Blossom: an Anytime Algorithm for Computing Optimal Decision Trees

Emir Demirović [*1] **Emmanuel Hebrard** [*2] **Louis Jean** [*3]

Abstract

We propose a simple algorithm to learn optimal decision trees of bounded depth. This algorithm is essentially an anytime version of the state-of-the-art dynamic programming approach. It has virtually no overhead compared to heuristic methods and is comparable to the best exact methods to prove optimality on most data sets. Experiments show that whereas existing exact methods hardly scale to deep trees, this algorithm learns trees comparable to standard heuristics without computational overhead, and can significantly improve their accuracy when given more computation time, even for deep trees.

1. Introduction

In their paper proving that computing decision trees of maximum accuracy is NP-hard, Hyafil and Rivest write: "it is to be expected that good heuristics for constructing near-optimal binary decision trees will be the best solution to this problem in the near future." (Laurent & Rivest, 1976).[1] Indeed, heuristics such as CART (Breiman et al., 1984), ID3 (Quinlan, 1986) or C4.5 (Quinlan, 1993) have been prevalent long afterward, and are still vastly more commonly used in practice than exact methods. We propose a new algorithm (Blossom) which, while being effective at proving optimality, has no computational nor memory overhead compared to heuristics.

There has been some experiments indicating that optimal trees (for some combination of accuracy, depth and size) generalize better to unseen data (Bertsimas & Dunn, 2017). However, these experiments were of limited scope because of the poor scalability of exact algorithms. Our experiments confirm a significant gain in test accuracy, for the objective criterion considered in this paper and in (Bertsimas & Dunn, 2017; Bertsimas & Shioda, 2007; Demirović et al., 2022; Hu et al., 2020; Nijssen & Fromont, 2007): maximizing the training accuracy given an upper bound on the depth. Other criteria have been considered. For instance, OSDT (Hu et al., 2019), GOSDT (Lin et al., 2020) optimize a linear combination of accuracy and number of leaves. However maximizing the accuracy under a constrained depth has is easier to tackle algorithmically, and the predictions of shallower trees are easier to explain.

Despite these desirable features, exact methods have not yet been widely adopted because they do not scale. Exact methods made significant progress lately, and the most recent approaches show very promising results. However, no exact method can replace heuristics in all contexts. For SAT (Avellaneda, 2020; Narodytska et al., 2018) and Integer Programming (Aghaei et al., 2020; Bertsimas & Dunn, 2017; Bertsimas & Shioda, 2007; Verwer & Zhang, 2019), the size of the encoding is a first hurdle. These models require a number of variables proportional to the size of the tree and to the number of datapoints and hence scaling beyond a few thousands datapoints is hard. On the other hand, dynamic programming algorithms DL8 (Nijssen & Fromont, 2007) and DL8.5 (Aglin et al., 2020) scale very well to large data sets. Moreover, these algorithms leverage branch independence: sibling subtrees can be optimized independently, which has a significant impact on the computational complexity. However, DL8.5 tends to be memory hungry and is not anytime. The constraint programming approach of Verhaeghe *et al.* emulates these positive features using propagation algorithms and search strategies (Verhaeghe et al., 2020), while being potentially anytime, although it does not quite match DL8.5's efficiency. Recently, some approaches aiming at improving the anytime aspect have been proposed. For instance, LDS-DL8.5 (Kiossou et al., 2022) uses DL8.5 within a limited discrepancy search method (Harvey & Ginsberg, 1995) thus prioritizing solutions with small deviation from an initial heuristic tree. Similarly to the dynamic programming approaches, the algorithm Murtree (Demirović et al., 2022) optimizes one subtree before moving to the other. However, this choice is dynamic. Moreover, it improves on these approaches in several ways. In particular, it optimizes with respect to

[*] Equal contribution [1] TU DELFT, The Netherlands [2] LAAS-CNRS, Université de Toulouse, CNRS, France [3] Jolibrain, Toulouse, France. Correspondence to: Emir Demirović <e.demirovic@tudelft.nl>, Emmanuel Hebrard <hebrard@laas.fr>, Louis Jean <louis.jean@jolibrain.com>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

[1] Their proof does not consider the exact same problem as in the present paper, however it does apply with trivial changes.

combinations of size and accuracy and implements several methods dedicated to exploring the whole search space efficiently: delaying feature frequency count to a specialized algorithm for subtree of depth two, and implementing an efficient recomputation method for the classification error. As a result, both `Murtree` and `LDS-DL8.5` have better anytime behavior than previous exact methods. However, experimental results show that for deeper trees `Blossom` can significantly outperform all of these methods, while being more memory efficient than approaches that rely on caching the results of subtree computation.

In a nutshell, `Blossom` emulates the dynamic programming approach (see Algorithm 1) implemented for instance in `DL8` (Nijssen & Fromont, 2007) and `DL8.5` (Aglin et al., 2020), while always expanding non-terminal branches (a.k.a "blossoms") before optimizing grown branches. As a result, this algorithm is in a sense strictly better than both the standard dynamic programming approach (because it is anytime and at least as fast) and than classic heuristics (because it emulates them during search, without significant overhead). Our experiments show that it outperforms the state of the art, to the exception of `Murtree` on relatively shallow trees (typically for depths up to 5), for which its more sophisticated (albeit more complex) algorithmic features can pay off.

