84.23|
| 32   | 8  | 86.34| 85.49| 86.38| 85.76| 85.65| **86.62**| 85.17|
| 128  | 8  | 86.40| 85.75| 86.27| 86.00| 86.03| **86.54**| 85.92|
| 256  | 8  | 87.37| 86.24| 87.14| 87.16| 86.36| **87.46**| 86.44|
| 32   | 8  | 88.97| 88.02| **89.07**| 88.70| 88.37| 89.01| 88.16|
| 128  | 8  | 88.97| 88.70| **89.15**| 88.97| 88.77| 89.07| 88.71|
| 512  | 8  | 88.36| 88.51| **88.96**| 88.78| 87.44| 88.79| 87.95|
| 32   | 8  | 91.11| 90.30| **91.34**| 90.67| 90.37| 90.68| 90.17|
| 128  | 8  | 91.22| 90.91| **91.41**| 91.30| 90.87| 91.23| 90.83|
| 1024 | 8  | 89.45| 89.26| **89.51**| 89.70| 88.30| 88.85| 88.83|
| 32   | 8  | 92.25| 92.30| **92.64**| 92.60| 91.82| 92.21| 91.91|
| 128  | 8  | 92.85| 92.85| **93.17**| 92.98| 92.70| 92.84| 92.70|

Table 2. 5-fold cross-validation accuracy over the number of members $K \in \{8, 32, 128\}$ in the ensemble for different $n_l$ parameters and different methods on the EEG dataset. Rounded to the second decimal digit. Larger is better. The best method is depicted in bold.

We also experimented with MQIP pruning methods (Zhang et al., 2006; Cavalcanti et al., 2016), but unfortunately the MQIP solver (in our case Gurobi\(^1\)) used during experiments would frequently fail or time-out. Thus we decided to not include any MQIP pruning methods in our evaluation.

Table 2 shows the result of this experiment. For space reasons we only depict results for $K \in \{8, 32, 128\}$. As expected, all pruning methods manage to improve the performance of the original RF for smaller $n_l \leq 256$ and keep this advantage to some degree for larger $n_l$. However, this advantage becomes smaller and smaller for larger $n_l$ until it is virtually non-existent for $n_l = 1024$ and the accuracies are near identical. Again, as expected setting $n_l$ to larger

---

\(^1\)https://www.gurobi.com/
values leads to the overall best accuracy.

For presentational purposes we highlighted this experiment on the EEG dataset, but we found that this behavior seems to hold universally across the other 15 datasets we experimented with. The detailed results for these experiments are given in the appendix. While the specific curves would differ we always found that the performance of a well-trained forest and its pruned counterpart would nearly match once the individual trees become large enough. For more plots with experiments on other datasets and other ‘base’ ensembles please consult the appendix.

4 IMPROVING THE ACCURACY-MEMORY TRADE-OFF OF RF

Clearly, the previous section shows that we cannot expect the accuracy of a pruned forest to improve much upon the performance of a well-trained Random Forest. On the one hand, this is a clear argument in favor of Random Forests – why should we prune a pre-trained forest if we can directly train a similar forest in the first place? On the other hand, pruning shows clear superior performance for smaller $n_t$ compared to RF. While pruning and RF both converge against a very similar maximum accuracy, pruning shows a better trade-off between the model size (controlled by $n_t$ and $K$) and the accuracy.

We argue that, from a hardware perspective, this trade-off is what really matters and a good algorithm should produce accurate models for a variety of different model sizes. Ensemble pruning improves this trade-off by removing unnecessary trees from the ensemble thereby reducing the memory consumption while keeping (or improving) its predictive power. But, the removal of entire trees does not offer a very fine-grained control over this trade-off. For example, it could be better to train a large forest with many, but comparably small trees instead of having one small forest of large trees. Hence, we propose to directly evaluate the accuracy-memory trade-off and to optimize towards it.

To do so, we present a simply and surprisingly effective method which refines the predictions of a given forest with Stochastic Gradient Descent (SGD). Our method trains a small initial Random Forest (e.g. by using small values for $n_t$ and $M$) and then refines the predictions of the individual trees to improve the overall performance: Recall that DTs use a series of axis-aligned splits of the form $\mathbb{I}\{x_i \leq t\}$ and $\mathbb{I}\{x_i > t\}$ where $i$ is a pre-computed feature index and $t$ is a pre-computed threshold to determine the leaf nodes. Let $s_l(x): \mathcal{X} \rightarrow \{0, 1\}$ be the series of splits which is ‘1’ if $x$ belongs to leaf $l$ and ‘0’ if not, then the prediction of a tree is given by

$$h_i(x) = \sum_{l=1}^{L_i} \tilde{y}_{i,l} s_{i,l}(x) \quad (8)$$

where $\tilde{y}_{i,l} \in \mathbb{R}^C$ is the (constant) prediction value of leaf $l$ and $L_i$ is the total number of leaves in tree $h_i$. Let $\theta_i$ be the parameter vector of tree $h_i$ (e.g. containing split values, feature indices and leaf-predictions) and let $\theta = (\theta_1, \ldots, \theta_M)$ be the parameter vector of the entire ensemble $f_\theta$. Then our goal is to solve

$$\arg\min_{\theta} \frac{1}{N} \sum_{(x,y) \in S} \ell(f_\theta(x), y) \quad (9)$$

for a given loss $\ell$. We propose to minimize this objective via stochastic gradient-descent. SGD is an iterative algorithm which takes a small step into the negative direction of the gradient in each iteration $t$ by using an estimation of the true gradient

$$\theta^{t+1} \leftarrow \theta^t - \alpha^t g_B(\theta^t) \quad (10)$$

where

$$g_B(\theta^t) = \nabla_{\theta^t} \left( \sum_{(x,y) \in B} \ell(f_{\theta^t}(x), y) \right) \quad (11)$$

is the gradient of $\ell$ wrt. to $\theta^t$ computed on a mini-batch $B$. Unfortunately, the axis-aligned splits of a DT are not differentiable and thus it is difficult to refine them further with gradient-based approaches. However, the leaf predictions $\tilde{y}_{i,l}$ are simple constants that can easily be updated via SGD. Formally, we use $\theta_i = (\tilde{y}_{i,1}, \tilde{y}_{i,2}, \ldots)$ leading to

$$g_B(\theta_i^t) = \frac{1}{|B|} \left( \sum_{(x,y) \in B} \frac{\partial \ell(f_{\theta^t}(x), y)}{\partial f_{\theta^t}(x)} w_{i,s_{i,l}(x)} \right)_{l=1,2,\ldots,L_i} \quad (12)$$

Algorithm 2 summarizes this approach. First, in $get\_forest$ a forest with $K$ trees each containing at most $n_t$ leaf nodes is loaded. This forest can either be a pre-trained forest with $M$ trees from which we randomly sample $K$ trees or we may train an entirely new forest with $K$ trees directly. Once the forest has been obtained SGD is performed over the leaf-predictions of each tree using the step-size $\alpha^t \in \mathbb{R}_+^+$ to minimize the given loss $\ell$.

Leaf-Refinement is a flexible technique and can be used in combination with any tree ensemble such as Bagging (Breiman, 1996), ExtraTrees (Geurts et al., 2006), Random Subspaces (Ho, 1998) or Random Patches (Louppe & Geurts, 2012). Moreover, we can also refine the individual weights $w_i$ of the trees via SGD, although we did not find a meaningful improvement optimizing the weights and leafs simultaneously in our pre-experiments. For simplicity we will only focus on leaf-refinement in this paper without optimizing the individual weights and leave this for future research.
Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

Algorithm 2 RF with Leaf-Refinement (RF-LR).

1: \{Load forest and use constant weights\}
2: \( h \leftarrow \text{get\_forest}(K, n_l) \)
3: \( w \leftarrow (1/K, \ldots, 1/K) \)
4: \{Init. leaf predictions\}
5: \( \text{for } i = 1, \ldots, K \ do \)
6: \( \theta_i \leftarrow (\hat{y}_{i,1}, \hat{y}_{i,2}, \ldots) \)
7: \( \text{end for} \)
8: \{Perform SGD using Eq. 10 + Eq. 12\}
9: \( \text{for receive batch } B \ do \)
10: \( \text{for } i = 1, \ldots, K \ do \)
11: \( \theta_i^t \leftarrow \theta_i^t - \alpha_t g_B(\theta_i^t) \)
12: \( \text{end for} \)
13: \( \text{end for} \)

5 Experiments

In this section we experimentally evaluate our method and compare its accuracy-memory trade-off with regular RF and pruned RF. As argued before, our main concern is the final model size as it determines the resource consumption, runtime, and energy of the model application during deployment (Buschjäger & Morik, 2017; Buschjäger et al., 2018). The model size is computed as follows: A baseline implementation of DTs stores each node in an array and iterates over it (Buschjäger et al., 2018). Each node inside the array requires a pointer to the left / right child (8 bytes in total), a boolean flag if it is a leaf-node (1 byte), the feature index as well as the threshold to compare the feature against (8 bytes). Last, entries for the class probabilities are required for the leaf nodes (4 bytes per class). Thus, in total, a single node requires \( 17 + 4 \cdot C \) Bytes per node which we sum over all nodes in the entire ensemble.

We follow a similar experimental protocol as before: As earlier we train various Random Forests with \( M = 256 \) trees using \( n_l \in \{64, 128, 256, 512, 1024\} \). Again, we compare the aforementioned pruning methods COMP, DREP, IC, IE, LMD and RE with our leaf-refinement method (RF-LR) as well as a random selection of trees from the RF. Since our method shares some overlap with gradient boosted trees (GB, (Friedman, 2001)) we also include these in our evaluation. Each pruning method is tasked to select \( K \in \{8, 16, 32, 64, 128\} \) trees from the ‘base’ forest. For DREP, we additionally varied \( \rho \in \{0.25, 0.3, 0.35, 0.4, 0.45, 0.5\} \). For GB we use the deviance loss and train \( 8, 16, 32, 64, 128 \) trees with the different \( n_l \) values. For RF-LR we randomly sample \( K \in \{8, 16, 32, 64, 128\} \) trees from the given forest and perform 50 epochs\(^2\) of SGD with a constant step size \( \alpha = 0.1 \) and a batch size of 128. We experimented with the mean-squared error (MSE) and the cross-entropy loss for minimization, but could not find meaningful differences between both losses. Hence, for these experiments we focus on the MSE loss. In all experiments we perform a 5-fold cross validation except when the dataset comes with a given train/test split. We use the training set for both, training the initial forest and pruning it. For a fair comparison we made sure that each method receives the same forest in each cross-validation run. In all experiments, we use minimal pre-processing and encode categorical features as one-hot encoding. The base ensembles have been trained with Scikit-Learn (Pedregosa et al., 2011) and the code for our experiments and all pruning methods are included in this submission. We implemented all pruning algorithm in a Python package for other researchers called PyPruning which is available under https://github.com/sbuschjaeger/PyPruning. The code for the experiments in this paper are available under https://github.com/sbuschjaeger/leaf-refinement-experiments. In total we performed 8 960 experiments on 16 different datasets which are detailed in the appendix. Additionally, more experiments with different ‘base’ ensembles and a dedicated pruning set are shown in the appendix.

5.1 Qualitative Analysis

We are interested in the most accurate models with the smallest memory consumption. Clearly these two metrics can contradict each other. For a fair comparison we therefore use the best parameter configuration of each method across both dimensions. More specifically, we compute the Pareto front of each method which contains those parameter configurations which are not dominated across one or more dimensions. For space reasons we start with a qualitative analysis and focus the EEG and the chess dataset as they represent distinct behaviors we found during our experiments.

Figure 2 shows the results on the EEG dataset. As before, the accuracy ranges from 75% to 92.5% and the model size ranges from a few KB to roughly 12MB (note the logarithmic scale on the x-axis). As before, larger models seem to generally perform better and all models seem to converge against a similar solution, expect GB which is stuck around 85% accuracy. In the range of 0 – 4000 KB, however, there are larger differences. For example, for roughly 1000 KB, CA performs sub-optimal only reaching an accuracy around 87.5% whereas the other methods all seem to have a similar performance around 90% except RF-LR which has an accuracy around 91%. For smaller model sizes below 4000 KB RF-LR seems to be the clear winner offering roughly up to 1% more accuracy compared to the other methods. Moreover, it shows a better overall accuracy-memory trade-off.

Figure 3 shows the results on the chess dataset. Here the accuracy ranges from 28% to 75% with model sizes up

\(^2\)In one epoch we iterate once over the entire dataset.
Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

To 12 MB (again note the logarithmic scale on the x-axis). Similar to before, the pruning methods all converge against similar solutions just above 65%. CA still seems to perform poorly for smaller model sizes, but not as bad as on the EEG data. Similar, GB also seems to struggle on this dataset. It is worth noting, that some pruning methods (e.g. IC or RE) have a better accuracy-memory trade-off compared to Random Forest and they outperform the original forest by about 2%. RF-LR offers the best performance on this dataset and outperforms the original forest by about 8% accuracy across all model sizes. This effect is only present for RF-LR and cannot be seen for the other methods. Overall, RF-LR offers a much better accuracy-memory trade-off and offers the best overall accuracy.

Conclusion: First we find that the well-performing RF models often require more than 1 MB easily breaking the available memory on small MCUs (cf. Table 1). Second, we found two different behaviors of RF-LR: In many cases all methods converge against a similar accuracy when more memory is available, e.g. as seen in Figure 2. This can be expected since all methods derive their models from the same RF model. Here, RF-LR often has better models and hence offers a better accuracy-memory trade-off. In many other cases we found that RF-LR significantly outperforms the other methods and offers a much better accuracy across most model sizes, e.g. as depicted in Figure 3. In these cases, RF-LR offers a much better accuracy-memory trade-off and the best overall accuracy.

5.2 Quantitative Analysis

The previous section showed that RF-LR can offer substantial improvements on some datasets and smaller improvements in other cases. To give a more complete picture we will now look at the performance of each method under various memory constraints. Table 3 shows the best accuracy of each method (across all hyperparameter configurations) with a final model size below 64 KB. Such models could for example easily be deployed on an Arduino Due MCU (cf. Table 1). We find that RF-LR offers the best accuracy in 9 out of 16 cases followed by RE which is first in 4 cases followed by GB with ranks first in 3 cases. IE shares the first place with RE on the mozilla dataset. On some datasets such as ida2016 the differences are comparably small which can be expected since all models are derived from the same base Random Forest. However, on other datasets such as the eeg, chess, japanese-vowels or connect dataset we can find more substantial improvements where RF-LR offers up to 5% better accuracy to the second ranking method.

A similar picture can be find in Table 4 which shows the best accuracy of each method (across all hyperparameter configurations) with a final model size below 256 KB. Such models could for example easily be deployed on an STM32F4 MCU (cf. Table 1). Now RF-LR offers the best accuracy in 10 out of 16 cases followed by RE which is first in only 1 case followed by GB with ranks first in 2 cases and IC which is now first in 3 cases. IE now shares the first place with IC on the mozilla dataset. As before, the differences are comparably small on some datasets (e.g. the mozilla dataset) and more substantial on other datasets where RF-LR now offers up 6% better accuracy against the second best method.

To give a more complete picture across different memory constraints we will now summarize the performance of each method by the (normalized) area-under the Pareto front: In- intuitively, we want to have an algorithm which gives small and accurate models and therefore places itself in the upper-left corner of the accuracy-memory plots. Similar to ‘regular’ ROC-AUC curves we can compute the area under the Pareto front (APF) normalized by the biggest model to summarize the accuracy for different models on the same dataset. Table 5 depicts the normalized APF for the experiments. Looking at RF-LR, we see that it is the clear winner. In total, it is the best method on 11 of 14 datasets, shares the

Figure 2. 5-fold cross-validation accuracy over the size of the ensemble on the EEG dataset. Single points are the individual parameter configurations whereas the solid line depicts the corresponding Pareto front. Best viewed in color.

Figure 3. 5-fold cross-validation accuracy over the size of the ensemble on the chess dataset. Single points are the individual parameter configurations whereas the solid line depicts the corresponding Pareto front. Best viewed in color.
### Table 3.