2. Preliminaries

A data set on a binary feature set $\mathcal{F}$ is a pair $\langle \mathcal{N}, \mathcal{P} \rangle$ where $\mathcal{N}$ and $\mathcal{P}$ are subsets of the feature space $\prod_{f \in \mathcal{F}} \{f, \bar{f}\}$, and are standing, respectively, for negative and positive datapoints, with $\mathcal{D} = \mathcal{N} \cup \mathcal{P}$ being the set containing all datapoints. Given a set of data points $\mathcal{S}$ and a feature f, $\mathcal{S}(f)$ stands for the subset $\{x \mid x \in \mathcal{S}, f \in x\}$ and $\mathcal{S}(\bar{f})$ stands for the subset $\{x \mid x \in \mathcal{S}, \bar{f} \in x\}$.

A *binary decision tree* is a tree whose internal vertices are labelled with features and the two edges exiting a node labelled with f are respectively labelled with the feature f and its negation $\bar{f}$. To a *branch* of a decision tree we associate the ordered set of labels on its edges, from root to leaf. E.g., the leftmost branch in the decision tree of Figure 2 is $(r, a, \bar{h})$. Given a branch b, and a set of data points $\mathcal{S}$ we write $\mathcal{S}(b)$ for $\bigcap_{f \in b} \mathcal{S}(f)$ and we write $b \oplus f$ as a shortcut for $b \cup \{f\}$. The classification error for branch b is $error(b) = \min(|\mathcal{N}(b)|, |\mathcal{P}(b)|)$.

Given a binary data set $\langle \mathcal{N}, \mathcal{P} \rangle$ on the features $\mathcal{F}$, the *minimum error bounded depth decision tree problem* consists of finding a binary decision tree with vertex labels in $\mathcal{F}$ whose branches have cardinality at most k and the sum of the classification error of its maximal branches is minimum. We say that a branch b of a tree is *terminal* if and only if its length $|b|$ is k or if $error(b)$ is null, and we say that a non-terminal branch is a *bud*.

2.1. Dynamic Programming Algorithm

The dynamic programming approach for the minimum error bounded depth decision tree problem relies on the independence of the two subproblems under a given feature test. Algorithm 1 gives a simplified view of this idea implemented e.g. in (Nijssen & Fromont, 2007; Aglin et al., 2020; Demirović et al., 2022).

Algorithm 1: Dynamic Programming Algorithm

```
Algorithm: DP
Data: $\mathcal{N}, \mathcal{P}, \mathcal{F}, k$
Result: Min. error of $k$-bounded trees on data set $\langle \mathcal{N}, \mathcal{P} \rangle$
$error \leftarrow \min(|\mathcal{N}|, |\mathcal{P}|)$
if $k > 0$ and $error > 0$ then
    foreach $f \in \mathcal{F}$ do
        $error \leftarrow \min(error, \mathtt{DP}(\mathcal{N}(f), \mathcal{P}(f), \mathcal{F} \setminus f, k - 1) + \mathtt{DP}(\mathcal{N}(\bar{f}), \mathcal{P}(\bar{f}), \mathcal{F} \setminus f, k - 1))$
return $error$
```

Let $n = |\mathcal{P}| + |\mathcal{N}|$, $m = |\mathcal{F}|$, and let k be the maximum depth. We can safely assume $k \leq m$ (otherwise all features can be tested on every branch) and $k \leq \log n$ (otherwise we could have one distinct branch per datapoint), and it is often assumed that $k \ll m$ and $k \ll \log n$.

Algorithm 1 explores, in the worst case, 2^k branches for each of the number of permutations of k features in $\mathcal{F}$ ($P(m, k) = m!/(m - k)!$) for a total of $\Theta(P(m, k)2^k)$ recursive calls. Note that this is a significant improvement with respect to the $\Theta(m^{2^k})$ trees (with redundant branches) explored by a brute-force algorithm.

Moreover, at each call, data points must be split into two subsets. This takes linear time. However, consider an ordered set of k features and the 2^{k-1} branches testing these features in that order. Their corresponding data sets form a partition of the original data set. Therefore, the k-th split can done in $\Theta(n)$ time amortized over these 2^{k-1} splits since each data points is considered only once, in one of these branches. It follows that the overall time complexity for splitting the data set is in $\Theta(P(m, k)n)$.

Algorithm 1 therefore runs in $\Theta((n + 2^k)P(m, k)) \subset O((n + 2^k)m^k)$ time. Moreover, with the above assumptions on n, m and k, $O(nm^k)$ is a good approximation of its worst-case time complexity.

3. An Anytime Algorithm

Algorithm 2 shows the pseudo-code of an iterative, anytime, version of Algorithm 1 (highlighted code should be ignored for now, it corresponds to improvements detailed in Section 4). This algorithm explores the same search space as Algorithm 1: the same branch is never explored twice. However, incomplete branches are expanded before trying alternative features for already explored branches. In other words, instead of optimizing the left subtree before exploring the right subtree as in Algorithm 1, Algorithm 2 first fully expands a decision tree before exploring alternatives for any branch. For instance, in Figure 1 DP explores branch (a, c) third, whereas Blossom explores $(\bar{a}, b)$ third, and (a, c) only in seventh. The first full decision tree has the branches $(a, b), (a, \bar{b}), (\bar{a}, b)$ and $(\bar{a}, \bar{b})$. This is the fourth leave for Blossom, but the sixth for DP.

Let $f_i <_b f_j$ if and only if feature f_i is selected before feature f_j when expanding branch b at Line 4. Algorithm 2 has the following invariants, from which a formal proof of correctness easily follows:

- If a branch b has been added to $\mathcal{S}$ but is not currently in $\mathcal{S}$, then the *optimal* subtree below this branch has been found and $best(b)$ is its classification error.
- If $(b, f) \in \mathcal{S}$, then for every feature $f' <_b f$, the optimal subtree of b starting with f' has been found. The set $dom(b)$ contains all *untried* feature tests for branch b ($dom(b) = \{f' \mid f' \in \mathcal{F} \wedge f <_b f'\}$).
- If $(b, f) \in \mathcal{S}$ but neither $b \oplus f$ nor $b \oplus \bar{f}$ are in $\mathcal{S}$, b' is *terminal*, or b' is a *bud* yet to be expanded, and $b' \in \mathcal{B}$.

As long as there is a bud ($\mathcal{B} \neq \emptyset$), we pick any one $b \in \mathcal{B}$ at Line 2 and check if it can or need to be expanded in Line 3. If so, we pick a feature f marked as *untried* for b, unmark it, expand the tree with the test f at branch b. The two children $b \oplus f$ and $b \oplus \bar{f}$ can then be added to $\mathcal{B}$.

If there is no bud ($\mathcal{B} = \emptyset$), then the current tree is complete: every maximal branch b is either terminal or optimal. In that case we pop the last assignment (b, f) from $\mathcal{S}$ and update the best error of its subtrees. If there is at least one untried feature for branch b, we add b to $\mathcal{B}$. Otherwise, it is optimal, and $best(b)$ contains the minimum error for any subtree of branch b. This branch will never be expanded anymore since it is not added to $\mathcal{B}$. When the algorithm ends, $best(\emptyset)$ contains the minimum error of any decision tree of depth k.

To simplify the pseudo-code, we use branches to index array-like data structures in Algorithm 2 (e.g. $dom(b)$). Actually, a set of *indices* (at most 2^k in the worst case) are used as proxy for branches in all contexts, since the current tree cannot have more than 2^k branches. At Line 9, the indices for $b \oplus f$, $b \oplus \bar{f}$ are released, and a free index is marked as used when expanding a branch at Line 5. Moreover, the pseudo-code in Algorithm 2 does not show how the best subtrees of optimal branches are recorded, nor how the overall best error is updated when completing a new decision tree at Line 6.

Algorithm 2: Blossom Algorithm

```
Algorithm: Blossom
Data: $\mathcal{N}, \mathcal{P}, k$
Result: Min. error of $k$-bounded trees on data set $\langle \mathcal{N}, \mathcal{P} \rangle$
$\mathcal{S} \leftarrow []$
$\mathcal{B} \leftarrow$ GrowBud($\emptyset, \emptyset$)
while $|\mathcal{S}| + |\mathcal{B}| > 0$ do
1      if $\mathcal{B} \neq \emptyset$ then
2          pick and remove $b$ from $\mathcal{B}$
3          if $|b| < k$ and $best(b) > 0$ then
4              pick and remove $f$ from $dom(b)$
               push $(b, f)$ on $\mathcal{S}$
5              foreach $v \in \{f, \bar{f}\}$ do
                   $\mathcal{B} \leftarrow$ GrowBud($\mathcal{B}, b \oplus v$)
6      else
7          while $|\mathcal{S}| > 0$ do
8              pop $(b, f)$ from $\mathcal{S}$
9              $best(b) \leftarrow$
                 $\min(best(b), best(b \oplus f) + best(b \oplus \bar{f}))$
10             if $dom(b) \neq \emptyset$ and $|b| < k$ then
11                 if $\forall b' \in \mathcal{A}(b), lb(b', b) < best(b')$ then
12                     $\mathcal{B} \leftarrow \mathcal{B} \cup \{b\}$
                       break
return $best(\emptyset)$
Procedure GrowBud($\mathcal{B}, b$)
13     compute $\mathcal{N}(b), \mathcal{P}(b)$, p($f$,$\mathcal{N}(b)$), p($f$,$\mathcal{P}(b)$), $\forall f \in \mathcal{F}$
14     $dom(b) \leftarrow \mathcal{F} \setminus \{f \mid f \in b \vee \bar{f} \in b\}$ sorted by score
       $best(b) \leftarrow \min(|\mathcal{N}(b)|, |\mathcal{P}(b)|)$
       return $\mathcal{B} \cup \{b\}$
```

Theorem 3.1. *The worst case time complexity of Algorithm 2 is in $\Theta((n + 2^k)P(m, k)) \subset O((n + 2^k)m^k)$ and its worst case space complexity is in $\Theta(2^k m)$.*

Proof. The invariants show that Algorithm 2 explores the same set of $P(m, k)2^k$ branches (i.e., the 2^k outcomes of each permutation of k features). Moreover, the "yes" branch of Condition 1 dominates the time complexity since at most one element is added to $\mathcal{S}$, whereas Loop 7 suppresses exactly one element of $\mathcal{S}$ at every (constant time) iteration.

The complexity is thus dominated by the splitting procedure whereby $\mathcal{N}(b \oplus f), \mathcal{N}(b \oplus \bar{f}), \mathcal{P}(b \oplus f)$ and $\mathcal{P}(b \oplus \bar{f})$ are computed from $\mathcal{N}(b)$ and $\mathcal{P}(b)$. As discussed earlier, this takes linear time amortized over the 2^k branches sharing the same set of k features. Therefore, the overall time complexity for the splitting operations is in $\Theta(P(m, k)n)$.