Test accuracies for models with a memory consumption below 64 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold. More details on the experiments and datasets can be found in the appendix.

| model       | CA   | COMP | DREP | GB   | IC   | IE   | LMD | RE   | RF   | RF-LR |
|-------------|------|------|------|------|------|------|-----|------|------|-------|
| adult       | 85.455 | 85.777 | 85.618 | **86.616** | 85.799 | 85.882 | 85.378 | 86.128 | 85.464 | 86.241 |
| anura       | 96.511 | 97.137 | 96.873 | 95.024 | 97.192 | 96.928 | 96.539 | 97.067 | 96.525 | **97.512** |
| avila       | 91.930 | 96.957 | 95.318 | 67.724 | 96.760 | 96.689 | 86.965 | **97.048** | 92.534 | 94.278 |
| bank        | 89.772 | 90.049 | 89.927 | **90.677** | 90.213 | 90.330 | 89.819 | 90.336 | 89.830 | 90.522 |
| chess       | 48.025 | 51.714 | 50.492 | 34.969 | 50.356 | 51.793 | 49.041 | 52.495 | 48.090 | **57.382** |
| connect     | 72.426 | 73.898 | 73.409 | 70.847 | 73.724 | 73.832 | 73.110 | 74.134 | 72.666 | **77.498** |
| eeg         | 84.419 | 85.574 | 84.419 | 82.463 | 84.780 | 85.033 | 83.852 | 85.527 | 84.606 | **86.622** |
| elec        | 83.523 | 83.620 | 84.075 | 83.878 | 82.749 | 84.556 | 82.817 | 84.680 | 83.391 | **84.894** |
| ida2016    | 90.061 | 91.487 | 91.045 | 87.341 | 91.517 | 90.332 | 90.794 | 91.587 | 90.503 | **93.173** |
| japanese-vowels | 86.109 | 86.613 | 86.282 | **87.355** | 86.692 | 86.771 | 86.456 | 86.845 | 86.419 | 86.997 |
| magic       | 80.700 | 85.700 | 84.300 | 74.400 | 84.700 | 84.100 | 85.000 | 84.900 | 84.200 | **87.000** |
| mnist       | 95.508 | 95.804 | 95.575 | 95.958 | 95.749 | 95.819 | 95.358 | 95.802 | 95.633 | **96.063** |
| nokia       | 79.913 | 81.688 | 80.827 | 69.058 | 81.137 | 80.996 | 79.599 | 81.727 | 80.246 | 81.081 |
| postures    | 88.647 | 88.880 | 88.663 | 87.449 | 88.616 | 88.647 | 89.020 | 88.538 | 88.834 | **95.016** |

### Table 4.

Test accuracies for models with a memory consumption below 256 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold. More details on the experiments and datasets can be found in the appendix.

| model       | CA   | COMP | DREP | GB   | IC   | IE   | LMD | RE   | RF   | RF-LR |
|-------------|------|------|------|------|------|------|-----|------|------|-------|
| adult       | 85.774 | 86.174 | 85.956 | **87.135** | 86.011 | 86.232 | 85.738 | 86.272 | 85.799 | 86.508 |
| anura       | 96.511 | 97.818 | 97.748 | 97.429 | **98.040** | 97.929 | 97.582 | 97.818 | 97.735 | **98.013** |
| avila       | 95.016 | 99.022 | 97.427 | 82.820 | 99.176 | 99.123 | 93.315 | **99.248** | 96.526 | 98.807 |
| bank        | 89.967 | 90.297 | 90.082 | 90.737 | 90.290 | 90.443 | 89.952 | 90.471 | 90.038 | **90.874** |
| chess       | 56.911 | 59.998 | 57.314 | 44.678 | 59.791 | 59.428 | 57.496 | 60.636 | 56.238 | **66.852** |
| connect     | 74.522 | 76.270 | 75.119 | 76.170 | 76.239 | 76.333 | 75.100 | 76.303 | 74.997 | **80.340** |
| eeg         | 87.233 | 88.364 | 88.511 | 85.734 | 88.959 | 88.778 | 87.884 | 87.951 | 87.951 | **90.454** |
| elec        | 85.220 | 85.810 | 85.832 | 85.748 | 86.043 | 86.692 | 84.922 | 86.562 | 85.362 | **87.829** |
| ida2016    | 99.106 | 99.225 | 99.119 | 99.169 | **99.262** | 99.175 | 99.238 | 99.175 | 99.194 | 99.244 |
| japanese-vowels | 91.979 | 94.810 | 94.428 | 94.860 | 94.790 | 94.077 | 94.278 | 94.709 | 94.408 | **96.205** |
| magic       | 86.655 | 87.281 | 87.218 | **87.733** | 87.234 | 87.239 | 87.302 | 87.365 | 86.950 | 87.570 |
| mnist       | 86.900 | 90.100 | 89.300 | 84.600 | 90.600 | 89.700 | 88.800 | 89.600 | 89.300 | **91.800** |
| mozilla     | 94.731 | 95.034 | 94.982 | 94.989 | **95.092** | **95.092** | 94.802 | 94.995 | 94.912 | 95.014 |
| nokia       | 96.039 | 96.222 | 96.150 | 96.408 | 96.318 | 96.269 | 96.077 | 96.356 | 96.135 | **96.539** |
| postures    | 88.587 | 89.596 | 88.747 | 81.100 | 89.231 | 89.390 | 88.085 | 89.633 | 88.281 | **90.504** |
| satimage    | 88.802 | 90.218 | 89.891 | 89.782 | 89.705 | 89.891 | 90.000 | 90.156 | 90.016 | **90.715** |
first place on 1 dataset (mozilla), is the second best method on 1 data-set (satimage), third best method (ida2016) and fourth best method (avila) each on one dataset. In the first block of datasets (chess, connect, eeg, elec, postures) RF-LR achieves substantial improvements with 1%−8% higher accuracies on average. Looking at the second block (adult, anura, bank, magic, mnist, nomao, japanese-vowels) RF-LR is still the best method, but the differences are smaller than before. Finally, in block three (ida2016, mozilla, satimage) RF-LR is not the best method alone anymore, but ranks among the best methods.

Table 5 implies that RF-LR can offer substantial improvements in some cases and moderate improvement in many other cases. To give a statistical meaningful comparison we present the results in Table 5 as a CD diagram (Demšar, 2006). A CD diagram ranks each method according to its performance on each dataset. Then, a Friedman-Test is performed to determine if there is a statistical difference between the average rank of each method. If this is the case, a pairwise Wilcoxon-Test between all methods is used to check whether there is a statistical difference between two classifiers. CD diagrams visualize this evaluation by plotting the average rank of each method on the x-axis and connect all classifiers whose performances are statistically similar via a horizontal bar.

Figure 4 shows the CD diagram for the experiments, where \( p = 0.95 \) was used for all statistical tests. RF-LR is the clear winner in this comparison. Its average rank is close to 1.5 and it has some distance to the second best method IC with an average rank around 2.5. It offers a statistically better performance compared to \{RE, IE, COMP, LMD, DREP, RF, GB, CA\}. The second clique is given by \{IC, RE, IE, COMP, LMD, DREP, RF, GB\} with ranks around 2.5−7. Overall, RE places third with an average rank around 4 which shows that a simple method can perform surprisingly well. We hypothesize that since RE minimizes the overall ensemble loss that it finds a good balance between the bias and the diversity of the ensemble as e.g. discussed in (Buschjäger et al., 2020).

Next, \{RF, GB, CA\} form the last clique with statistically similar performances which shows that a unpruned RF and GB do not offer a good accuracy-memory trade-off. CA ranks last with some distance to the other methods. We are not sure why CA has such a bad performance and suspect a bug in our implementation which we could not find so far.
5.3 Case-Study On Raspberry Pi0

To showcase the effectiveness of our approach we will now compare the performance of ensemble pruning and leaf-refinement on a Raspberry Pi0. The Raspberry Pi0 has 512 MB RAM and uses a BCM 2835 SoC CPU clocked at 1 GHz. This makes it considerably more powerful than the MCUs mentioned in Table 1, but also allows us to run a full Linux environment which simplifies the evaluation. Again we will now focus on the EEG dataset as our standard example. From our previous experiment we selected pruning configurations that resulted in an ensemble size below 256 KB and generate ensemble-specific C++ code as outline in (Buschjäger et al., 2018) which is then compiled on the Pi0 itself. We compare the latency, the accuracy and the total binary size of these implementations. Note that the total binary size may exceed 256 KB because the binary also contains additional functions from the standard library as well as a start routine and the corresponding ELF header. However, this overhead is the same for all implementations. For simplicity we measure the accuracy as well as the latency using the first cross-validation set. To ensure a fair comparison we repeat each experiment 5 times. Table 6 contains the results for this evaluation. As one can see the binary sizes range from 268 KB to roughly 800 KB and each implementation requires 0.8 to 1.6 µs to classify a single observation. As expected, RF-LR offers the best accuracy around 91% while ranking third in memory usage. Somewhat surprisingly, RF-LR has a comparably high latency. We conjecture that the structure of the trees in RF-LR is not very homogeneous which seems to be beneficial for the accuracy, but may hurt the caching behavior of the trees. A more thorough discussion can be found in (Buschjäger et al., 2018) on this topic and a combination of both approaches should be considered in future research. Nevertheless, this evaluation shows that our approach can be applied in a real-world scenario and we believe that these results can be transferred to other hardware architectures as well.

6 Conclusion

Ensemble algorithms are among the state-of-the-art in many machine learning applications. With the ongoing integration of ML models into everyday life, the deployment and continuous application of models becomes more and more an important issue. By today’s standard, Random Forests are trained with large trees for the best performance which can challenge the resources of small devices and sometimes make deployment impossible. Ensemble pruning is a standard technique to remove unnecessary classifiers from the ensemble to reduce the overall resource consumption while potentially improving its accuracy. This makes ensemble pruning ideal to bring accurate ensembles to small devices. While ensemble pruning improves the performance of ensembles of small trees we found that this improvement diminishes for ensembles of large trees. Moreover, it does not offer fine-grained control over this trade-off because it removes entire trees at once from the ensemble. We argue that, from a hardware perspective, the fine-grained control over the accuracy-memory trade-off is what really matters. We propose a simple and surprisingly effective algorithm which refines the predictions of the trees in a forest using SGD. We compared our Leaf-Refinement method against 7 state-of-the-art pruning methods on 16 datasets. Leaf-Refinement outperforms the other methods on 11 of 16 datasets with a statistically significant better accuracy-memory trade-off compared to most methods. In a small study we showed that our approach can be applied in real-world scenarios, and we believe that our results can be transferred to other hardware architectures. Since our approach is orthogonal to existing approaches it can be freely combined with other methods for efficient deployment. Hence future research should include not only the combination of more diverse hardware, but also the combination of different methods.

Table 6. Accuracy, Size and Latency on a Raspberry Pi0. Models are filtered so that the ensemble size does not exceed 256 KB.

| Method | Accuracy | Size [Bytes] | Latency [µs/obs.] |
|--------|----------|--------------|-------------------|
| CA | 86.2817 | 268 536 | 0.80107 |
| COMP | 89.8531 | 689 712 | 1.06809 |
| DREP | 88.7850 | 342 696 | 1.00134 |
| IC | 89.2523 | 793 280 | 1.06809 |
| IE | 88.8518 | 743 232 | 1.00134 |
| LMD | 88.8518 | 784 896 | 1.06809 |
| RE | 89.2523 | 792 456 | 1.13485 |
| RF | 89.5194 | 588 336 | 1.60214 |
| RF-LR | 91.0881 | 588 088 | 1.46862 |

References

Barros, R. C., de Carvalho, A. C. P. L. F., and Freitas, A. A. Decision-Tree Induction, pp. 7–45. Springer International Publishing, Cham, 2015. ISBN 978-3-319-14231-9. doi: 10.1007/978-3-319-14231-9_2. URL https://doi.org/10.1007/978-3-319-14231-9_2.

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

Biau, G. and Scornet, E. A random forest guided tour. Test, 25(2):197–227, 2016.

Branco, S., Ferreira, A. G., and Cabral, J. Machine learning
Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement in resource-scarce embedded systems, fpgas, and end-devices: A survey. *Electronics*, 8(11):1289, 2019.

Breiman, L. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

Breiman, L. Some infinity theory for predictor ensembles. Technical report, Technical Report 579, Statistics Dept. UCB, 2000.

Buschjäger, S. and Morik, K. Decision tree and random forest implementations for fast filtering of sensor data. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 65(1):209–222, 2017.

Buschjäger, S., Pfahler, L., and Morik, K. Generalized negative correlation learning for deep ensembling. *arXiv preprint arXiv:2011.02952*, 2020.

Buschjäger, S., Chen, K., Chen, J., and Morik, K. Realization of random forest for real-time evaluation through tree framing. In *ICDM*, pp. 19–28, 2018. doi: 10.1109/ICDM.2018.00017.

Cavalcanti, G. D., Oliveira, L. S., Moura, T. J., and Carvalho, G. V. Combining diversity measures for ensemble pruning. *Pattern Recognition Letters*, 74:38–45, 2016.

Choudhary, T., Mishra, V., Goswami, A., and Sarangapani, J. A comprehensive survey on model compression and acceleration. *Artificial Intelligence Review*, 53(7):5113–5155, 2020.

Cortes, C., Mohri, M., and Syed, U. Deep boosting. In *Proceedings of the Thirty-First International Conference on Machine Learning (ICML 2014)*, 2014.

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

Denil, M., Matheson, D., and De Freitas, N. Narrowing the gap: Random forests in theory and in practice. In *International conference on machine learning (ICML)*, 2014.

Domingos, P. A unified bias-variance decomposition for zero-one and squared loss. *AAAI/IAAI*, 2000:564–569, 2000.

Friedman, J. H. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pp. 1189–1232, 2001.

Geurts, P., Ernst, D., and Wehenkel, L. Extremely randomized trees. *Machine learning*, 63(1):3–42, 2006.

Giacinto, G., Rolli, F., and Fumera, G. Design of effective multiple classifier systems by clustering of classifiers. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, volume 2, pp. 160–163. IEEE, 2000.

Guo, H., Liu, H., Li, R., Wu, C., Guo, Y., and Xu, M. Margin & diversity based ordering ensemble pruning. *Neurocomputing*, 275:237–246, 2018.

Ho, T. K. The random subspace method for constructing decision forests. *IEEE transactions on pattern analysis and machine intelligence*, 20(8):832–844, 1998.

Jiang, Z., Liu, H., Fu, B., and Wu, Z. Generalized ambiguity decompositions for classification with applications in active learning and unsupervised ensemble pruning. *31st AAAI Conference on Artificial Intelligence, AAAI 2017*, pp. 2073–2079, 2017.

Koltchinskii, V. et al. Empirical margin distributions and bounding the generalization error of combined classifiers. *The Annals of Statistics*, 30(1):1–50, 2002.

Kumar, A., Goyal, S., and Varma, M. Resource-efficient machine learning in 2 kb ram for the internet of things. In *International Conference on Machine Learning*, pp. 1935–1944. PMLR, 2017.

Lazarevic, A. and Obradovic, Z. Effective pruning of neural network classifier ensembles. In *IJCNN'01*, volume 2, pp. 796–801. IEEE, 2001.

Li, N., Yu, Y., and Zhou, Z.-H. Diversity regularized ensemble pruning. In *ECML PKDD*, pp. 330–345. Springer, 2012.

Louppe, G. and Geurts, P. Ensembles on random patches. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 346–361. Springer, 2012.

Lu, Z., Wu, X., Zhu, X., and Bongard, J. Ensemble pruning via individual contribution ordering. In *Proc. of the ACM SIGKDD*, pp. 871–880, 2010.

Margineantu, D. D. and Dietterich, T. G. Pruning adaptive boosting. In *ICML*, volume 97, pp. 211–218, 1997.

Martinez-Munoz, G. and Suárez, A. Aggregation ordering in bagging. In *Proc. of the IASTED*, pp. 258–263, 2004.

Martínez-Muñoz, G. and Suárez, A. Pruning in ordered bagging ensembles. In *ICML*, pp. 609–616, 2006.

Martínez-Muñoz, G., Hernández-Lobato, D., and Suárez, A. An analysis of ensemble pruning techniques based on ordered aggregation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):245–259, 2008.
Oshiro, T. M., Perez, P. S., and Baranauskas, J. A. How many trees in a random forest? In *International workshop on machine learning and data mining in pattern recognition*, pp. 154–168. Springer, 2012.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Shotton, J., Sharp, T., Kohli, P., Nowozin, S., Winn, J., and Criminisi, A. Decision jungles: Compact and rich models for classification. In *NIPS’13 Proceedings of the 26th International Conference on Neural Information Processing Systems*, pp. 234–242, 2013.

Tsoumakas, G., Partalas, I., and Vlahavas, I. An ensemble pruning primer. In *Applications of supervised and unsupervised ensemble methods*. Springer, 2009.

Ye, T., Zhou, H., Zou, W. Y., Gao, B., and Zhang, R. Rapidscorer: fast tree ensemble evaluation by maximizing compactness in data level parallelization. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 941–950, 2018.

Zhang, Y., Burer, S., and Street, W. N. Ensemble pruning via semi-definite programming. *Journal of machine learning research*, 7(Jul):1315–1338, 2006.

Zhou, Z.-H. *Ensemble methods: foundations and algorithms*. CRC press, 2012.

Zhou, Z.-H., Wu, J., and Tang, W. Ensembling neural networks: many could be better than all. *Artificial intelligence*, 137(1-2):239–263, 2002.
APPENDIX: Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

Sebastian Buschjäger  
sebastian.buschjaeger@tu-dortmund.de
Katharina Morik  
katharina.morik@tu-dortmund.de

Abstract

This appendix accompanies the paper ‘Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement’. It provides results for more experiments which are not given in the paper due to space reasons.