A branch can be stored in constant space via indexing, so the worst case space complexity is in $\Theta(2^k m)$, to record in the sets dom which features have been tried. □

The key difference between Algorithms 1 and 2 is the or-

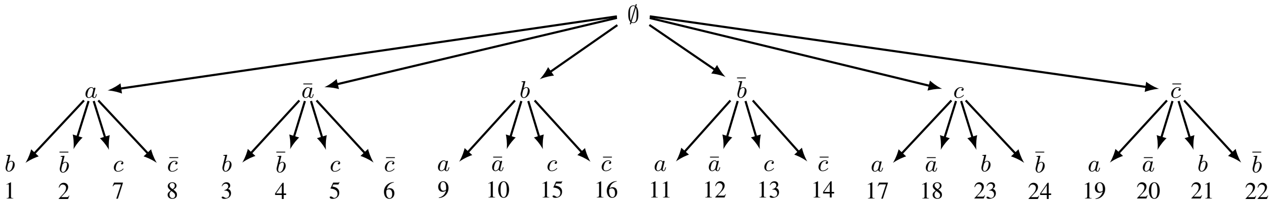


Figure 1: The search tree order: left to right for Algorithm 1 and in the order indicated below for Algorithm 2

der in which branches are explored (see Figure 1). In particular, DP must complete the first recursive call before finding a full tree. Therefore, it finds a first full tree in $\Theta((n+2^k)P(m-1,k-1))$, that is $O(n(m-1)^{k-1})$ time. On the other hand, Blossom finds a first tree in linear time: $\Theta(2^k + nk) = \Theta(nk)$. Another difference with actual implementations of Algorithm 1 (DL8, DL8.5 or Murtree) is that the latter methods use a cache structure in order to reduce the number of branches that need to be explored. Our experimental evaluations, however, show that the overhead of cache lookups may not always be beneficial. Moreover, Algorithm 2 is essentially memoryless. Indeed, under the standard assumption that $2^k \leq n$, its worst-case space complexity is not greater than the size of the input.

4. Extensions of the Algorithm

The pseudo-code given in Algorithm 2 shows the basic structure of the algorithm. Some important (but rather tedious) parts of the algorithm have been omitted, such as how the best subtrees are stored in Line 9 when the best classification score is updated. We discuss here some additional features that have an impact on the efficiency of the algorithm.

4.1. Heuristic Ordering

Testing first the most promising features helps finding accurate trees faster. We used three scores to minimize: *classification error* ($error(b \oplus f) + error(b \oplus \bar{f})$), *entropy* (Quinlan, 1986), and *Gini impurity* (Breiman et al., 1984).

In Line 4, the features are explored in non-decreasing order of the chosen score. All scores are based on the frequencies $p(f, \mathcal{N}(b))$ and $p(f, \mathcal{P}(b))$ of each feature f in each class. These scores are computed by the highlighted instruction in Line 13, and used for sorting in Line 14. Gini impurity was the best in our experiments, and all reported results are using it unless stated otherwise. For branches of length $k-1$ we use classification error instead. Indeed, the optimal feature f for a terminal branch b is the one minimizing the error and other features can be ignored. This is implemented by the highlighted code at Line 10: since the first feature choice cannot be improved, branches of length $k-1$ do not have to be put back into $\mathcal{B}$, and can be backtracked upon.

Computing the frequencies of all features can be done in $\Theta(mn)$ time where $n = |\mathcal{N}(b)| + |\mathcal{P}(b)|$, while $p(\bar{f}, \mathcal{N}(b))$ and $p(\bar{f}, \mathcal{P}(b))$ can be obtained by taking the complement to $|\mathcal{N}(b)|$ and $|\mathcal{P}(b)|$, respectively. In other words, this is more expensive than the splitting procedure by a factor m, but can be similarly amortized. However, since the depth of the branches is effectively reduced by one, the number of terminal branches is reduced by the same factor m, hence this incurs no asymptotic increase in complexity. Furthermore, ordering the features (at Line 14) costs $\Theta(m \log m)$ for each of the $2^{k-1}m^{k-1}$ branches added to $\mathcal{B}$ at Line 5. Again, there is no need to branch on the k-th test of a branch, and the complexity is $O((n + 2^k \log m)m^k)$. This slight increase is inconsequential when n is the dominating term.

The feature ordering has a significant impact on how quickly the algorithm can improve the accuracy of the classifier. Moreover, it also has an indirect, and less significant, impact on the time necessary to prove optimality, because of the lower bound technique detailed in the next section.

4.2. Lower Bound

One can fail early thanks to a lower bound on the error given prior decisions, similarly as DL8.5 does (Nijssen & Fromont, 2007). When some subtrees along a branch b are optimal and their total error is larger than the current upper bound, then there is no point in exploring branch b further.

Observe that $best(b)$ is an upper bound on the classification error for any subtree rooted at b (witnessed by a tree when $best(b)$ is updated). We can propagate this bound to parent nodes efficiently (in $O(|b|)$ time). We assume that this is done recursively for the parent branch, every time the value $best(b)$ is updated. Now, when the condition in Line 10 fails for a branch b, it means that $best(b)$ is *optimal*: no subtree rooted at b of maximum depth $k - |b|$ has a classification error lower than $best(b)$. This is true either because every subtree has been explored, or, with the changes described in Section 4.1, because $k - |b| = 1$ and the feature f with least $error(b \oplus f) + error(b \oplus \bar{f})$ has been chosen. Let $optimal(b) = 1$ if the branch is optimal and $optimal(b) = 0$ otherwise. Notice that this is equivalent, for a branch $b \oplus v$ ending on test $v \in \{f, \bar{f}\}$, to checking if $|b| = k-1$, or if $(b, f) \in \mathcal{S}$ but there is no $g \in \mathcal{F}$ such that $(b \oplus v, g) \in \mathcal{S}$.