1. Transformation of the Many-Could-Be-Better-Than-All-Theorem

Let

\[ C_{k,k} = \mathbb{E}_{x,y \sim \mathcal{D}} [(h_k(x) - y)^2] \]  
\[ C_{k,i} = \mathbb{E}_{x,y \sim \mathcal{D}} [(h_k(x) - y)(h_i(x) - y)] \]  

then from Eq. (14) and Eq. (15) in \cite{paper} we have:

\[
\sum_{i=1}^{M} \sum_{j=1, j \neq k}^{M} \frac{C_{i,j}}{(M-1)^2} \leq \sum_{i=1}^{M} \sum_{h=1}^{M} \frac{C_{i,j}}{M^2} = \sum_{i=1}^{M} \sum_{j=1, j \neq k}^{M} \frac{C_{i,j}}{M^2} + 2 \sum_{i=1}^{M} \frac{C_{i,k}}{M^2} + \frac{C_{k,k}}{M^2} \\
\leq \frac{(M - 1)^2}{M^2} \left( \sum_{i=1, i \neq k}^{M} \sum_{j=1, j \neq k}^{M} C_{i,j} + 2 \sum_{i=1, i \neq k}^{M} C_{i,k} + C_{k,k} \right) \\
\leq \sum_{i=1, i \neq k}^{M} \sum_{j=1, j \neq k}^{M} C_{i,j} + 2 \sum_{i=1, i \neq k}^{M} C_{i,k} + C_{k,k} \\
0 \leq 2 \sum_{i=1, i \neq k}^{M} C_{i,k} + C_{k,k} \\
-2 \sum_{i=1, i \neq k}^{M} C_{i,k} \leq C_{k,k}
\]

2. Dataset

Table 1 gives an overview of the datasets used for all experiments. All datasets are freely available online. The detailed download scripts for each dataset are provided in the anonymized version of the source code.

3. Revisiting Ensemble Pruning on More Datasets

The section ‘Revisiting Ensemble Pruning’ showed results for the EEG dataset. In this section, we show the results for this experiment on the other dataset depicted in Table 1.
Table 1: Summary of data sets for our experiments. All datasets are publicly available and
download scripts are included in this submission. In all experiments we use a 5-fold
cross validation except for mnist and ida2016 in which we used the given test/train
split. We use minimal pre-processing and by removing examples containing NaN
values and computing a one-hot encoding for categorical features. \( N \) depicts total
number of datapoints (after removing NaN), \( d \) is the dimensionality including all
one-hot encoded features and \( C \) is the number of classes.

| Dataset   | \( N \)   | \( d \) | \( C \) |
|-----------|-----------|--------|--------|
| adult     | 32,561    | 108    | 2      |
| anura     | 7,195     | 22     | 10     |
| avila     | 20,867    | 10     | 11     |
| bank      | 45,211    | 51     | 2      |
| chess     | 28,056    | 23     | 17     |
| connect   | 67,557    | 42     | 3      |
| eeg       | 14,980    | 14     | 2      |
| elec      | 45,312    | 14     | 2      |

| Dataset       | \( N \)   | \( d \) | \( C \) |
|---------------|-----------|--------|--------|
| ida2016       | 76,000    | 170    | 2      |
| japanese-vowels| 9,961    | 14     | 9      |
| magic         | 19,019    | 10     | 2      |
| mnist         | 70,000    | 784    | 10     |
| mozilla       | 15,545    | 5      | 2      |
| nomao         | 34,465    | 174    | 2      |
| postures      | 78,095    | 9      | 5      |
| satimage      | 6,430     | 36     | 6      |

Recall the following experimental protocol: Oshiro et al. showed in [7] that the prediction of
a RF stabilizes between 128 and 256 trees in the ensemble and adding more trees to the
ensemble does not yield significantly better results. Hence, we train the ‘base’ Random
Forests with \( M = 256 \) trees. To control the individual errors of trees we set the maximum
number of leaf nodes \( n_l \) to values between \( n_l \in \{64, 128, 256, 512, 1024\} \). For ensemble
pruning we use RE and compare it against a random selection of trees from the original
ensemble (which is the same a training a smaller forest directly). In both cases a sub-ensemble
with \( K \in \{2, 4, 8, 16, 32, 64, 128, 256\} \) members is selected so that for \( K = 256 \) the original
RF is recovered. For RE we use the training data as pruning set. We report the average
accuracy over a 5-fold cross-validation.
Figure 1: (Left) The error over the number of trees in the ensemble on the adult dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the adult dataset. Rounded to the second decimal digit. Larger is better.

Figure 2: (Left) The error over the number of trees in the ensemble on the anura dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the anura dataset. Rounded to the second decimal digit. Larger is better.
Figure 3: (Left) The error over the number of trees in the ensemble on the avila dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the avila dataset. Rounded to the second decimal digit. Larger is better.

Figure 4: (Left) The error over the number of trees in the ensemble on the bank dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the bank dataset. Rounded to the second decimal digit. Larger is better.
Figure 5: (Left) The error over the number of trees in the ensemble on the chess dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the chess dataset. Rounded to the second decimal digit. Larger is better.

Figure 6: (Left) The error over the number of trees in the ensemble on the connect dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the connect dataset. Rounded to the second decimal digit. Larger is better.
Figure 7: (Left) The error over the number of trees in the ensemble on the eeg dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the eeg dataset. Rounded to the second decimal digit. Larger is better.

Figure 8: (Left) The error over the number of trees in the ensemble on the elec dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the elec dataset. Rounded to the second decimal digit. Larger is better.
APPENDIX: Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

| max_leaf_nodes | model | COMP | DREP | IC | IE | LMD | RE |
|----------------|-------|------|------|---|---|-----|----|
| 64             | 8.0   | 80.04| 96.97| 99.09| 99.02| 98.95| 99.09|
| 16.0           | 99.12| 99.01| 99.08| 99.05| 99.02| 99.12|
| 32.0           | 99.05| 99.04| 99.09| 99.02| 99.02| 99.08|
| 64.0           | 99.05| 99.06| 99.07| 99.07| 99.04| 99.08|
| 128.0          | 99.06| 99.09| 99.10| 99.09| 99.10| 99.03| 99.10|
| 256.0          | 99.08| 99.04| 99.09| 99.10| 99.04| 99.04|
| 512.0          | 99.08| 99.04| 99.09| 99.10| 99.04| 99.04|
| 1024.0         | 99.08| 99.04| 99.09| 99.10| 99.04| 99.04|

Figure 9: (Left) The error over the number of trees in the ensemble on the ida2016 dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the ida2016 dataset. Rounded to the second decimal digit. Larger is better.

| max_leaf_nodes | model | COMP | DREP | IC | IE | LMD | RE |
|----------------|-------|------|------|---|---|-----|----|
| 64             | 8.0   | 88.78| 87.31| 88.51| 86.86| 87.28| 88.93|
| 16.0           | 90.24| 89.10| 90.62| 88.53| 89.80| 90.72|
| 32.0           | 91.00| 90.21| 91.16| 89.54| 90.96| 91.24|
| 64.0           | 91.54| 90.79| 91.37| 90.98| 91.63| 91.61|
| 128.0          | 91.60| 91.03| 91.15| 90.40| 91.11| 91.55|
| 256.0          | 91.74| 91.57| 92.06| 91.62| 92.22| 92.18|
| 512.0          | 92.02| 92.68| 93.37| 92.96| 93.64| 93.95|
| 1024.0         | 92.19| 93.44| 94.10| 93.98| 94.66| 94.95|

Figure 10: (Left) The error over the number of trees in the ensemble on the japanese-vowels dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the japanese-vowels dataset. Rounded to the second decimal digit. Larger is better.
Figure 11: (Left) The error over the number of trees in the ensemble on the magic dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the magic dataset. Rounded to the second decimal digit. Larger is better.

Figure 12: (Left) The error over the number of trees in the ensemble on the mnist dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the mnist dataset. Rounded to the second decimal digit. Larger is better.
Figure 13: (Left) The error over the number of trees in the ensemble on the mozilla dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the mozilla dataset. Rounded to the second decimal digit. Larger is better.

Figure 14: (Left) The error over the number of trees in the ensemble on the nomao dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the nomao dataset. Rounded to the second decimal digit. Larger is better.
Figure 15: (Left) The error over the number of trees in the ensemble on the postures dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the postures dataset. Rounded to the second decimal digit. Larger is better.

Figure 16: (Left) The error over the number of trees in the ensemble on the satimage dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the satimage dataset. Rounded to the second decimal digit. Larger is better.
4. Plotting the Pareto Front For More Datasets

Figure 17: 5-fold cross-validation accuracy over the size of the ensemble for different $n_l$ and different $M$ on the chess dataset. Single points are the individual parameter configurations whereas the solid line depicts the corresponding Pareto Front. Left side show the adult dataset, right side shows the anura dataset.

Figure 18: 5-fold cross-validation accuracy over the size of the ensemble for different $n_l$ and different $M$ on the chess dataset. Single points are the individual parameter configurations whereas the solid line depicts the corresponding Pareto Front. Left side show the avila dataset, right side shows the bank dataset.
Figure 19: 5-fold cross-validation accuracy over the size of the ensemble for different $n_l$ and different $M$ on the chess dataset. Single points are the individual parameter configurations whereas the solid line depicts the corresponding Pareto Front. Left side show the chess dataset, right side shows the connect dataset.

Figure 20: 5-fold cross-validation accuracy over the size of the ensemble for different $n_l$ and different $M$ on the chess dataset. Single points are the individual parameter configurations whereas the solid line depicts the corresponding Pareto Front. Left side show the eeg dataset, right side shows the elec dataset.
Figure 21: 5-fold cross-validation accuracy over the size of the ensemble for different $n_l$ and different $M$ on the chess dataset. Single points are the individual parameter configurations whereas the solid line depicts the corresponding Pareto Front. Left side show the ida2016 dataset, right side shows the japanese-vowels dataset.

Figure 22: 5-fold cross-validation accuracy over the size of the ensemble for different $n_l$ and different $M$ on the chess dataset. Single points are the individual parameter configurations whereas the solid line depicts the corresponding Pareto Front. Left side show the magic dataset, right side shows the mnist dataset.
Figure 23: 5-fold cross-validation accuracy over the size of the ensemble for different $n_l$ and different $M$ on the chess dataset. Single points are the individual parameter configurations whereas the solid line depicts the corresponding Pareto Front. Left side show the mozilla dataset, right side shows the nomao dataset.

Figure 24: 5-fold cross-validation accuracy over the size of the ensemble for different $n_l$ and different $M$ on the chess dataset. Single points are the individual parameter configurations whereas the solid line depicts the corresponding Pareto Front. Left side show the postures dataset, right side shows the satimage dataset.

4.1 Accuracies under various resource constraints

5. Revisiting Ensemble Pruning on More Datasets with a dedicated pruning set

Some authors use a dedicated pruning set (see e.g.) for ensemble pruning which was not used for training the ensemble. For completeness, we adapt this approach into the experimental protocol. We now split the training data into two sets with 2/3 and 1/3 of the original training data. The 2/3 of the training data is used to train the base ensemble, and the 1/3 of the data is used for pruning. As before, we either use a 5-fold cross validation or the given test/train split. For reference, recall our experimental protocol: Oshiro et al. showed in? that the prediction of a RF stabilizes between 128 and 256 trees in the ensemble and adding more trees to the ensemble does not yield significantly better results. Hence, we train the ‘base’ Random Forests with $M = 256$ trees. To control the individual errors of trees we set the maximum number of leaf nodes $n_l$ to values between $n_l \in \{64, 128, 256, 512, 1024\}$. 

7.ERM
Table 2: Test accuracies for models with a memory consumption below 32 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model         | CA    | COMP | DREP | GB   | IC   | IE   | LMD  | RE   | RF   | RF-LR |
|---------------|-------|------|------|------|------|------|------|------|------|-------|
| adult         | 85.120| 85.467| 85.403| **86.244**| 85.295| 85.824| 85.056| 86.035| 85.234| 85.891 |
| anura         | 95.789| 96.484| 96.220| 91.564| 96.609| 96.372| 96.081| 96.609| 96.178| **96.831**|
| avila         | 86.102| 93.382| 91.115| 59.496| 93.435| 93.219| 82.446| 94.197| 86.285| 88.355 |
| bank          | 89.741| 89.985| 89.927| **90.527**| 90.213| 90.226| 89.706| 90.246| 89.764| 90.343 |
| chess         | 44.914| 47.986| 46.696| 30.175| 46.753| 48.371| 45.391| 48.571| 44.614| 51.205 |
| connect       | 71.551| 73.052| 72.595| 67.584| 73.057| 73.223| 71.993| 73.313| 71.466| **75.160**|
| eeg           | 81.015| 83.037| 82.917| 79.226| 83.451| 81.455| 83.565| 82.443| 84.059| **84.059**|
| elec          | 82.252| 82.122| 83.009| 81.941| 83.737| 81.649| 83.942| 82.027| 83.464| **83.464**|
| ida2016       | 98.956| 99.038| 98.969| 99.038| 99.087| 99.025| 98.950| 99.094| 99.031| 99.031 |
| japanese-vowels| 86.819| 88.796| 87.843| 81.930| 88.716| 88.626| 87.280| 88.927| 87.632| **90.433**|
| magic         | 86.109| 86.366| 86.140| 86.629| 86.072| 86.487| 86.114| 86.640| 86.019| 86.187 |
| mnist         | 92.534| 94.278|
| moza          | 94.442| 94.628| 94.661| 94.493| **94.815**| 94.693| 94.275| 94.783| 94.532| 94.538 |
| nomao         | 95.242| 95.372| 95.239| 95.273| 95.381| 95.567| 95.065| **95.683**| 95.358| 95.668 |
| postures      | 74.960| 76.138| 75.261| 64.126| 75.477| 75.978| 74.101| **76.873**| 74.673| 75.069 |
| satimage      | 87.807| **88.491**| 87.776| 85.925| 88.289| 87.589| 87.403| 88.258| 87.760| 87.216 |

Table 3: Test accuracies for models with a memory consumption below 64 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model         | CA    | COMP | DREP | GB   | IC   | IE   | LMD  | RE   | RF   | RF-LR |
|---------------|-------|------|------|------|------|------|------|------|------|-------|
| adult         | 85.455| 85.777| 85.618| **86.616**| 85.799| 85.882| 85.378| 86.128| 85.464| 86.241 |
| anura         | 96.511| 97.137| 96.873| 95.024| 97.192| 96.928| 96.539| 97.067| 96.525| **97.512**|
| avila         | 91.930| 96.957| 95.318| 67.724| 96.760| 96.689| 96.965| 97.048| 92.534| 94.278 |
| bank          | 89.772| 90.049| 89.927| **90.677**| 90.213| 90.330| 89.819| 90.336| 89.830| 90.522 |
| chess         | 48.025| 51.714| 50.492| 34.969| 50.356| 51.793| 49.041| 52.495| 48.090| **57.382**|
| connect       | 72.426| 73.898| 73.409| 70.847| 73.724| 73.832| 73.110| 74.134| 72.666| **77.498**|
| eeg           | 84.199| 85.574| 84.419| 82.463| 84.780| 85.033| 83.852| 85.527| 84.606| **86.622**|
| elec          | 83.523| 83.620| 84.075| 83.878| 82.749| 84.556| 82.817| 84.680| 83.391| **84.894**|
| ida2016       | 99.044| 99.119| 99.025| 99.094| 99.106| 99.100| 99.031| 99.125| 99.075| **99.219**|
| japanese-vowels| 90.061| 91.487| 91.045| 87.341| 91.517| 90.332| 90.794| 91.587| 90.503| **93.173**|
| magic         | 86.109| 86.613| 86.282| **87.355**| 86.692| 86.771| 86.456| 86.845| 86.419| 86.997 |
| mnist         | 80.700| 85.700| 84.300| 74.400| 84.700| 84.100| 85.000| 84.900| 84.200| **87.000**|
| moza          | 94.590| 94.764| 94.661| 94.590| 94.815| **94.860**| 94.468| 94.860| 94.545| 94.699 |
| nomao         | 95.508| 95.804| 95.575| 95.958| 95.749| 95.819| 95.358| 95.802| 95.633| **96.063**|
| postures      | 79.913| 81.688| 80.827| 69.058| 81.137| 80.996| 79.599| **81.727**| 80.246| 81.081 |
| satimage      | 88.647| 88.880| 88.663| 87.449| 88.911| 88.616| 88.647| **89.020**| 88.538| 88.834 |

APPENDIX: Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement
Table 4: Test accuracies for models with a memory consumption below 128 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model          | CA   | COMP | DREP | GB  | IC  | LMD | RE  | RF  | RF-LR |
|----------------|------|------|------|-----|-----|-----|-----|-----|-------|
| adult          | 85.685 | 85.925 | 85.784 | 87.135 | 85.980 | 86.164 | 85.738 | 86.155 | 85.787 | 86.508 |
| anura          | 96.511 | 97.568 | 97.206 | 96.553 | 97.679 | 97.512 | 97.137 | 97.415 | 97.206 | 97.693 |
| avila          | 94.738 | 98.720 | 95.697 | 75.262 | 98.845 | 98.730 | 91.015 | 98.860 | 93.411 | 97.791 |
| bank           | 89.967 | 90.148 | 89.985 | 90.737 | 90.259 | 90.376 | 89.830 | 90.354 | 89.994 | 90.794 |
| chess          | 53.315 | 56.027 | 54.370 | 40.719 | 55.318 | 55.703 | 53.201 | 56.672 | 53.261 | 62.749 |
| connect        | 73.348 | 74.897 | 74.065 | 73.850 | 75.026 | 74.327 | 75.305 | 73.825 | 97.791 |
| eeg            | 85.534 | 87.370 | 86.242 | 85.734 | 87.136 | 87.163 | 86.355 | 87.457 | 86.442 | 88.992 |
| elec           | 84.589 | 84.318 | 84.872 | 85.748 | 84.635 | 83.896 | 85.851 | 84.574 | 86.416 | 86.416 |
| ida2016        | 99.044 | 99.119 | 99.106 | 99.169 | 99.125 | 99.131 | 99.156 | 99.087 | 99.244 | 94.699 |
| japanese-vowels| 91.979 | 93.575 | 92.912 | 91.748 | 93.444 | 92.651 | 93.143 | 92.410 | 94.409 | 94.699 |
| magic          | 86.608 | 86.882 | 86.661 | 87.733 | 87.071 | 86.987 | 87.087 | 86.918 | 87.570 | 90.200 |
| mnist          | 86.900 | 89.000 | 87.600 | 79.300 | 88.500 | 87.400 | 87.200 | 86.800 | 86.900 | 96.298 |
| mozilla        | 94.622 | 95.000 | 94.744 | 94.989 | 94.995 | 94.957 | 94.878 | 94.834 | 94.957 | 94.957 |
| nomao          | 95.738 | 95.927 | 95.903 | 96.408 | 96.073 | 96.060 | 95.920 | 95.871 | 96.298 | 94.619 |
| postures       | 84.582 | 86.166 | 85.655 | 75.760 | 85.933 | 85.169 | 84.167 | 86.267 | 84.400 | 86.300 |
| satimage       | 88.647 | 89.378 | 89.300 | 88.631 | 89.425 | 89.393 | 89.518 | 89.580 | 90.140 |

Table 5: Test accuracies for models with a memory consumption below 256 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model          | CA   | COMP | DREP | GB  | IC  | LMD | RE  | RF  | RF-LR |
|----------------|------|------|------|-----|-----|-----|-----|-----|-------|
| adult          | 85.774 | 86.174 | 85.956 | 87.135 | 86.011 | 86.232 | 85.738 | 86.272 | 85.799 | 86.508 |
| anura          | 96.511 | 97.818 | 97.748 | 97.429 | 98.040 | 97.929 | 97.582 | 97.818 | 97.735 | 98.013 |
| avila          | 95.016 | 99.022 | 97.427 | 82.820 | 99.176 | 99.123 | 93.315 | 99.248 | 96.526 | 98.807 |
| bank           | 89.967 | 90.297 | 90.082 | 90.737 | 90.290 | 90.443 | 89.952 | 90.471 | 90.038 | 90.874 |
| chess          | 56.911 | 59.998 | 57.314 | 44.678 | 59.791 | 59.428 | 57.496 | 60.636 | 56.238 | 66.852 |
| connect        | 74.522 | 76.270 | 75.119 | 76.170 | 76.239 | 76.333 | 75.100 | 76.303 | 74.997 | 80.340 |
| eeg            | 87.223 | 88.364 | 88.511 | 85.734 | 88.509 | 88.778 | 88.784 | 88.758 | 87.951 | 90.454 |
| elec           | 85.220 | 85.810 | 85.832 | 85.748 | 86.043 | 86.692 | 84.922 | 86.926 | 85.362 | 87.829 |
| ida2016        | 99.106 | 99.225 | 99.119 | 99.169 | 99.262 | 99.175 | 99.238 | 99.175 | 99.994 | 99.244 |
| japanese-vowels| 91.979 | 94.810 | 94.428 | 94.860 | 94.790 | 94.077 | 94.278 | 94.709 | 94.408 | 96.205 |
| magic          | 86.655 | 87.281 | 87.218 | 87.733 | 87.234 | 87.339 | 87.302 | 87.365 | 86.950 | 87.570 |
| mnist          | 86.900 | 90.100 | 89.300 | 84.600 | 90.600 | 89.700 | 88.800 | 89.600 | 89.300 | 91.800 |
| mozilla        | 94.731 | 95.034 | 94.982 | 94.989 | 95.092 | 95.092 | 94.802 | 94.905 | 94.912 | 95.014 |
| nomao          | 96.393 | 96.222 | 96.150 | 96.408 | 96.318 | 96.269 | 96.077 | 96.356 | 96.135 | 96.539 |
| postures       | 88.587 | 89.596 | 88.747 | 81.100 | 89.231 | 89.390 | 88.085 | 89.633 | 88.281 | 90.504 |
| satimage       | 88.802 | 90.218 | 89.891 | 89.782 | 89.705 | 89.891 | 90.000 | 90.156 | 90.016 | 90.715 |
For ensemble pruning we use RE and compare it against a random selection of trees from the original ensemble (which is the same as training a smaller forest directly). In both cases a sub-ensemble with $K \in \{2, 4, 8, 16, 32, 64, 128, 256\}$ members is selected so that for $K = 256$ the original RF is recovered.

![Graph](image)

Figure 25: (Left) The error over the number of trees in the ensemble on the adult dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the adult dataset. Rounded to the second decimal digit. Larger is better.
Figure 26: (Left) The error over the number of trees in the ensemble on the anura dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the anura dataset. Rounded to the second decimal digit. Larger is better.

Figure 27: (Left) The error over the number of trees in the ensemble on the avila dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the avila dataset. Rounded to the second decimal digit. Larger is better.
APPENDIX: Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

| max_leaf_nodes | model COMP DREP IC IE LMD REmax |
|----------------|---------------------------------|
| 64             | 8.0 4.84 3.72 5.13 4.74 4.03 4.51 |
| 128            | 8.0 4.48 2.83 3.76 4.41 4.07 4.31 |
| 256            | 8.0 4.51 2.83 4.35 4.04 4.45 4.51 |
| 512            | 8.0 4.50 2.83 2.14 4.45 4.04 4.45 |
| 1024           | 8.0 4.48 2.83 1.35 4.45 4.04 4.45 |

Figure 28: (Left) The error over the number of trees in the ensemble on the bank dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the bank dataset. Rounded to the second decimal digit. Larger is better.

Figure 29: (Left) The error over the number of trees in the ensemble on the chess dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the chess dataset. Rounded to the second decimal digit. Larger is better.
Figure 30: (Left) The error over the number of trees in the ensemble on the connect dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the connect dataset. Rounded to the second decimal digit. Larger is better.

Figure 31: (Left) The error over the number of trees in the ensemble on the eeg dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the eeg dataset. Rounded to the second decimal digit. Larger is better.
APPENDIX: Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

| max_leaf_nodes | model COMP DREP IC RE LMD RE |
|----------------|-------------------------------|
| 64             | 8.0  98.88 99.07 98.96 98.83 99.03 |
|                | 16.0 99.04 99.02 99.01 99.01 99.00 |
|                | 32.0 99.01 99.00 99.00 99.01 99.08 |
|                | 64.0 99.03 99.08 99.04 99.04 99.05 |
|                | 128.0 99.07 99.03 99.09 99.06 99.04 99.05 |
| 128            | 8.0  99.08 99.02 99.11 98.98 98.98 99.04 |
|                | 16.0 99.08 99.09 99.11 99.08 99.09 99.14 |
|                | 32.0 99.14 99.12 99.09 99.08 99.09 99.14 |
|                | 64.0 99.14 99.13 99.12 99.11 99.13 99.12 |
|                | 128.0 99.14 99.16 99.11 99.12 99.15 99.14 |
| 256            | 8.0  99.03 98.98 98.98 98.98 98.98 99.00 |
|                | 16.0 99.12 99.11 99.09 99.05 99.11 99.10 |
|                | 32.0 99.13 99.18 99.06 99.12 99.14 99.19 |
|                | 64.0 99.14 99.17 99.14 99.13 99.14 99.18 |
|                | 128.0 99.18 99.17 99.14 99.12 99.14 99.15 |
| 512            | 8.0  99.01 99.06 99.06 98.96 98.98 98.98 |
|                | 16.0 99.09 99.07 99.11 99.13 99.09 99.11 |
|                | 32.0 99.16 99.19 99.14 99.13 99.14 99.12 |
|                | 64.0 99.19 99.19 99.18 99.16 99.18 99.17 |
|                | 128.0 99.22 99.16 99.17 99.14 99.17 99.16 |
| 1024           | 8.0  99.00 99.07 99.03 99.11 98.98 99.00 |
|                | 16.0 99.09 99.12 99.16 99.17 99.07 99.13 |
|                | 32.0 99.18 99.19 99.18 99.14 99.14 99.18 |
|                | 64.0 99.22 99.26 99.22 99.16 99.18 99.21 |
|                | 128.0 99.22 99.24 99.21 99.19 99.20 99.21 |

Figure 32: (Left) The error over the number of trees in the ensemble on the elec dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the elec dataset. Rounded to the second decimal digit. Larger is better.

Figure 33: (Left) The error over the number of trees in the ensemble on the ida2016 dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the ida2016 dataset. Rounded to the second decimal digit. Larger is better.
Figure 34: (Left) The error over the number of trees in the ensemble on the japanese-vowels dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the japanese-vowels dataset. Rounded to the second decimal digit. Larger is better.

Figure 35: (Left) The error over the number of trees in the ensemble on the magic dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the magic dataset. Rounded to the second decimal digit. Larger is better.
Figure 36: (Left) The error over the number of trees in the ensemble on the mnist dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the mnist dataset. Rounded to the second decimal digit. Larger is better.

Figure 37: (Left) The error over the number of trees in the ensemble on the mozilla dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the mozilla dataset. Rounded to the second decimal digit. Larger is better.
Figure 38: (Left) The error over the number of trees in the ensemble on the nomao dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the nomao dataset. Rounded to the second decimal digit. Larger is better.

Figure 39: (Left) The error over the number of trees in the ensemble on the postures dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the postures dataset. Rounded to the second decimal digit. Larger is better.
APPENDIX: Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

### Figure 40

(Left) The error over the number of trees in the ensemble on the satimage dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the satimage dataset. Rounded to the second decimal digit. Larger is better.

### 5.1 Plotting the Pareto Front For More Datasets with Dedicated Pruning Set

![Figure 41: (left) 5-fold cross-validation accuracy on the adult dataset. (right) 5-fold cross-validation accuracy on the anura dataset.](image-url)
Figure 42: (left) 5-fold cross-validation accuracy on the avila dataset. (right) 5-fold cross-validation accuracy on the bank dataset.

Figure 43: (left) 5-fold cross-validation accuracy on the chess dataset. (right) 5-fold cross-validation accuracy on the connect dataset.

Figure 44: (left) 5-fold cross-validation accuracy on the eeg dataset. (right) 5-fold cross-validation accuracy on the elec dataset.
Figure 45: (left) 5-fold cross-validation accuracy on the ida2016 dataset. (right) 5-fold cross-validation accuracy on the japanese-vowels dataset.

Figure 46: (left) 5-fold cross-validation accuracy on the magic dataset. (right) 5-fold cross-validation accuracy on the mnist dataset.

Figure 47: (left) 5-fold cross-validation accuracy on the mozilla dataset. (right) 5-fold cross-validation accuracy on the nomao dataset.
Table 6: Test accuracies for models with a memory consumption below 32 KB for each method and each dataset averaged over a 5 fold cross validation using a dedicated pruning set. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model         | CA   | COMP | DREP | GB   | IC   | IE   | LMD  | RE   | RF   | RF-LR |
|---------------|------|------|------|------|------|------|------|------|------|-------|
| adult         | 85.102 | 85.587 | 85.400 | 86.244 | 85.381 | 85.756 | 85.071 | 86.085 | 85.188 | 85.541 |
| anura         | 85.360 | 85.087 | 85.150 | 91.564 | 96.650 | 95.866 | 96.317 | 95.928 | 96.678 |
| avila         | 89.717 | 88.887 | 89.830 | 90.527 | 90.053 | 90.184 | 89.695 | 90.140 | 89.775 |
| bank          | 43.834 | 46.571 | 45.801 | 30.175 | 46.970 | 43.814 | 47.762 | 43.545 | 48.065 |
| chess         | 70.771 | 72.530 | 72.540 | 67.584 | 72.524 | 72.993 | 71.360 | 73.197 | 71.100 |
| connect       | 81.509 | 82.590 | 81.569 | 79.226 | 82.029 | 82.677 | 81.202 | 82.041 | 81.515 |
| eeg           | 82.080 | 81.502 | 82.398 | 81.941 | 80.407 | 83.205 | 80.941 | 83.082 | 82.067 |
| ida2016       | 98.988 | 98.931 | 99.069 | 99.038 | 98.994 | 98.962 | 98.831 | 99.031 | 98.906 |
| japanese-vowels | 86.287 | 89.017 | 87.321 | 81.930 | 89.178 | 87.702 | 87.351 | 88.847 | 87.391 |
| magic         | 85.762 | 86.161 | 85.714 | 86.629 | 86.240 | 85.961 | 86.067 | 86.445 | 85.814 |
| mnist         | 79.900 | 80.100 | 79.000 | 70.600 | 78.100 | 80.100 | 77.300 | 80.100 | 77.400 |
| mozilla       | 94.378 | 94.519 | 94.538 | 94.493 | 94.577 | 94.622 | 94.210 | 94.641 | 94.474 |
| nomao         | 94.998 | 95.259 | 95.271 | 95.273 | 95.250 | 95.323 | 94.972 | 95.404 | 95.152 |
| postures      | 74.165 | 76.115 | 75.412 | 64.126 | 75.319 | 76.093 | 73.922 | 76.518 | 73.986 |
| satimage      | 87.372 | 87.683 | 87.247 | 85.925 | 87.481 | 87.434 | 87.030 | 87.683 | 87.185 |

Figure 48: (left) 5-fold cross-validation accuracy on the postures dataset. (right) 5-fold cross-validation accuracy on the satimage dataset.
Table 7: Test accuracies for models with a memory consumption below 64 KB for each method and each dataset averaged over a 5 fold cross validation using a dedicated pruning set. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model          | CA   | COMP | DREP | GB   | IC   | IE   | LMD  | RE   | RF   | RF-LR |
|----------------|------|------|------|------|------|------|------|------|------|-------|
| adult          | 85.179 | 85.639 | 85.523 | **86.616** | 85.609 | 85.928 | 85.172 | 86.085 | 85.507 | 86.075 |
| anura          | 95.719 | 96.692 | 96.692 | 95.024 | 96.845 | 96.599 | 96.498 | 97.040 | 96.289 | **97.054** |
| avila          | 89.610 | 94.522 | 93.171 | 67.724 | 94.719 | **94.930** | 94.863 | 88.690 | 91.757 |
| bank           | 89.741 | 90.113 | 89.941 | **90.677** | 90.230 | 90.206 | 89.733 | 90.248 | 89.775 | 90.151 |
| chess          | 48.314 | 50.235 | 48.849 | 34.969 | 50.307 | 50.403 | 48.236 | 50.834 | 47.220 | **53.728** |
| connect        | 72.465 | 73.408 | 73.351 | 70.847 | 73.473 | 73.813 | 72.043 | 74.075 | 72.244 | **76.676** |
| eeg            | 83.091 | 84.766 | 83.718 | 82.463 | 84.399 | 84.272 | 83.478 | 84.619 | 83.271 | **85.501** |
| electric       | 83.073 | 83.038 | 83.333 | 83.878 | 82.018 | 83.940 | 82.289 | 84.176 | 82.826 | **84.040** |
| ida2016        | 99.012 | 99.081 | 99.069 | 99.094 | 99.106 | 99.006 | 99.012 | 99.044 | 99.050 | **99.131** |
| japanese-vowels| 89.459 | 90.955 | 89.840 | 87.341 | 90.884 | 90.061 | 90.212 | 90.915 | 89.589 | **91.668** |
| magic          | 86.003 | 86.624 | 86.124 | **87.355** | 86.598 | 86.303 | 86.445 | 86.503 | 86.245 | 86.892 |
| mnist          | 82.800 | 84.700 | 82.900 | 74.400 | 84.000 | 83.100 | 83.900 | 84.200 | 83.700 |
| nomao          | 95.358 | 95.564 | 95.491 | **95.958** | 95.610 | 95.555 | 95.331 | 95.575 | 95.363 | 95.784 |
| postures       | 79.571 | 80.910 | 80.392 | 69.058 | 80.513 | 81.127 | 79.857 | **81.268** | 79.476 | 81.161 |
| satimage       | 87.869 | 88.694 | 88.305 | 87.449 | **88.849** | 88.165 | 87.776 | 88.585 | 88.616 | 88.305 |