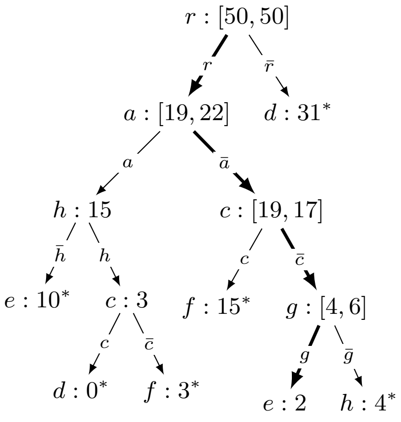


Figure 2: Example of lower bound for a given branch

Moreover, let $\mathcal{A}(b)$ denote the ancestors of branch b in the current tree, i.e., $\mathcal{A}(b) = \{b' \mid (b', v) \in \mathcal{S} \wedge b' \subset b\}$. Now, consider the highlighted code in Line 11. For any ancestor b' of b, we define a lower bound $lb(b', b)$, given the feature tests in $b \setminus b'$ as follows:

$$lb(b', b) = \sum_{b' \subset b'' \oplus f \subseteq b} optimal(b'' \oplus \bar{f}) \cdot best(b'' \oplus \bar{f})$$

That is, $lb(b', b)$ is the total errors of optimal "sibling" branches between b' and b. As long as b belongs to the current tree, these subtrees cannot be improved, hence this lower bound is correct.

Example 1 (Lower bound reasoning). Figure 2 shows a snapshot of the excution of Blossom. Every node is labelled with the feature test on that node, and with the values of $best(b)$ for the branch b ending on that node. When all subtrees of a branch b have been explored (hence $optimal(b) = 1$), this is marked by a "*". We assume that the branch considered at Line 11 is $b = \{r, \bar{a}, \bar{c}, g\}$. For instance, we can suppose that a tree rooted at b with feature e has been found (misclassifying 2 data points). Then, search moved to the sibling branch $\{r, \bar{a}, \bar{c}, \bar{g}\}$, which was then optimized for a total error of 4, and now the pair (b, e) is popped out of $\mathcal{S}$. For all branches b' of b, we give the values of $lb(b', b)$ and $best(b')$ between brackets. Since there exists b' such that $lb(b', b) \geq best(b')$ (e.g., $\emptyset$ and $\{r, \bar{a}\}$), we know that b cannot belong to an improving solution, and hence there is no need to try to extend it further.

This reasoning is more effective when good upper bounds are found early, hence the feature ordering discussed in the previous section has an impact. Moreover, the choice of branch in Line 2 has an impact as well. We found that the simplest branch selection strategy was also the one giving the best results: we expand first the branch that was inserted into $\mathcal{B}$ first (i.e., $\mathcal{B}$ is *FIFO*). One possible explanation is that by avoiding to unnecessarily "jump" to different parts of the decision tree, this strategy promotes optimizing sibling subtrees first, and therefore, deeper tree earlier.

4.3. Preprocessing

Finally, we use two preprocessing techniques, one on the data set and one on the feature set. Although straightforward, they both have a significant impact.

Dataset reduction. It is easy to adapt Blossom to handle weighted data sets by redefining the error as follows, given a weight function w on $\mathcal{D}$: $error(b) = \min\left(\sum_{x \in \mathcal{N}(b)} w(x), \sum_{x \in \mathcal{P}(b)} w(x)\right)$.

We can use weights as an optimization in the context of noisy data, by merging duplicated datapoints and suppressing inconsistent datapoints. Let w^- (resp. w^+) denote the number of occurrences of x in $\mathcal{D}^-$ (resp. $\mathcal{D}^+$). We use the weight function $w(x) = |w^-(x) - w^+(x)|$. Then, for any datapoint x, we remove all but one of its occurrences, in $\mathcal{D}^-$ if $w^- > w^+$, in $\mathcal{D}^+$ if $w^+ > w^-$, and suppress it completely if $w^+ = w^-$. The reported error will then need to be offset by the number of pairs of suppressed inconsistent datapoints, that is: $\sum_{x \in \mathcal{D}} \min(w^-(x), w^+(x))$. Reducing the number of datapoints in the data set has a non-null, although tiny impact on efficiency. However, suppressing inconsistent datapoints is very important. In particular, proving optimality when the minimum error is positive basically requires to exhaust the search space and is therefore extremely costly. On the other hand, when there exists a perfect tree within the maximum depth, we can stop as soon as we find it. This preprocessing allows to benefit from that when we find a tree whose error is equal to the number of pair of inconsistent datapoints in the original data set. This preprocessing can be done in $O(mn \log n)$ by ordering the datapoints in lexicographic order and then processing them in sequence.

Feature reduction. A feature f is redundant if there exists another feature f' such that either: $\forall x \in \mathcal{D}, f \in x \Leftrightarrow f' \in x$, or $\forall x \in \mathcal{D}, f \in x \Leftrightarrow f' \notin x$. They can be found in $O(nm^2)$ time by comparing pairs of rows of the data set.