Table 8: Test accuracies for models with a memory consumption below 128 KB for each method and each dataset averaged over a 5 fold cross validation using a dedicated pruning set. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model          | CA   | COMP | DREP | GB   | IC   | IE   | LMD  | RE   | RF   | RF-LR |
|----------------|------|------|------|------|------|------|------|------|------|-------|
| adult          | 85.741 | 85.854 | 85.710 | **87.135** | 85.839 | 86.060 | 85.434 | 86.085 | 85.633 | 86.269 |
| anura          | 90.386 | 97.262 | 97.095 | 96.553 | 97.331 | 97.582 | 97.165 | 97.373 | 97.067 | **97.609** |
| avila          | 92.107 | 97.623 | 95.067 | 75.262 | 97.791 | 97.738 | 88.154 | **97.796** | 92.126 | 95.534 |
| bank           | 89.876 | 90.184 | 89.980 | **90.737** | 90.259 | 90.206 | 89.949 | 90.268 | 89.978 | 90.644 |
| chess          | 51.066 | 54.926 | 51.900 | 40.719 | 53.682 | 53.486 | 52.146 | 54.716 | 50.150 | **57.018** |
| connect        | 73.270 | 74.738 | 74.105 | 73.850 | 74.969 | 74.854 | 73.972 | 75.147 | 73.354 | **77.865** |
| eeg            | 85.547 | 86.595 | 85.861 | 85.374 | 86.976 | 86.435 | 85.314 | 86.182 | 85.107 | **87.977** |
| electric       | 83.647 | 84.192 | 84.454 | **85.748** | 84.324 | 85.050 | 83.453 | 85.068 | 84.223 | 85.454 |
| ida2016        | 99.012 | 99.081 | 99.087 | **99.169** | 99.106 | 99.106 | 99.119 | 99.112 | 99.087 | 99.131 |
| japanese-vowels| 90.433 | 92.461 | 91.688 | 91.748 | 92.471 | 91.447 | 92.069 | 92.390 | 91.878 | **93.575** |
| magic          | 86.251 | 86.818 | 86.508 | **87.733** | 86.887 | 86.613 | 86.692 | 86.813 | 86.603 | 87.255 |
| mnist          | 85.900 | **88.100** | 86.200 | 79.300 | 87.000 | 85.900 | 87.100 | 85.900 | 86.400 | 87.200 |
| mozilla        | 94.455 | 94.738 | 94.603 | **94.989** | 94.821 | 94.802 | 94.526 | 94.751 | 94.603 | 94.667 |
| nomao          | 95.697 | 95.891 | 95.833 | **96.408** | 95.897 | 95.854 | 95.764 | 95.941 | 95.825 | 96.089 |
| postures       | 83.912 | 85.294 | 84.616 | 75.760 | 84.726 | 85.003 | 83.849 | 85.527 | 84.016 | **85.691** |
| satimage       | 88.243 | 89.176 | 88.802 | 88.631 | 89.145 | 88.849 | 89.082 | **89.440** | 88.740 | 89.440 |
Table 9: Test accuracies for models with a memory consumption below 256 KB for each method and each dataset averaged over a 5 fold cross validation using a dedicated pruning set. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

5.2 Accuracies under various resource constraints with Dedicated Pruning Set

5.3 Area Under the Pareto Front with Dedicated Pruning Set
6. Revisiting Ensemble Pruning with a Bagging Classifier

For space reasons, the paper focuses on Random Forest classifier. Here we will repeat our experiment with a Bagging Classifier implemented in Scikit-Learn. As before, we either use a 5-fold cross validation or the given test/train split. For reference, recall our experimental protocol: Oshiro et al. showed in that the prediction of a RF stabilizes between 128 and 256 trees in the ensemble and adding more trees to the ensemble does not yield significantly better results. Hence, we train the ‘base’ Random Forests with $M = 256$ trees. To control the individual errors of trees we set the maximum number of leaf nodes $n_l$ to values between $n_l \in \{64, 128, 256, 512, 1024\}$. For ensemble pruning we use RE and compare it against a random selection of trees from the original ensemble (which is the same a training a smaller forest directly). In both cases a sub-ensemble with $K \in \{2, 4, 8, 16, 32, 64, 128, 256\}$ members is selected so that for $K = 256$ the original RF is recovered.
Figure 50: (Left) The error over the number of trees in the ensemble on the adult dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the adult dataset. Rounded to the second decimal digit. Larger is better.

Figure 51: (Left) The error over the number of trees in the ensemble on the anura dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the anura dataset. Rounded to the second decimal digit. Larger is better.
APPENDIX: Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|-----|
| 64             | 8.0   | 81.87| 79.17| 79.22| 78.95| 80.45| 81.30|
| 128            | 8.0   | 80.74| 79.69| 80.14| 79.23| 80.44| 82.02|
| 256            | 8.0   | 80.86| 79.49| 80.03| 79.52| 80.86| 82.05|
| 512            | 8.0   | 80.52| 79.45| 80.25| 79.26| 79.75| 81.85|
| 1024           | 8.0   | 80.20| NaN  | 80.16| 79.16| 79.60| 81.21|
| 128            | 8.0   | 90.28| 89.67| 90.37| 89.24| 89.15| 91.16|
| 256            | 8.0   | 90.54| 89.20| 90.54| 89.68| 89.60| 92.02|
| 512            | 8.0   | 90.21| NaN  | 90.21| 89.52| 89.50| 91.60|
| 1024           | 8.0   | 90.02| NaN  | 89.89| 89.22| 89.33| 90.89|
| 128            | 8.0   | 97.72| 96.27| 97.50| 97.36| 94.64| 97.89|
| 256            | 8.0   | 97.65| 96.35| 97.61| 97.34| 95.95| 97.92|
| 512            | 8.0   | 97.42| NaN  | 97.43| 97.26| 95.17| 97.63|
| 1024           | 8.0   | 97.08| NaN  | 97.06| 96.91| 95.21| 97.29|
| 128            | 8.0   | 99.34| 99.09| 99.36| 99.34| 97.49| 99.34|
| 256            | 8.0   | 99.37| 99.23| 99.47| 99.44| 97.66| 99.38|
| 512            | 8.0   | 99.40| 99.29| 99.47| 99.47| 97.82| 99.43|
| 1024           | 8.0   | 99.42| 99.34| 99.47| 99.49| 98.03| 99.49|
| 128            | 8.0   | 99.43| NaN  | 99.45| 99.44| 98.53| 99.41|
| 256            | 8.0   | 99.39| 99.37| 99.46| 99.42| 98.77| 99.39|
| 512            | 8.0   | 99.40| 99.38| 99.46| 99.42| 98.91| 99.41|
| 1024           | 8.0   | 99.43| NaN  | 99.43| 99.44| 99.17| 99.44|

Figure 52: (Left) The error over the number of trees in the ensemble on the avila dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the avila dataset. Rounded to the second decimal digit. Larger is better.

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|-----|
| 64             | 8.0   | 90.55| 90.49| 90.38| 90.49| 90.27| 90.51|
| 128            | 8.0   | 90.55| 90.58| 90.56| 90.52| 90.47| 90.53|
| 256            | 8.0   | 90.61| 90.54| 90.62| 90.57| 90.54| 90.64|
| 512            | 8.0   | 90.64| 90.65| 90.72| 90.65| 90.62| 90.64|
| 1024           | 8.0   | 90.56| 90.59| 90.59| 90.58| 90.60| 90.64|
| 128            | 8.0   | 90.57| 90.60| 90.63| 90.59| 90.54| 90.60|
| 256            | 8.0   | 90.64| 90.66| 90.64| 90.65| 90.60| 90.64|
| 512            | 8.0   | 90.65| 90.67| 90.71| 90.65| 90.71| 90.69|
| 1024           | 8.0   | 90.68| 90.68| 90.68| 90.65| 90.61| 90.79|
| 128            | 8.0   | 90.73| 90.73| 90.73| 90.65| 90.66| 90.73|
| 256            | 8.0   | 90.76| 90.76| 90.76| 90.70| 90.68| 90.73|
| 512            | 8.0   | 90.70| 90.72| 90.76| 90.76| 90.70| 90.72|
| 1024           | 8.0   | 90.40| 90.32| 90.62| 90.33| 90.40| 90.48|
| 128            | 8.0   | 90.52| 90.51| 90.64| 90.58| 90.54| 90.55|
| 256            | 8.0   | 90.62| 90.57| 90.68| 90.65| 90.61| 90.60|
| 512            | 8.0   | 90.64| 90.64| 90.65| 90.69| 90.67| 90.63|
| 1024           | 8.0   | 90.67| 90.72| 90.69| 90.69| 90.66| 90.66|
| 128            | 8.0   | 90.70| 90.72| 90.76| 90.70| 90.70| 90.70|
| 256            | 8.0   | 90.67| 90.72| 90.69| 90.69| 90.66| 90.66|
| 512            | 8.0   | 90.68| 90.70| 90.70| 90.70| 90.70| 90.70|
| 1024           | 8.0   | 90.65| 90.64| 90.73| 90.69| 90.66| 90.64|

Figure 53: (Left) The error over the number of trees in the ensemble on the bank dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the bank dataset. Rounded to the second decimal digit. Larger is better.
Figure 54: (Left) The error over the number of trees in the ensemble on the chess dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the chess dataset. Rounded to the second decimal digit. Larger is better.

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|----|
| 64             | 8.0   | 44.40| 43.93| 43.35| 43.99| 44.36| 45.63|
| 128            | 8.0   | 50.34| 49.80| 50.35| 50.19| 49.62| 51.69|
| 256            | 8.0   | 50.71| 49.59| 50.53| 50.55| 50.02| 51.54|
| 512            | 8.0   | 50.75| 49.80| 50.73| 50.19| 50.17| 51.63|
| 1024           | 8.0   | 50.63| 50.58| 50.16| 50.28| 51.53|       |

Figure 55: (Left) The error over the number of trees in the ensemble on the connect dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the connect dataset. Rounded to the second decimal digit. Larger is better.
APPENDIX: Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|----|
| 64             | 8.0   | 83.43| 82.75| 83.14| 83.02| 82.94| 83.47|
| 16.0           | 83.99| 83.24| 83.71| 83.26| 83.34| 83.96|    |
| 32.0           | 84.10| 83.34| 83.92| 83.71| 83.68| 84.26|    |
| 64.0           | 84.04| 83.45| 84.05| 83.57| 83.68| 84.11|    |
| 128.0          | 83.85| 83.44| 83.79| 83.72| 83.55| 83.93|    |
| 128.0          | 83.85| 83.44| 83.79| 83.72| 83.55| 83.93|    |
| 16.0           | 87.15| 86.07| 86.86| 86.70| 86.49| 87.14|    |
| 32.0           | 87.49| 86.42| 87.25| 87.06| 86.96| 87.58|    |
| 64.0           | 87.41| 86.70| 87.38| 87.19| 87.01| 87.26|    |
| 128.0          | 87.37| 86.78| 87.42| 87.08| 86.93| 87.16|    |
| 256            | 8.0   | 88.30| 87.38| 88.12| 88.11| 87.66| 88.55|
| 16.0           | 89.45| 88.58| 89.10| 89.07| 88.83| 89.05|    |
| 32.0           | 89.71| 88.92| 89.81| 89.42| 89.21| 89.51|    |
| 64.0           | 89.80| 89.13| 89.67| 89.67| 89.31| 89.64|    |
| 128.0          | 89.75| 89.27| 89.81| 89.57| 89.47| 89.67|    |
| 512            | 8.0   | 89.26| 89.10| 89.35| 89.61| 88.82| 89.75|
| 16.0           | 90.51| 90.18| 90.71| 90.93| 90.67| 90.58|    |
| 32.0           | 91.67| 90.86| 91.49| 91.36| 90.62| 91.01|    |
| 64.0           | 91.37| 91.02| 91.77| 91.44| 90.95| 91.25|    |
| 128.0          | 91.36| 91.16| 91.55| 91.48| 91.16| 91.35|    |
| 1024           | 8.0   | 89.98| 89.36| 89.84| 89.95| 89.03| 89.77|
| 16.0           | 91.30| 91.03| 91.82| 91.48| 90.77| 90.96|    |
| 32.0           | 92.06| 91.97| 92.60| 92.32| 91.55| 91.98|    |
| 64.0           | 92.32| 92.32| 92.76| 92.66| 92.07| 92.36|    |
| 128.0          | 92.68| 92.66| 92.90| 92.75| 92.54| 92.67|    |

Figure 56: (Left) The error over the number of trees in the ensemble on the eeg dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the eeg dataset. Rounded to the second decimal digit. Larger is better.

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|----|
| 64             | 8.0   | 83.77| 83.82| 83.98| 83.97| 83.83| 84.43|
| 16.0           | 84.11| 83.87| 84.04| 84.02| 84.27| 84.53|    |
| 32.0           | 84.26| 84.01| 84.11| 84.04| 84.13| 84.59|    |
| 64.0           | 84.16| 84.04| 84.17| 84.05| 84.02| 84.45|    |
| 128.0          | 84.07| 84.02| 84.08| 84.07| 84.05| 84.28|    |
| 256            | 8.0   | 86.20| 85.76| 86.00| 86.18| 85.79| 86.50|
| 16.0           | 86.29| 85.81| 86.20| 86.01| 86.03| 86.64|    |
| 32.0           | 86.36| 85.92| 86.30| 86.04| 86.06| 86.60|    |
| 64.0           | 86.29| 86.07| 86.25| 86.15| 86.09| 86.52|    |
| 128.0          | 86.28| 86.13| 86.17| 86.12| 86.16| 86.28|    |
| 512            | 8.0   | 87.88| 87.64| 88.03| 88.02| 87.59| 88.18|
| 16.0           | 88.24| 87.93| 88.29| 88.16| 87.90| 88.38|    |
| 32.0           | 88.42| 88.00| 88.39| 88.25| 88.14| 88.44|    |
| 64.0           | 88.35| 88.12| 88.36| 88.21| 88.19| 88.44|    |
| 128.0          | 88.34| 88.18| 88.33| 88.27| 88.23| 88.38|    |
| 35 | 512 | 8.0 | 89.41| 89.10| 89.36| 89.41| 89.03| 89.56|    |
| 16.0           | 89.76| 89.48| 89.76| 89.67| 89.34| 89.76|    |
| 32.0           | 89.82| 89.65| 89.80| 89.86| 89.57| 89.87|    |
| 64.0           | 89.87| 89.69| 89.85| 89.93| 89.66| 89.93|    |
| 128.0          | 89.90| 89.74| 89.90| 89.92| 89.75| 89.88|    |
| 1024           | 8.0   | 90.60| 90.29| 90.52| 90.59| 90.62| 90.51|    |
| 16.0           | 90.87| 90.76| 90.87| 90.86| 90.45| 90.84|    |
| 32.0           | 91.65| 90.88| 91.18| 91.09| 90.55| 90.94|    |
| 64.0           | 91.90| 90.96| 91.11| 91.11| 90.79| 91.06|    |
| 128.0          | 91.99| 91.02| 91.17| 91.09| 90.93| 91.02|    |

Figure 57: (Left) The error over the number of trees in the ensemble on the elec dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the elec dataset. Rounded to the second decimal digit. Larger is better.
Figure 58: (Left) The error over the number of trees in the ensemble on the ida2016 dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the ida2016 dataset. Rounded to the second decimal digit. Larger is better.

Figure 59: (Left) The error over the number of trees in the ensemble on the japanese-vowels dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the japanese-vowels dataset. Rounded to the second decimal digit. Larger is better.
APPENDIX: Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

Figure 60: (Left) The error over the number of trees in the ensemble on the magic dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the magic dataset. Rounded to the second decimal digit. Larger is better.

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|----|
| 64             | 8.0   | 86.35| 86.15| 86.25| 86.24| 86.11| 86.37|
| 128            | 8.0   | 86.49| 86.20| 86.35| 86.26| 86.30| 86.59|
| 512            | 8.0   | 86.49| 86.35| 86.43| 86.34| 86.40| 86.67|
| 256            | 8.0   | 86.52| 86.41| 86.46| 86.43| 86.47| 86.44|
| 1024           | 8.0   | 86.50| 86.45| 86.46| 86.43| 86.46| 86.53|
|                | 16.0  | 86.63| 86.61| 86.71| 86.63| 86.68| 86.68|
|                | 32.0  | 86.81| 86.63| 86.80| 86.88| 86.77| 86.94|
|                | 64.0  | 86.87| 86.89| 86.91| 86.83| 86.65| 86.94|
|                | 128.0 | 87.01| 87.02| 87.04| 87.04| 86.90| 86.91|
|                | 512   | 87.11| 86.89| 87.13| 87.13| 87.15| 87.07|
|                | 1024  | 87.28| 87.18| 87.33| 87.13| 87.23| 87.35|

Figure 61: (Left) The error over the number of trees in the ensemble on the mnist dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the mnist dataset. Rounded to the second decimal digit. Larger is better.

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|----|
| 64             | 8.0   | 84.7 | 84.3 | 85.6 | 82.6 | 84.4 | 85.4 |
| 128            | 8.0   | 86.2 | 82.1 | 85.4 | 82.9 | 85.3 | 86.0 |
| 512            | 8.0   | 86.2 | 86.1 | 85.6 | 82.4 | 86.6 | 87.0 |
| 256            | 8.0   | 86.6 | 86.4 | 85.9 | 83.4 | 87.2 | 87.1 |
| 1024           | 8.0   | 87.8 | 87.4 | 88.4 | 88.4 | 88.4 | 88.4 |
|                | 16.0  | 89.3 | 88.1 | 88.6 | 87.4 | 89.6 | 88.9 |
|                | 32.0  | 89.3 | 89.3 | 89.4 | 86.7 | 89.4 | 89.5 |
|                | 64.0  | 89.5 | 89.1 | 89.5 | 88.4 | 89.0 | 89.0 |
|                | 128.0 | 89.3 | 89.0 | 89.5 | 89.0 | 89.5 | 88.7 |
|                | 512   | 89.0 | 89.0 | 89.0 | 90.0 | 90.0 | 90.0 |
|                | 1024  | 91.5 | 90.8 | 90.4 | 90.9 | 91.5 | 90.8 |

37
### Figure 62: (Left) The error over the number of trees in the ensemble on the mozilla dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the mozilla dataset. Rounded to the second decimal digit. Larger is better.