Removing redundant features may appear naive, however, it turns out that the binarization techniques (one-hot encoding) are often not optimized and many redundant features do exist in standard data sets. The number of features (m) has a huge impact on the complexity: the branching factor of the algorithm is indeed $2m$ (see Figure 1). Moreover, at every branch, "informationless" features (i.e., features f such that $(\forall x \in \mathcal{P}(b) f \in x) \iff (\forall x \in \mathcal{N}(b) f \in x)$) can be suppressed at no additional cost since this can be detected when computing the feature ordering criterion.

k	Blossom			Murtree			Murtree^P			CP			DL8.5				binOCT	
	opt.	acc.	cpu	opt.	acc.	cpu*	opt.	acc.	cpu*	opt.	acc.	cpu*	sol.	opt.	acc.*	cpu*	sol.	acc.*
$m < 100$ (29 data sets)																		
3	1.00	0.8871	0.23	1.00	0.8871	+0.31	1.00	0.8871	+0.22	1.00	0.8871	+3.2	1.00	1.00	-0.00%	+2.5	0.52	-1.21%
4	1.00	0.9130	14	1.00	0.9130	+8.0	1.00	0.9130	+4.0	1.00	0.9130	+115	1.00	1.00	+0.00%	+105	0.52	-2.62%
5	0.93	0.9344	187	0.97	0.9344	-12	0.97	0.9344	-89	0.62	0.9337	+121	1.00	0.86	-0.01%	+2.0	0.52	-3.81%
7	0.66	0.9596	81	0.69	0.9564	+90	0.79	0.9587	-69	0.45	0.9075	+193	1.00	0.84	-0.46%	+6.7	0.52	-7.30%
10	0.79	0.9733	85	0.52	0.9623	+83	0.83	0.9669	-70	0.45	0.8691	+2.6	1.00	0.83	-1.34%	+49	0.41	-29.46%
$m \geq 100$ (29 data sets)																		
3	0.89	0.8874	100	0.89	0.8874	-42	0.89	0.8874	-45	0.75	0.8870	+256	1.00	0.86	-0.25%	+247	0.64	-2.36%
4	0.57	0.9060	662	0.75	0.9053	+64	0.75	0.9052	-266	0.29	0.8930	+576	1.00	0.54	-0.85%	+258	0.64	-4.78%
5	0.36	0.9211	452	0.36	0.9202	+99	0.39	0.9207	-357	0.14	0.8327	+11	1.00	0.33	-1.85%	+12	0.64	-8.55%
7	0.32	0.9449	11	0.32	0.9337	+7.4	0.32	0.9389	-3.4	0.29	0.8170	+571	1.00	0.44	-3.02%	+793	0.57	-22.33%
10	0.46	0.9613	101	0.43	0.9542	+19	0.46	0.9555	-12	0.39	0.8604	+85	1.00	0.67	-2.59%	+183	0.21	-55.08%

Table 1: Comparison with the state of the art: accuracy and optimality proofs

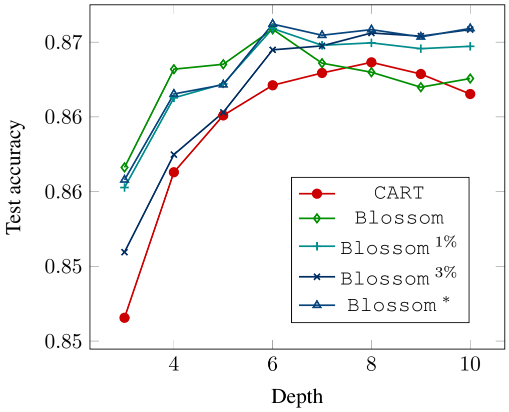


Figure 3: Test accuracy w.r.t. depth (All data sets)

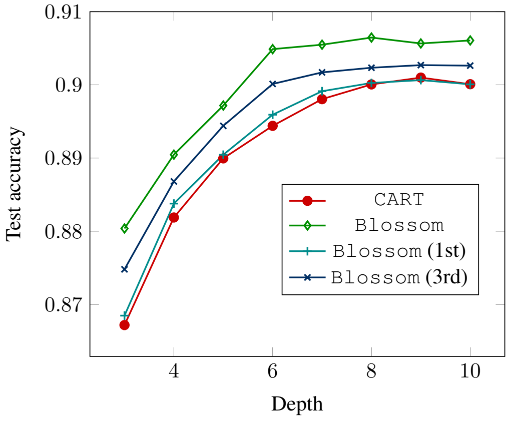


Figure 4: Test accuracy w.r.t. depth (Filtered data sets)

5. Experimental Results

All experiments were run on 4 cluster nodes, each with 36 Intel Xeon CPU E5-2695 v4 2.10GHz cores running Linux Ubuntu 16.04.4. Sources were compiled using g++8.[2] Every algorithm was run until completion or until reaching a time limit of one hour, and within a memory limit of 50GB.

We used a collection of 58 data sets from related work (Aglin et al., 2020; Narodytska et al., 2018; Verwer & Zhang, 2019), plus 8 large extra data sets from Kaggle and UCI. Further details, as well as the raw data of our experiments can be found in the appendix.

5.1. Generalization of optimal decision trees

The capacity of optimized trees to generalize to unseen data has been empirically evaluated and evidence of gains over heuristic trees have been found (Bertsimas & Dunn, 2017; Demirović et al., 2022). We reproduce this type of experiments here, however, because they are rare and often restricted to shallow trees. Therefore we first tried to reproduce those findings.

Figure 3 shows the test accuracy, averaged across all 58 data sets of CART (in red), Blossom (in green) and several *post-pruned* versions of Blossom (in shades of blue). For each data set we generated 50 random samples of 20% of the datapoints to be used for validation, while the rest was used for training, and we report the averages. Given the large number of runs, we stopped Blossom after 10 minutes in each run, and kept the last tree it returned. For shallow trees ($k \leq 6$), Blossom is more accurate on these data sets. The difference is statistically significant for $k = 3$ (99.9% confidence) and $k = 4$ (95%), but with only 50%

[2] The source code for Blossom is available at https://gitlab.laas.fr/ehebrard/blossom

and 60% confidence for $k = 5$ and 6 respectively. For deeper trees there is no statistically significant difference, and CART seems even slightly better.