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|----|
| 64             | 8.0   | 94.64 | 94.62 | 94.70 | 94.70 | 94.62 | 94.82 |
|                | 16.0  | 94.68 | 94.63 | 94.71 | 94.77 | 94.61 | 94.81 |
|                | 32.0  | 94.71 | 94.60 | 94.72 | 94.74 | 94.62 | 94.73 |
|                | 64.0  | 94.69 | 94.58 | 94.69 | 94.69 | 94.59 | 94.71 |
|                | 128.0 | 94.63 | 94.58 | 94.63 | 94.65 | 94.59 | 94.72 |
|                | 256   | 8.0   | 94.78 | 94.76 | 94.80 | 94.91 | 94.76 | 94.92 |
|                | 16.0  | 94.89 | 94.77 | 94.88 | 94.93 | 94.79 | 94.93 |
|                | 32.0  | 94.81 | 94.74 | 94.91 | 94.94 | 94.75 | 94.91 |
|                | 64.0  | 94.87 | 94.80 | 94.87 | 94.94 | 94.78 | 94.87 |
|                | 128.0 | 94.85 | 94.81 | 94.85 | 94.86 | 94.79 | 94.86 |
|                | 512   | 8.0   | 94.94 | 94.94 | 95.05 | 94.86 | 94.85 | 94.94 |
|                | 16.0  | 95.04 | 94.98 | 95.07 | 95.09 | 94.99 | 95.06 |
|                | 32.0  | 95.07 | 95.03 | 95.03 | 95.15 | 95.05 | 95.13 |
|                | 64.0  | 95.10 | 95.10 | 95.16 | 95.19 | 95.05 | 95.12 |
|                | 128.0 | 95.10 | 95.09 | 95.16 | 95.19 | 95.09 | 95.14 |
|                | 256   | 8.0   | 94.79 | 94.87 | 94.89 | 94.99 | 94.92 | 94.78 |
|                | 16.0  | 95.21 | 95.14 | 95.21 | 95.12 | 95.18 | 95.23 |
|                | 32.0  | 95.29 | 95.42 | 95.34 | 95.37 | 95.32 | 95.33 |
|                | 64.0  | 95.37 | 95.39 | 95.43 | 95.39 | 95.39 | 95.39 |
|                | 128.0 | 95.41 | 95.40 | 95.44 | 95.45 | 95.39 | 95.42 |
|                | 512   | 8.0   | 94.75 | 94.69 | 94.87 | 94.90 | 94.56 | 94.85 |
|                | 16.0  | 95.08 | 95.08 | 95.25 | 95.24 | 95.17 | 95.18 |
|                | 32.0  | 95.25 | 95.18 | 95.36 | 95.27 | 95.27 | 95.27 |
|                | 64.0  | 95.35 | 95.31 | 95.37 | 95.36 | 95.25 | 95.36 |
|                | 128.0 | 95.37 | 95.36 | 95.33 | 95.34 | 95.34 | 95.39 |

### Figure 63: (Left) The error over the number of trees in the ensemble on the nomao dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the nomao dataset. Rounded to the second decimal digit. Larger is better.

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|----|
| 64             | 8.0   | 95.63 | 95.34 | 95.67 | 95.61 | 95.52 | 95.68 |
|                | 16.0  | 95.74 | 95.39 | 95.69 | 95.57 | 95.59 | 95.71 |
|                | 32.0  | 95.63 | 95.47 | 95.67 | 95.60 | 95.59 | 95.63 |
|                | 64.0  | 95.86 | 95.49 | 95.64 | 95.54 | 95.57 | 95.60 |
|                | 128.0 | 95.55 | 95.52 | 95.58 | 95.51 | 95.53 | 95.54 |
|                | 256   | 8.0   | 95.97 | 95.86 | 95.97 | 95.93 | 95.88 | 95.98 |
|                | 16.0  | 96.09 | 95.97 | 96.11 | 96.03 | 95.98 | 96.04 |
|                | 32.0  | 96.10 | 95.95 | 96.14 | 95.97 | 95.98 | 96.04 |
|                | 64.0  | 96.06 | 95.96 | 96.07 | 95.99 | 96.01 | 96.06 |
|                | 128.0 | 96.08 | 95.96 | 96.06 | 96.04 | 96.04 | 96.02 |
|                | 512   | 8.0   | 96.22 | 96.16 | 96.20 | 96.25 | 96.17 | 96.42 |
|                | 16.0  | 96.30 | 96.27 | 96.33 | 96.33 | 96.32 | 96.42 |
|                | 32.0  | 96.34 | 96.36 | 96.38 | 96.45 | 96.36 | 96.46 |
|                | 64.0  | 96.41 | 96.47 | 96.45 | 96.43 | 96.44 | 96.47 |
|                | 128.0 | 96.42 | 96.43 | 96.45 | 96.45 | 96.43 | 96.47 |
|                | 256   | 8.0   | 96.25 | 96.23 | 96.43 | 96.31 | 96.23 | 96.26 |
|                | 16.0  | 96.57 | 96.41 | 96.52 | 96.61 | 96.48 | 96.57 |
|                | 32.0  | 96.69 | 96.60 | 96.69 | 96.65 | 96.57 | 96.62 |
|                | 64.0  | 96.74 | 96.68 | 96.72 | 96.70 | 96.70 | 96.71 |
|                | 128.0 | 96.73 | 96.70 | 96.73 | 96.70 | 96.72 | 96.71 |
|                | 512   | 8.0   | 96.27 | 96.37 | 96.34 | 96.29 | 96.26 | 96.27 |
|                | 16.0  | 96.54 | 96.47 | 96.62 | 96.60 | 96.54 | 96.54 |
|                | 32.0  | 96.64 | 96.65 | 96.72 | 96.77 | 96.74 | 96.67 |
|                | 64.0  | 96.72 | 96.72 | 96.77 | 96.74 | 96.75 | 96.74 |
|                | 128.0 | 96.71 | 96.72 | 96.81 | 96.77 | 96.72 | 96.72 |
APPENDIX: IMPROVING THE ACCURACY-MEMORY TRADE-OFF OF RANDOM FORESTS VIA LEAF-REFINEMENT

Table 1: Performance of Random Forests and Reduced Error Pruning Ensembles.

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|----|
| 64             | 8.0   | 74.93| 72.36| 72.28| 72.88| 73.10| 73.37|
| 128            | 8.0   | 74.75| 72.63| 73.01| 73.20| 73.14| 73.73|
| 256            | 8.0   | 81.48| 78.99| 79.79| 78.96| 80.40| 81.81|
| 512            | 8.0   | 81.72| 81.53| 79.66| 80.35| 82.07|       |
| 1024           | 8.0   | 87.21| 84.98| 84.88| 85.46| 86.49| 87.54|
|                | 16.0  | 87.78| 85.46| 86.19| 86.10| 86.49| 87.97|
|                | 32.0  | 87.86| 87.53| 86.15| 86.60| 88.01|       |
|                | 64.0  | 87.46| 87.31| 86.33| 86.47| 87.62|       |
|                | 128.0 | 86.82| 86.77| 86.45| 86.35| 86.95|       |
| 512            | 8.0   | 91.57| 89.71| 89.98| 89.82| 90.76| 91.77|
|                | 16.0  | 91.96| 90.22| 90.91| 90.40| 91.04| 92.17|
|                | 32.0  | 92.08| 92.02| 91.71| 91.11| 92.12|       |
|                | 64.0  | 91.87| 91.82| 90.89| 91.03| 91.94|       |
|                | 128.0 | 91.47| 91.34| 91.03| 90.93| 91.50|       |
| 1024           | 8.0   | 94.57| 93.06| 93.18| 93.43| 93.72| 94.53|
|                | 16.0  | 94.96| 93.53| 94.53| 93.93| 94.24| 95.00|
|                | 32.0  | 95.10| 95.05| 94.27| 94.37| 95.05|       |
|                | 64.0  | 95.02| 94.96| 94.47| 94.39| 94.92|       |
|                | 128.0 | 94.69| 94.68| 94.48| 94.35| 94.66|       |

Figure 64: (Left) The error over the number of trees in the ensemble on the postures dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the postures dataset. Rounded to the second decimal digit. Larger is better.

Figure 65: (Left) The error over the number of trees in the ensemble on the satimage dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the satimage dataset. Rounded to the second decimal digit. Larger is better.
### Table 10: Test accuracies for models with a memory consumption below 32 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model     | Bag-LR | Bag. | CA | COMP | DREP | GB | IC | IE | LMD | RE |
|-----------|--------|------|----|------|------|----|----|----|-----|----|
| adult     | 86.045 | 86.204 | 86.118 | 86.260 | 86.254 | 86.244 | 86.155 | 86.315 | 86.082 | 86.441 |
| anura     | 96.081 | 95.386 | 95.817 | 96.081 | 95.622 | 91.564 | 96.081 | 96.067 | 95.942 | 95.830 |
| avila     | 95.193 | 94.843 | 93.200 | 97.125 | 96.545 | 59.496 | 97.000 | 96.947 | 93.468 | 97.446 |
| bank      | 90.507 | 90.493 | 90.558 | 90.549 | 90.487 | 90.487 | 90.487 | 90.398 | 90.511 |
| chess     | 55.314 | 54.470 | 54.088 | 56.247 | 54.705 | 30.175 | 54.324 | 55.047 | 54.081 | 56.049 |
| connect   | 73.073 | 72.968 | 72.300 | 74.174 | 73.890 | 67.584 | 74.134 | 73.738 | 74.278 | 74.414 |
| eeg       | 84.680 | 83.865 | 83.852 | 84.666 | 84.666 | 84.666 | 84.666 | 84.666 | 84.666 | 85.060 |
| elec      | 86.675 | 86.202 | 86.456 | 86.333 | 86.743 | 81.914 | 86.723 | 86.871 | 86.207 | 86.990 |
| ida2016   | 99.106 | 99.119 | 99.038 | 99.112 | 99.138 | 99.038 | 99.038 | 99.112 | 99.075 | 99.125 |
| japanese-vowels | 88.274 | 86.538 | 86.166 | 87.873 | 87.080 | 81.930 | 87.501 | 87.290 | 87.622 | 87.923 |
| magic     | 86.187 | 86.251 | 85.993 | 86.345 | 86.245 | 86.254 | 86.395 | 86.250 | 86.441 |
| mnist     | 82.800 | 82.100 | 83.200 | 83.000 | 80.600 | 70.600 | 84.200 | 81.000 | 81.700 | 83.300 |
| mozilla   | 94.596 | 94.616 | 94.129 | 94.738 | 94.622 | 94.393 | 94.719 | 94.699 | 94.622 | 94.821 |
| nomao     | 95.656 | 95.645 | 95.645 | 95.848 | 95.616 | 95.273 | 95.799 | 95.726 | 95.671 | 95.761 |
| postures  | 75.707 | 76.366 | 75.421 | 78.615 | 77.029 | 64.126 | 75.705 | 76.337 | 78.192 | 78.566 |
| satimage  | 87.823 | 87.543 | 87.574 | 88.212 | 87.900 | 85.925 | 87.543 | 87.838 | 87.900 | 88.383 |

### Table 11: Test accuracies for models with a memory consumption below 64 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model     | Bag-LR | Bag. | CA | COMP | DREP | GB | IC | IE | LMD | RE |
|-----------|--------|------|----|------|------|----|----|----|-----|----|
| adult     | 86.220 | 86.373 | 86.306 | 86.300 | 86.254 | 86.616 | 86.254 | 86.395 | 86.250 | 86.441 |
| anura     | 96.762 | 96.233 | 96.470 | 96.595 | 96.400 | 95.024 | 96.498 | 96.289 | 96.331 | 96.609 |
| avila     | 98.596 | 97.101 | 97.350 | 98.457 | 98.160 | 67.724 | 98.328 | 98.342 | 96.535 | 98.356 |
| bank      | 90.507 | 90.573 | 90.624 | 90.569 | 90.595 | 90.677 | 90.628 | 90.591 | 90.538 | 90.602 |
| chess     | 61.762 | 61.085 | 61.217 | 61.616 | 61.506 | 34.969 | 61.188 | 61.965 | 61.242 | 62.332 |
| connect   | 74.814 | 74.824 | 74.349 | 75.443 | 75.267 | 70.847 | 75.428 | 75.391 | 75.248 | 75.643 |
| eeg       | 86.789 | 85.621 | 85.300 | 85.943 | 85.654 | 82.463 | 86.709 | 86.068 | 85.661 | 86.642 |
| elec      | 87.805 | 87.427 | 87.264 | 87.730 | 87.813 | 83.878 | 87.725 | 88.014 | 87.189 | 88.041 |
| ida2016   | 99.200 | 99.238 | 99.112 | 99.119 | 99.212 | 99.094 | 99.119 | 99.175 | 99.156 | 99.181 |
| japanese-vowels | 90.894 | 89.338 | 88.776 | 90.342 | 89.499 | 87.341 | 89.620 | 89.830 | 90.182 | 89.760 |
| magic     | 86.671 | 86.419 | 86.519 | 86.629 | 86.613 | 87.355 | 86.708 | 86.634 | 86.477 | 86.876 |
| mnist     | 86.600 | 82.900 | 83.300 | 85.600 | 84.300 | 74.400 | 85.600 | 85.100 | 84.400 | 86.200 |
| mozilla   | 94.937 | 94.751 | 94.622 | 94.776 | 94.764 | 94.590 | 94.802 | 94.905 | 94.764 | 94.918 |
| nomao     | 96.013 | 95.717 | 95.839 | 95.967 | 95.862 | 95.958 | 95.967 | 95.929 | 95.880 | 96.121 |
| postures  | 83.236 | 82.082 | 81.839 | 83.999 | 82.767 | 69.058 | 81.880 | 82.675 | 83.356 | 84.218 |
| satimage  | 88.818 | 88.274 | 88.600 | 88.818 | 88.896 | 87.449 | 88.538 | 88.476 | 88.694 | 89.082 |
Table 12: Test accuracies for models with a memory consumption below 128 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model       | Bag-LR | Bag. | CA | COMP | DREP | GB | IC | IE | LMD | RE  |
|-------------|--------|------|----|------|------|----|----|----|-----|-----|
| adult       | 86.373 | 86.496| 86.306| 86.355| 86.370| 87.135| 86.468| 86.496| 86.450| 86.478|
| anura       | 96.928 | 96.678| 96.470| 96.845| 96.720| 96.553| 96.928| 96.734| 96.692| 96.803|
| avila       | 99.161 | 98.505| 98.428| 99.219| 98.994| 75.262| 99.157| 99.147| 97.407| 99.157|
| bank        | 90.666 | 90.659| 90.624| 90.637| 90.664| **90.737**| 90.644| 90.648| 90.602| 90.679|
| chess       | **69.087**| 67.526| 68.881| 67.412| 40.719| 67.469| 67.658| 66.895| 68.524|
| connect     | 76.461 | 75.658| 75.579| 75.926| 73.850| 76.259| 76.114| 76.048|
| eeg         | **88.959**| 86.856| 88.304| 87.383| 85.734| 88.124| 87.664| 88.551|
| elec        | **89.131**| 88.484| 88.696| 85.748| 87.771| 88.930| 88.451| 89.124|
| ida2016     | **99.275**| 99.238| 99.200| 99.212| 99.169| 99.160| 99.238| 99.156| 99.238|
| japanse-vowels | **92.541**| 91.035| 90.904| 91.828| 90.985| 91.748| 91.728| 91.878| 91.407|
| magic       | 87.060 | 86.761| 86.813| 86.634| **87.733**| 86.797| 86.882| 86.818| 86.939|
| mnist       | 97.322 | 96.300| 95.700| 94.937| 94.989| **95.047**| 94.931| 94.854| 94.937|
| mozilla     | 96.292 | 96.170| 95.979| 96.156| 96.408| 96.202| 96.251| 96.173| **96.417**|
| nomao       | 87.476 | 86.634| 86.494| 87.460| 75.760| 86.991| 87.798| 87.481| **88.558**|
| postures    | **89.658**| 89.145| 88.647| 89.316| 89.207| 88.631| 89.160| 89.331| 89.565|

Table 13: Test accuracies for models with a memory consumption below 256 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model       | Bag-LR | Bag. | CA | COMP | DREP | GB | IC | IE | LMD | RE  |
|-------------|--------|------|----|------|------|----|----|----|-----|-----|
| adult       | 86.401 | 86.511| 86.306| 86.493| 86.401| **87.135**| 86.493| 86.515| 86.465| 86.576|
| anura       | 97.040 | 96.873| 96.470| 97.040| 96.859| **97.429**| 97.248| 97.040| 97.067| 96.928|
| avila       | **99.406**| 99.224| 98.428| 99.339| 99.224| 82.820| 99.391| 99.353| 97.824| 99.363|
| bank        | 90.719 | 90.659| 90.635| 90.670| 90.664| 90.737| 90.677| 90.655| 90.710| **90.788**|
| chess       | **72.822**| 69.347| 69.365| 70.021| 69.440| 44.678| 69.554| 69.372| 68.545| 70.406|
| connect     | **77.606**| 76.577| 76.383| 77.275| 76.726| 76.170| 77.376| 77.080| 77.002| 77.333|
| eeg         | 90.641 | 88.939| 88.211| 89.099| 85.734| 89.346| 89.613| 89.533|
| elec        | 90.462 | 89.431| 89.349| 89.460| 85.748| 89.753| 89.140| 89.751|
| ida2016     | **99.275**| 99.238| 99.200| 99.231| 99.212| 99.169| 99.238| 99.181| 99.238|
| japanse-vowels | 93.866 | 93.063| 92.029| 93.444| 93.294| **94.860**| 93.424| 93.013| 93.053| 93.083|
| magic       | 87.465 | 87.013| 86.519| 87.188| 86.892| **87.733**| 87.129| 87.155| 87.071| 87.046|
| mnist       | **90.200**| 87.600| 88.300| 89.000| 84.600| 89.600| 89.700| 89.600| 89.200|
| mozilla     | 95.085 | **95.117**| 94.867| 95.040| 94.976| 94.931| 94.989| 95.072| 95.085| 94.989| 95.079|
| nomao       | **96.521**| 96.205| 96.092| 96.298| 96.274| 96.408| 96.434| 96.333| 96.315| 96.417|
| postures    | 90.705 | 90.115| 90.170| **91.119**| 90.469| 81.100| 90.203| 90.482| 90.204| 91.106|
| satimage    | **90.093**| 89.580| 89.362| 89.751| 89.705| 89.782| 89.969| 89.751| 89.969| 89.984|

41
6.1 Accuracies under various resource constraints with a BaggingClassifier

6.2 Plotting the Pareto Front For More Datasets with Dedicated Pruning Set

Figure 66: (left) 5-fold cross-validation accuracy on the adult dataset. (right) 5-fold cross-validation accuracy on the anura dataset.

Figure 67: (left) 5-fold cross-validation accuracy on the avila dataset. (right) 5-fold cross-validation accuracy on the bank dataset.

Figure 68: (left) 5-fold cross-validation accuracy on the chess dataset. (right) 5-fold cross-validation accuracy on the connect dataset.
Figure 69: (left) 5-fold cross-validation accuracy on the eeg dataset. (right) 5-fold cross-validation accuracy on the elec dataset.

Figure 70: (left) 5-fold cross-validation accuracy on the ida2016 dataset. (right) 5-fold cross-validation accuracy on the japanese-vowels dataset.

Figure 71: (left) 5-fold cross-validation accuracy on the magic dataset. (right) 5-fold cross-validation accuracy on the mnist dataset.
Figure 72: (left) 5-fold cross-validation accuracy on the mozilla dataset. (right) 5-fold cross-validation accuracy on the nomao dataset.

Figure 73: (left) 5-fold cross-validation accuracy on the postures dataset. (right) 5-fold cross-validation accuracy on the satimage dataset.

### 6.3 Area Under the Pareto Front with a Bagging Classifier

| model dataset | AUC Bag-LR | Bag. | CA | COMP | DREP | GB | IC | IE | LMD | RE |
|----------------|------------|------|----|------|------|----|----|----|-----|----|
| adult          | 0.8648     | 0.8653 | 0.8652 | 0.8654 | 0.8653 | 0.8705 | 0.8653 | 0.8654 | 0.8653 | 0.8654 |
| anura          | 0.9686     | 0.9679 | 0.9638 | 0.9684 | 0.9681 | 0.9746 | 0.9724 | 0.9684 | 0.9693 | 0.9685 |
| avila          | 0.9930     | 0.9926 | 0.9885 | 0.9930 | 0.9926 | 0.9864 | 0.9934 | 0.9936 | 0.9892 | 0.9931 |
| bank           | 0.9079     | 0.9069 | 0.9067 | 0.9071 | 0.9070 | 0.9065 | 0.9071 | 0.9071 | 0.9068 | 0.9074 |
| chess          | 0.7627     | 0.7129 | 0.7057 | 0.7174 | 0.7127 | 0.6265 | 0.7194 | 0.7150 | 0.7157 | 0.7214 |
| connect        | 0.7947     | 0.7768 | 0.7736 | 0.7805 | 0.7768 | 0.7777 | 0.7806 | 0.7794 | 0.7782 | 0.7808 |
| eeg            | 0.9268     | 0.9215 | 0.9050 | 0.9227 | 0.9222 | 0.8562 | 0.9256 | 0.9241 | 0.9206 | 0.9225 |
| elec           | 0.9192     | 0.9082 | 0.9050 | 0.9097 | 0.9086 | 0.8566 | 0.9104 | 0.9099 | 0.9072 | 0.9092 |
| ida2016        | 0.9906     | 0.9906 | 0.9901 | 0.9904 | 0.9903 | 0.9878 | 0.9905 | 0.9904 | 0.9902 | 0.9906 |
| japanese-vowels | 0.9484    | 0.9448 | 0.9301 | 0.9461 | 0.9456 | 0.9721 | 0.9493 | 0.9460 | 0.9463 | 0.9456 |
| magic          | 0.8780     | 0.8758 | 0.8741 | 0.8771 | 0.8769 | 0.8765 | 0.8773 | 0.8769 | 0.8768 | 0.8767 |
| mnist          | 0.9218     | 0.9135 | 0.9034 | 0.9180 | 0.9173 | 0.9358 | 0.9187 | 0.9178 | 0.9198 | 0.9185 |
| mozilla        | 0.9533     | 0.9538 | 0.9502 | 0.9527 | 0.9529 | 0.9483 | 0.9530 | 0.9531 | 0.9526 | 0.9528 |
| nomao          | 0.9669     | 0.9662 | 0.9643 | 0.9664 | 0.9661 | 0.9627 | 0.9669 | 0.9667 | 0.9664 | 0.9664 |
| postures       | 0.9577     | 0.9405 | 0.9343 | 0.9493 | 0.9375 | 0.9104 | 0.9480 | 0.9425 | 0.9421 | 0.9489 |
| satimage       | 0.9086     | 0.9063 | 0.9009 | 0.9072 | 0.9076 | 0.9102 | 0.9083 | 0.9063 | 0.9069 | 0.9072 |
7. Revisiting Ensemble Pruning with ExtaTrees Classifier

For space reasons, the paper focuses on Random Forest classifier. Here we will repeat our experiment with ExtaTreesClassifier implemented in Scikit-Learn?. As before, we either use a 5-fold cross validation or the given test/train split. For reference, recall our experimental protocol: Oshiro et al. showed in ? that the prediction of a RF stabilizes between 128 and 256 trees in the ensemble and adding more trees to the ensemble does not yield significantly better results. Hence, we train the ‘base’ Random Forests with $M = 256$ trees. To control the individual errors of trees we set the maximum number of leaf nodes $n_l$ to values between $n_l \in \{64, 128, 256, 512, 1024\}$. For ensemble pruning we use RE and compare it against a random selection of trees from the original ensemble (which is the same as training a smaller forest directly). In both cases a sub-ensemble with $K \in \{2, 4, 8, 16, 32, 64, 128, 256\}$ members is selected so that for $K = 256$ the original RF is recovered.
Figure 75: (Left) The error over the number of trees in the ensemble on the adult dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the adult dataset. Rounded to the second decimal digit. Larger is better.

Figure 76: (Left) The error over the number of trees in the ensemble on the anura dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the anura dataset. Rounded to the second decimal digit. Larger is better.
Figure 77: (Left) The error over the number of trees in the ensemble on the avila dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the avila dataset. Rounded to the second decimal digit. Larger is better.

Figure 78: (Left) The error over the number of trees in the ensemble on the bank dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the bank dataset. Rounded to the second decimal digit. Larger is better.
Figure 79: (Left) The error over the number of trees in the ensemble on the chess dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the chess dataset. Rounded to the second decimal digit. Larger is better.

Figure 80: (Left) The error over the number of trees in the ensemble on the connect dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the connect dataset. Rounded to the second decimal digit. Larger is better.
Figure 81: (Left) The error over the number of trees in the ensemble on the eeg dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the eeg dataset. Rounded to the second decimal digit. Larger is better.

Figure 82: (Left) The error over the number of trees in the ensemble on the elec dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the elec dataset. Rounded to the second decimal digit. Larger is better.
Figure 83: (Left) The error over the number of trees in the ensemble on the ida2016 dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the ida2016 dataset. Rounded to the second decimal digit. Larger is better.

Figure 84: (Left) The error over the number of trees in the ensemble on the japanese-vowels dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the japanese-vowels dataset. Rounded to the second decimal digit. Larger is better.
APPENDIX: Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|----|
| 64             | 8.0   | 84.12| 83.48| 84.54| 84.52| 83.22| 85.26|
| 16.0           | 84.74| 83.73| 84.99| 84.74| 83.81| 85.48|
| 32.0           | 84.90| 83.72| 85.03| 84.58| 84.13| 85.15|
| 64.0           | 84.90| 83.73| 84.92| 84.47| 84.11| 85.01|
| 128.0          | 84.56| 83.78| 84.53| 84.14| 84.10| 84.65|
| 16.0           | 85.21| 84.85| 85.73| 85.24| 84.77| 85.84|
| 32.0           | 85.47| 84.82| 85.77| 85.27| 85.04| 85.98|
| 64.0           | 85.49| 84.80| 85.66| 85.23| 84.96| 85.75|
| 128.0          | 85.44| 84.81| 85.51| 85.09| 85.00| 85.51|
| 8.0            | 85.38| 84.89| 85.80| 85.70| 84.95| 85.86|
| 16.0           | 85.97| 85.48| 86.21| 86.01| 85.32| 86.27|
| 32.0           | 86.23| 85.61| 86.35| 85.99| 85.71| 86.19|
| 64.0           | 86.21| 85.55| 86.39| 85.93| 85.80| 86.14|
| 128.0          | 86.10| 85.60| 86.19| 85.71| 85.88| 86.11|
| 512            | 8.0   | 85.96| 85.65| 86.04| 85.94| 84.97| 86.06|
| 16.0           | 86.46| 86.27| 86.82| 86.35| 85.98| 86.39|
| 32.0           | 86.63| 86.05| 86.92| 86.55| 86.45| 86.53|
| 64.0           | 86.67| 86.18| 86.74| 86.55| 86.46| 86.58|
| 128.0          | 86.74| 86.23| 86.77| 86.55| 86.51| 86.62|
| 256            | 8.0   | 87.79| 87.69| 86.24| 86.10| 85.73| 86.07|
| 16.0           | 86.51| 86.21| 86.77| 86.59| 86.26| 86.63|
| 32.0           | 86.79| 86.71| 87.03| 86.86| 86.53| 86.86|
| 64.0           | 87.09| 86.97| 87.11| 86.97| 86.98| 86.86|
| 128.0          | 87.03| 86.89| 87.19| 87.07| 86.92| 87.02|

Figure 85: (Left) The error over the number of trees in the ensemble on the magic dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the magic dataset. Rounded to the second decimal digit. Larger is better.

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|----|
| 64             | 8.0   | 82.3 | 82.3 | 81.4 | 81.3 | 79.7 | 84.7 |
| 16.0           | 85.8 | 85.1 | 84.8 | 83.0 | 83.1 | 81.4 | 88.5 |
| 32.0           | 87.3 | 87.0 | 86.1 | 85.4 | 85.1 | 83.9 | 88.7 |
| 64.0           | 88.1 | 87.6 | 88.1 | 87.1 | 88.1 | 87.1 | 88.5 |
| 128.0          | 88.6 | 88.1 | 87.9 | 88.0 | 88.0 | 87.9 | 88.5 |
| 256            | 8.0   | 85.5 | 83.9 | 86.1 | 84.4 | 83.4 | 85.1 |
| 16.0           | 88.6 | 87.0 | 88.2 | 89.2 | 88.8 | 88.6 | 88.6 |
| 32.0           | 90.2 | 88.7 | 89.3 | 89.4 | 88.1 | 89.5 | 89.5 |
| 64.0           | 90.6 | 89.2 | 91.2 | 91.2 | 90.0 | 90.7 | 90.7 |
| 128.0          | 90.9 | 90.1 | 91.4 | 90.8 | 90.1 | 90.8 | 90.8 |
| 512            | 8.0   | 87.7 | 87.6 | 85.8 | 86.4 | 85.6 | 86.4 |
| 16.0           | 90.0 | 88.8 | 88.9 | 88.8 | 88.8 | 88.8 | 88.8 |
| 32.0           | 91.7 | 91.4 | 90.9 | 90.2 | 90.7 | 92.4 | 92.4 |
| 64.0           | 92.3 | 91.3 | 91.9 | 91.1 | 91.5 | 92.9 | 92.9 |
| 128.0          | 92.2 | 92.5 | 91.9 | 92.5 | 92.2 | 92.2 | 92.2 |
| 1024           | 8.0   | 88.3 | 86.4 | 89.1 | 88.5 | 88.4 | 87.9 |
| 16.0           | 90.5 | 88.5 | 91.0 | 91.2 | 90.5 | 90.0 | 90.0 |
| 32.0           | 91.9 | 91.5 | 92.0 | 92.1 | 91.2 | 91.9 | 91.9 |
| 64.0           | 92.1 | 92.8 | 92.8 | 92.0 | 92.1 | 92.1 | 92.1 |
| 128.0          | 93.2 | 93.0 | 93.7 | 93.5 | 92.6 | 93.1 | 93.1 |
| 1024           | 8.0   | 85.6 | 85.1 | 87.5 | 87.7 | 86.3 | 84.8 |
| 16.0           | 89.5 | 90.5 | 90.8 | 91.0 | 90.4 | 90.8 | 90.8 |
| 32.0           | 92.3 | 92.3 | 92.6 | 92.6 | 92.2 | 91.3 | 91.3 |
| 64.0           | 93.3 | 93.1 | 93.1 | 93.6 | 92.7 | 92.6 | 92.6 |
| 128.0          | 93.4 | 93.7 | 93.8 | 94.0 | 93.8 | 93.6 | 93.6 |

Figure 86: (Left) The error over the number of trees in the ensemble on the mnist dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the mnist dataset. Rounded to the second decimal digit. Larger is better.
Figure 87: (Left) The error over the number of trees in the ensemble on the mozilla dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the mozilla dataset. Rounded to the second decimal digit. Larger is better.

Figure 88: (Left) The error over the number of trees in the ensemble on the nomao dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the nomao dataset. Rounded to the second decimal digit. Larger is better.
### APPENDIX: Improving the Accuracy-Memory Trade-Off of Random Forests Via Leaf-Refinement

#### Figure 89: (Left) The error over the number of trees in the ensemble on the postures dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the postures dataset. Rounded to the second decimal digit. Larger is better.

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|----|
| 64             | 8.0   | 70.21| 68.15| 69.78| 68.81| 68.17| 71.74|
| 16.0           | 72.13| 70.30| 72.34| 70.73| 71.17| 73.94|
| 32.0           | 73.11| NaN  | 73.10| 71.59| 72.71| 75.25|
| 64.0           | 73.58| NaN  | 73.37| 72.15| 73.09| 75.23|
| 128.0          | 73.38| NaN  | 73.30| 72.38| 73.12| 74.47|
| 256            | 72.76| 72.93| 73.49| 74.27| 74.04| 78.78|
| 512            | 72.80| 72.91| 73.55| 74.62| 74.70| 78.99|
| 1024           | 72.88| 73.33| 73.52| 74.68| 75.18| 79.19|

#### Figure 90: (Left) The error over the number of trees in the ensemble on the satimage dataset. Dashed lines depict the Random Forest and solid lines are the corresponding pruned ensemble via Reduced Error pruning. (Right) The 5-fold cross-validation accuracy on the satimage dataset. Rounded to the second decimal digit. Larger is better.

| max_leaf_nodes | model | COMP | DREP | IC | RE | LMD | RE |
|----------------|-------|------|------|----|----|-----|----|
| 64             | 8.0   | 87.59| 87.17| 87.85| 87.93| 87.22| 88.86|
| 16.0           | 88.16| 87.56| 88.35| 87.87| 87.78| 88.07|
| 32.0           | 88.02| 87.96| 88.41| 88.12| 87.82| 88.09|
| 64.0           | 88.34| 88.15| 88.26| 88.32| 88.53| 90.20|
| 128.0          | 88.16| 88.20| 88.27| 88.24| 88.21| 88.20|
| 256            | 88.44| 88.32| 88.77| 88.46| 87.76| 88.69|
| 512            | 88.82| 88.60| 88.18| 89.18| 88.62| 89.16|
| 1024           | 89.13| 89.18| 89.22| 89.30| 89.85| 90.24|
| 128.0          | 89.35| 89.59| 89.39| 89.30| 89.24| 89.30|
| 256            | 89.60| 89.88| 89.80| 89.39| 89.62| 89.32|
| 512            | 89.81| 89.39| 89.97| 90.12| 89.19| 89.94|
| 1024           | 89.95| 89.61| 90.14| 90.02| 89.75| 90.28|
| 128.0          | 90.19| 90.00| 90.36| 90.12| 89.98| 90.60|
| 256            | 90.22| 90.28| 90.17| 90.20| 90.08| 90.23|
| 512            | 89.27| 89.13| 89.04| 89.44| 88.15| 89.35|
| 1024           | 90.26| 89.80| 89.81| 89.45| 89.55| 90.00|
| 128.0          | 90.68| 90.40| 90.39| 90.72| 90.30| 90.42|
| 256            | 90.81| 90.51| 90.59| 90.81| 90.81| 90.67|
| 512            | 90.89| 90.90| 90.90| 91.09| 90.96| 90.95|
| 1024           | 90.42| 90.05| 89.42| 89.41| 88.88| 89.36|
| 128.0          | 90.30| 90.16| 90.31| 90.20| 89.83| 90.45|
| 256            | 90.62| 90.48| 90.76| 90.90| 90.33| 90.72|
| 512            | 90.93| 90.65| 91.07| 91.17| 90.92| 90.64|
| 1024           | 91.83| 90.89| 91.34| 91.06| 91.14| 90.85|
7.1 Plotting the Pareto Front For More Datasets with Dedicated Pruning Set

Figure 91: (left) 5-fold cross-validation accuracy on the adult dataset. (right) 5-fold cross-validation accuracy on the anura dataset.