We believe that this is due to overfitting. In order to test this hypothesis, we report on the same figure the accuracy of Blossom's trees, but pruned using *cost complexity pruning* up until a cumulative loss of accuracy of 1% (Blossom $^{1\%}$) or 3% (Blossom $^{3\%}$).[3] Pruned optimized trees have a better accuracy than CART's even when the depth grows. Moreover, since pruning is more helpful for deeper trees, we also report the results of Blossom pruned with an adaptive limit of $0.3 \cdot k^2\%$. This approach is the best overall, although the confidence level is below 50% for $k = 7$ and 8.

The statistical significance of these results is not entirely satisfying. Moreover, the difference is very much dependent on the data set.[4] After a careful analysis of the experiment data, we observed that in some data sets, none of the methods is able to learn a meaningful decision tree, in the sense that their test accuracy is very weakly correlated to training accuracy. For instance, suppose we are trying to learn the parity function. There is no way to represent this function with a tree whose depth is not equal to the number of features. We believe that some of these data sets have similar properties. Under this hypothesis, we tried to design a mecanism to filter out such data sets, and came up with the following simple test: *we keep only those data sets where* CART *with* $k = 10$ *is at least as good as the best single-feature test*. This condition excludes 18 of the 58 data sets. Although this selection is by design favorable to CART, we can observe in Figure 4 that Blossom (without post-pruning), generalizes better in average to unseen data on these 40 data sets (the paired t-test gives about 75% confidence that Blossom is more accurate than CART for $k \geq 5$).

Finally, to further test the hypothesis that more search improves test accuracy, we also report the accuracy of the first (Blossom (1st)), or the third (Blossom (3rd)) solution found by Blossom. The former is equivalent to CART, and the latter yields trees with higher accuracy, but not as high as the full version, which supports our hypothesis.[5]

5.2. Computing (optimally) accurate classifiers

We then compare Blossom to state-of-the-art algorithms, Murtree (Demirović et al., 2022) and DL8.5 (Aglin et al., 2020), as well as the best MIP (binOCT) (Verwer & Zhang, 2019) and CP (CP) (Verhaeghe et al., 2020) models, for computing and proving optimal trees. We also ran all these methods on *preprocessed* data sets, using the approach described in Section 4.3 to remove inconsistent data points and redundant features. We only report those results for Murtree (with superscript '^{P}') for lack of space, although all results are in the appendix. Infortunately, at the time of writing, we were not aware of LDS-DL8.5 and did not use it in our experimental evaluation. Since it is similar to DL8.5 in terms of overall performance, we are confident that Blossom is more efficient for deep trees. The anytime behavior of this algorithm is however vastly better than DL8.5's and would need to be compared with Blossom's. We did not compare our algorithm to approaches that directly learn on non-binary data sets (e.g. (Shati et al., 2021; Mazumder et al., 2022)), factoring in the binary encoding makes fair comparisons difficult.

In Table 1 we split the data sets in two classes according to the number of features m. Every method is run with the bound k on the depth shown in the first column. We report the ratio of data sets for which an optimality proof could be derived (opt.); the average training accuracy (acc.); and the average CPU time (cpu) to prove optimality. Since comparing CPU times only makes sense when optimality is proven by both methods, we report the average difference w.r.t. Blossom's on the data sets for which both methods *prove optimality*. Similarly, since DL8.5 and binOCT exceed the memory limit of 50GB in some cases, we compute their accuracy *relative* to Blossom on the subset of runs where at least one tree was found. Moreover, the column (sol.) gives the ratio of such "successful" runs. Columns with results *relative to* Blossom are marked by a "*".

The first observation is that Blossom and Murtree are significantly more efficient than other methods on all criteria. Moreover CP, DL8.5 and binOCT clearly do not scale to high depths, the two latter even often exceeding the memory limit (set to 50GB). With respect to Murtree, the experimental results support two conclusions:

Blossom systematically dominates Murtree for the average training accuracy, especially as the depth and the feature set grow. Since the accuracy results in Table 1 include data sets for which an optimal tree is found, the gap on other data sets is larger. Moreover, they are averaged over 58 data sets, so a gap of a fraction of a point is significant: a careful analysis of the results shows that if the gaps are variable, they are consistently in favor of Blossom. For instance, for $k = 7$ (resp. $k = 10$) the average gain of Blossom w.r.t. Murtree is of 1.6 (resp. 1.1) points, over the 31 (resp. 39) data sets not solved to optimality. This gain is due to Blossom's integration of feature frequency counting to guide search via heuristics toward promising trees (and also reduce the search space) as described in Section 4.

However, Murtree's use of caching makes it better at proving optimality. The raw results are not always in favor of Murtree, especially for high depths. However, there

[3] CART's trees are pruned with default settings

[4] A few data sets account for most of the gap, cf. appendix.