Figure 92: (left) 5-fold cross-validation accuracy on the avila dataset. (right) 5-fold cross-validation accuracy on the bank dataset.

Figure 93: (left) 5-fold cross-validation accuracy on the chess dataset. (right) 5-fold cross-validation accuracy on the connect dataset.
Figure 94: (left) 5-fold cross-validation accuracy on the eeg dataset. (right) 5-fold cross-validation accuracy on the elec dataset.

Figure 95: (left) 5-fold cross-validation accuracy on the ida2016 dataset. (right) 5-fold cross-validation accuracy on the japanese-vowels dataset.

Figure 96: (left) 5-fold cross-validation accuracy on the magic dataset. (right) 5-fold cross-validation accuracy on the mnist dataset.
Figure 97: (left) 5-fold cross-validation accuracy on the mozilla dataset. (right) 5-fold cross-validation accuracy on the nomao dataset.

Figure 98: (left) 5-fold cross-validation accuracy on the postures dataset. (right) 5-fold cross-validation accuracy on the satimage dataset.
| model       | CA   | COMP | DREP | ET   | ET-LR | GB   | IC   | IE   | LMD  | RE   |
|-------------|------|------|------|------|-------|------|------|------|------|------|
| adult       | 83.483 | 84.248 | 84.303 | 83.655 | 84.306 | 86.244 | 84.392 | 85.031 | 83.523 | 85.123 |
| anura       | 94.329 | 95.261 | 95.038 | 94.580 | 96.178 | 91.564 | 95.427 | 95.372 | 93.871 | 95.525 |
| avila       | 66.191 | 83.548 | 80.381 | 68.735 | 74.328 | 59.496 | 83.836 | 83.874 | 73.532 | 84.406 |
| bank        | 89.427 | 89.737 | 89.578 | 89.491 | 89.772 | 90.527 | 90.117 | 90.115 | 89.496 | 90.124 |
| chess       | 40.797 | 45.577 | 48.810 | 41.121 | 48.282 | 30.175 | 43.638 | 45.897 | 43.684 | 46.650 |
| connect     | 70.036 | 72.050 | 72.013 | 70.117 | 74.735 | 67.584 | 71.808 | 72.320 | 70.546 | 72.699 |
| eeg         | 75.935 | 76.168 | 76.702 | 76.175 | 76.832 | 79.226 | 76.762 | 76.652 | 74.693 | 79.393 |
| eeg         | 75.448 | 75.980 | 78.686 | 75.872 | 77.344 | 81.941 | 78.136 | 79.063 | 74.347 | 79.844 |
| eeg         | 87.725 | 98.800 | 98.881 | 98.719 | 98.906 | 99.038 | 98.962 | 98.894 | 98.744 | 98.894 |
| japanese-vowels | 84.018 | 86.969 | 84.309 | 83.847 | 88.706 | 81.930 | 86.357 | 85.584 | 84.520 | 87.210 |
| magic       | 83.464 | 84.121 | 83.932 | 83.332 | 85.225 | 86.629 | 84.542 | 84.752 | 83.217 | 85.283 |
| mnist       | 72.000 | 78.300 | 77.700 | 76.500 | 78.600 | 70.600 | 77.700 | 73.500 | 78.800 |
| mozilla     | 90.621 | 91.875 | 92.950 | 91.534 | 93.001 | 94.493 | 93.361 | 93.355 | 88.215 | 93.573 |
| nomao       | 94.612 | 94.728 | 94.658 | 94.516 | 95.125 | 95.273 | 94.818 | 94.914 | 95.554 | 95.099 |
| postures    | 63.365 | 67.290 | 65.118 | 63.535 | 67.743 | 64.126 | 65.807 | 66.641 | 59.460 | 68.149 |
| satimage    | 86.034 | 86.579 | 86.454 | 86.376 | 87.076 | 85.925 | 87.170 | 86.905 | 86.283 | 86.921 |

Table 14: Test accuracies for models with a memory consumption below 32 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model       | CA   | COMP | DREP | ET   | ET-LR | GB   | IC   | IE   | LMD  | RE   |
|-------------|------|------|------|------|-------|------|------|------|------|------|
| adult       | 83.855 | 84.282 | 84.405 | 83.815 | 84.847 | 86.616 | 84.534 | 85.059 | 83.876 | 85.166 |
| anura       | 94.329 | 96.331 | 95.835 | 95.691 | 97.276 | 95.024 | 96.261 | 96.275 | 95.817 | 96.178 |
| avila       | 78.157 | 92.462 | 89.112 | 74.395 | 82.298 | 90.031 | 91.011 | 91.341 | 87.799 | 92.687 |
| bank        | 89.516 | 89.949 | 89.587 | 89.491 | 90.367 | 90.677 | 90.117 | 90.115 | 89.593 | 90.124 |
| chess       | 44.978 | 49.615 | 46.749 | 44.956 | 54.398 | 34.969 | 47.754 | 48.777 | 46.838 | 49.832 |
| connect     | 71.813 | 73.009 | 72.894 | 71.784 | 72.212 | 70.847 | 72.990 | 73.568 | 71.580 | 73.822 |
| eeg         | 78.184 | 79.386 | 79.586 | 78.698 | 82.684 | 82.463 | 80.247 | 81.348 | 77.724 | 81.469 |
| eeg         | 76.033 | 78.156 | 79.348 | 76.740 | 79.707 | 83.878 | 78.889 | 79.928 | 76.382 | 80.577 |
| eeg         | 98.925 | 98.900 | 98.881 | 98.856 | 99.031 | 99.094 | 98.962 | 98.894 | 98.862 | 98.969 |
| japanes-vowels | 85.915 | 89.760 | 87.939 | 88.033 | 93.254 | 87.341 | 89.158 | 88.264 | 88.465 | 90.132 |
| magic       | 84.300 | 84.878 | 84.657 | 84.368 | 86.198 | 87.355 | 85.210 | 85.115 | 84.074 | 85.641 |
| mnist       | 80.500 | 82.300 | 82.300 | 82.300 | 84.900 | 74.000 | 82.000 | 81.300 | 79.700 | 84.700 |
| mozilla     | 92.030 | 92.203 | 92.950 | 91.682 | 93.451 | 94.590 | 93.985 | 93.786 | 89.894 | 93.734 |
| nomao       | 94.876 | 95.244 | 95.111 | 94.957 | 95.691 | 95.935 | 95.361 | 95.213 | 94.917 | 95.358 |
| postures    | 70.184 | 72.281 | 70.851 | 69.851 | 76.214 | 69.058 | 71.476 | 71.861 | 68.168 | 73.158 |
| satimage    | 87.449 | 87.589 | 87.247 | 86.983 | 88.834 | 87.449 | 87.854 | 87.932 | 87.216 | 88.056 |

Table 15: Test accuracies for models with a memory consumption below 64 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.
Table 16: Test accuracies for models with a memory consumption below 128 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model      | CA     | COMP   | DREP   | ET     | ET-LR   | GB     | IC     | IE     | LMD    | RE     |
|------------|--------|--------|--------|--------|---------|--------|--------|--------|--------|--------|
| adult      | 83.898 | 84.601 | 84.445 | 83.941 | 85.369  | **87.135** | 84.712 | 85.188 | 84.085 | 85.172 |
| anura      | 95.344 | 96.942 | 96.692 | 95.267 | 97.726  | 97.197 | 96.331 | 96.915 |
| avila      | 83.323 | 96.454 | 93.051 | 82.206 | 88.796  | 75.262 | **96.454** | 80.817 | 96.454 |
| bank       | 89.516 | 89.960 | 89.633 | 89.589 | 90.595  | **90.737** | 90.117 | 90.115 | 90.124 |
| chess      | 50.121 | 53.393 | 50.866 | 48.695 | 57.973  | 40.719 | 52.481 | 50.845 | 53.532 |
| connect    | 72.413 | 73.927 | 73.314 | 72.429 | **78.861** | 73.850 | 74.080 | 74.697 | 74.795 |
| eeg        | 85.182 | 81.575 | 81.636 | 81.495 | 85.534  | **85.142** | 85.188 | 84.085 | 85.172 |
| elec       | 78.791 | 79.147 | 80.091 | 78.849 | 81.661  | **85.748** | 80.508 | 80.806 | 77.560 | 81.257 |
| ida2016    | 89.925 | 89.980 | 89.894 | 99.131 | **99.169** | 98.994 | 98.862 | 98.975 |
| japanese-vowels | 88.636 | 91.908 | 90.563 | 90.533 | **94.699** | 91.748 | 91.878 | 91.065 | 91.979 |
| magic      | 84.773 | 85.383 | 84.952 | 84.621 | 86.939  | **87.733** | 85.798 | 85.704 | 84.952 | 85.862 |
| mnist      | 83.800 | 85.800 | 85.600 | 85.000 | **89.200** | 79.300 | 86.100 | 84.800 | 83.300 |
| mozilla    | 92.499 | 92.924 | 93.483 | 92.570 | 94.017  | **94.989** | 94.107 | 94.024 | 91.547 | 93.747 |
| nomao      | 95.462 | 95.593 | 95.453 | 95.419 | 95.851  | **96.408** | 95.654 | 95.688 | 95.337 | 95.656 |
| postures   | 75.577 | 77.789 | 76.502 | 75.981 | 82.953  | 75.760 | 77.215 | 77.465 | 73.751 | 77.983 |
| satimage   | 87.854 | 88.445 | 88.320 | 88.090 | **89.736** | 88.631 | 88.771 | 88.460 | 87.776 | 88.694 |

Table 17: Test accuracies for models with a memory consumption below 256 KB for each method and each dataset averaged over a 5 fold cross validation. Rounded to the third decimal digit. Larger is better. The best method is depicted in bold.

| model      | CA     | COMP   | DREP   | ET     | ET-LR   | GB     | IC     | IE     | LMD    | RE     |
|------------|--------|--------|--------|--------|---------|--------|--------|--------|--------|--------|
| adult      | 84.137 | 84.896 | 84.445 | 83.285 | 85.719  | **87.135** | 85.142 | 85.188 | 84.187 | 85.212 |
| anura      | 95.344 | 97.248 | 97.109 | 97.192 | **97.707** | 97.429 | 97.582 | 97.345 | 96.942 | 97.429 |
| avila      | 85.925 | 98.888 | 93.051 | 89.869 | 94.393  | 82.820 | **98.922** | 98.840 | 85.527 |
| bank       | 89.644 | 89.965 | 89.695 | 89.589 | 90.668  | **90.737** | 90.117 | 90.115 | 89.783 | 90.173 |
| chess      | 52.253 | 56.708 | 54.031 | 52.956 | **63.131** | 44.678 | 55.935 | 54.965 | 55.118 | 56.836 |
| connect    | 73.230 | 75.254 | 74.194 | 73.500 | **80.553** | 76.170 | 75.307 | 75.588 | 73.945 | 75.563 |
| eeg        | 82.570 | 84.092 | 83.772 | 84.119 | **88.144** | 85.734 | 84.900 | 84.967 | 82.924 | 85.527 |
| elec       | 79.586 | 80.224 | 81.140 | 79.904 | 82.861  | **85.748** | 82.223 | 81.861 | 79.065 | 82.523 |
| ida2016    | 98.925 | 99.031 | 99.006 | 98.950 | 99.138  | **99.169** | 99.025 | 99.062 | 98.887 | 99.012 |
| japanese-vowels | 89.720 | 93.836 | 92.701 | 92.902 | **95.894** | 94.860 | 93.725 | 92.892 | 93.123 | 93.816 |
| magic      | 85.015 | 85.972 | 85.646 | 85.404 | 87.081  | **87.733** | 86.230 | 86.009 | 85.520 | 86.266 |
| mnist      | 83.800 | 88.600 | 87.600 | 88.300 | **91.200** | 84.600 | 88.200 | 89.000 | 88.600 |
| mozilla    | 93.284 | 94.037 | 93.483 | 93.097 | 94.114  | **94.989** | 94.172 | 94.037 | 92.576 | 94.088 |
| nomao      | 95.668 | 95.886 | 95.825 | 95.651 | 96.240  | **96.408** | 95.871 | 95.860 | 95.665 | 95.900 |
| postures   | 80.599 | 82.708 | 81.608 | 80.563 | **87.939** | 81.100 | 82.304 | 82.918 | 79.493 | 83.077 |
| satimage   | 88.445 | 89.596 | 88.880 | 88.694 | **90.140** | 89.782 | 89.176 | 89.393 | 88.647 | 89.316 |
7.2 Accuracies under various resource constraints with a ExtraTrees Classifier

7.3 Area Under the Pareto Front with ExtaTrees Classifier

| model dataset | AUC | CA  | COMP | DREP | ET  | ET-LR | GB  | IC  | IE  | LMD | RE  |
|---------------|-----|-----|------|------|-----|-------|-----|-----|-----|-----|-----|
| adult         | 0.8451 | 0.8501 | 0.8477 | 0.8474 | 0.8567 | 0.8705 | 0.8511 | 0.8527 | 0.8477 | 0.8517 |
| anura         | 0.9686 | 0.9789 | 0.9788 | 0.9786 | 0.9811 | 0.9794 | 0.9797 | 0.9793 | 0.9782 | 0.9787 |
| avila         | 0.9210 | 0.9929 | 0.9501 | 0.9408 | 0.9852 | 0.9909 | 0.9936 | 0.9926 | 0.8991 | 0.9926 |
| bank          | 0.8965 | 0.9013 | 0.8972 | 0.8972 | 0.9064 | 0.9065 | 0.9017 | 0.9022 | 0.8982 | 0.9018 |
| chess         | 0.5722 | 0.6232 | 0.5936 | 0.5990 | 0.6779 | 0.6265 | 0.6235 | 0.6044 | 0.6128 | 0.6267 |
| connect       | 0.7406 | 0.7573 | 0.7451 | 0.7426 | 0.8234 | 0.7777 | 0.7608 | 0.7600 | 0.7441 | 0.7617 |
| eeg           | 0.8759 | 0.9021 | 0.8985 | 0.8976 | 0.9228 | 0.8563 | 0.9042 | 0.9023 | 0.8964 | 0.9027 |
| elec          | 0.8229 | 0.8316 | 0.8288 | 0.8278 | 0.8577 | 0.8566 | 0.8340 | 0.8304 | 0.8276 | 0.8369 |
| ida2016       | 0.9897 | 0.9905 | 0.9906 | 0.9906 | 0.9911 | 0.9903 | 0.9906 | 0.9905 | 0.9904 | 0.9907 |
| japanese-vowels | 0.9556 | 0.9717 | 0.9701 | 0.9704 | 0.9750 | 0.9743 | 0.9719 | 0.9704 | 0.9704 | 0.9711 |
| magic         | 0.8627 | 0.8693 | 0.8683 | 0.8668 | 0.8751 | 0.8765 | 0.8706 | 0.8692 | 0.8679 | 0.8688 |
| mnist         | 0.9235 | 0.9306 | 0.9311 | 0.9313 | 0.9385 | 0.9385 | 0.9334 | 0.9343 | 0.9296 | 0.9327 |
| mozilla       | 0.9361 | 0.9461 | 0.9428 | 0.9420 | 0.9471 | 0.9491 | 0.9476 | 0.9473 | 0.9404 | 0.9461 |
| nomao         | 0.9609 | 0.9645 | 0.9641 | 0.9638 | 0.9663 | 0.9632 | 0.9648 | 0.9644 | 0.9642 | 0.9647 |
| postures      | 0.8951 | 0.9152 | 0.9049 | 0.9085 | 0.9619 | 0.9104 | 0.9149 | 0.9117 | 0.9102 | 0.9179 |
| satimage      | 0.9012 | 0.9087 | 0.9071 | 0.9090 | 0.9093 | 0.9123 | 0.9102 | 0.9097 | 0.9079 | 0.9082 |