[5] The number of improving solutions found by Blossom in 10 minutes was in average 7.47 for $k = 3$ and up to 31.94 for $k = 10$.

are two types of proofs: "non-trivial" proofs obtained by exhausting the search space, and "trivial" proofs obtained by finding a perfect tree (whose classification error is null, or equal to the number of inconsistent data points when using preprocessing). Caching helps for the former hence Murtree's higher proof ratio for shallower trees. In the case of deeper trees, however, non-trivial proofs are harder to obtain, and the ratio of trivial proofs increases because finding a perfect tree is easier. Since Blossom finds more accurate tree faster, it also obtains more of these proofs. Indeed, Murtree^P (i.e., with our preprocessing) is the best method to prove optimality overall, since preprocessing means that more data sets have a trivial proof.

5.3. Anytime behavior

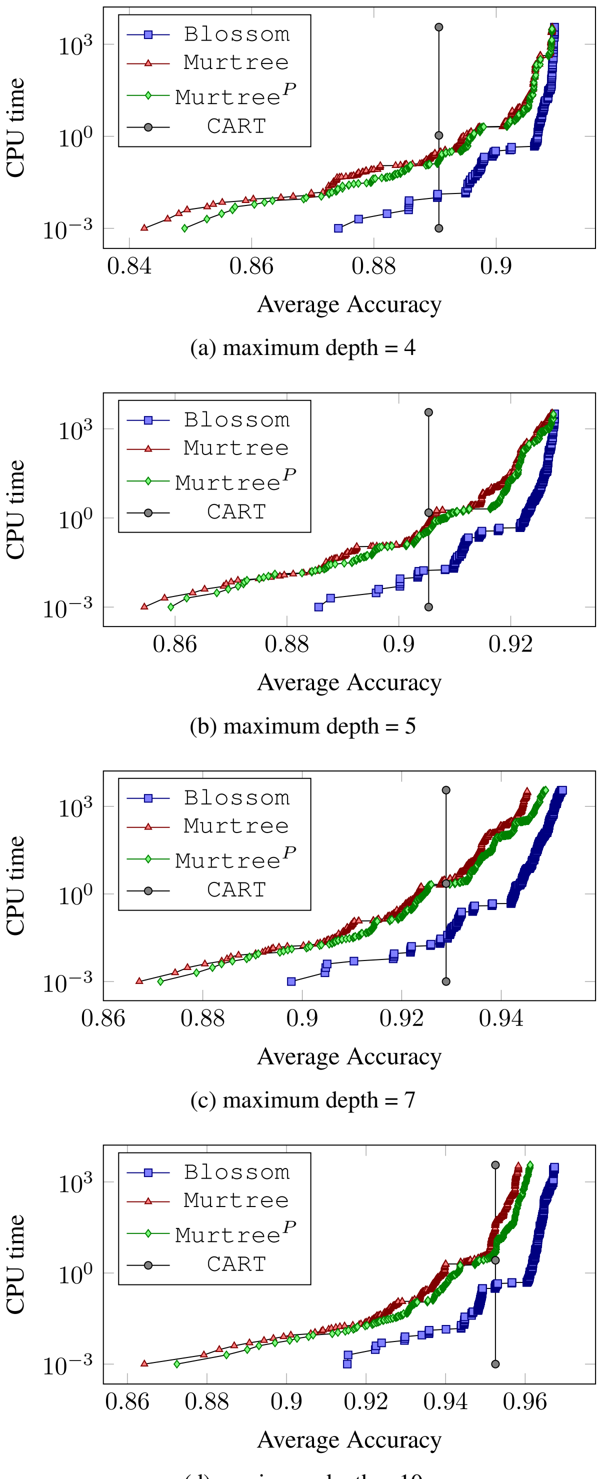
Next, we shift our focus to how fast can we obtain accurate trees and how fast can we improve the accuracy over basic solutions found by heuristics. We use the scikit-learn implementation of CART, which has a parameter to bound the depth of the tree, as baseline. Here we report the average error, over all 58 data sets, after a given period of time (first solution, after 10 seconds, or after 1 minute), both for Murtree and Blossom in Table 3.

Firstly, this experiment shows that the initial solution is very similar to that found by CART. There is actually a slight advantage for Blossom, which can be explained by CART's post-pruning as well as by the fact that whereas CART systematically selects the feature with minimum Gini impurity, Blossom does so for all *but the deepest feature tests*, for which it uses least classification error instead. Secondly, this first tree is found extremely quickly, and there is no scaling issue with respect to the depth of the tree nor to the size of the data set. Thirdly, even for large data sets and deep trees, the accuracy of the initial classifier can be significantly improved given more computation time.

Figure 5 reports the evolution, averaged across all 58 data sets, of the accuracy over time, giving a good view of the difference between Murtree and Blossom during search. The accuracy of the tree returned by CART is given for reference. We can see in those graphs Blossom's anytime accuracy is significantly higher than that of Murtree. Notice that the first few improving solutions reported by Murtree are not complete trees. This is apparent in Table 3 which shows the acurracy at different time points during search. However, this has an impact only on the first milliseconds of search and have no impact on the graphs in the long run.

5.4. Analysis of the extensions

Finally, we report in Table 2 the results of three variants, in order to analyse the impact of the factors described in Section 4. For each variant, we give the average accuracy (acc.), ratio of proofs (opt.) and relative increase in cpu time



(a) maximum depth = 4

(b) maximum depth = 5

(c) maximum depth = 7

(d) maximum depth = 10

Figure 5: Accuracy over time, averaged across all data sets

k	Blossom			No heuristic			No preprocessing			No lower bound		
	acc.	opt.	cpu	acc.	opt.	cpu*	acc.	opt.	cpu*	acc.	opt.	cpu*
3	0.8875	0.93	67	0.8871	0.93	-1.6	0.8875	0.93	+4.3	0.8875	0.93	-2.8
4	0.9095	0.78	388	0.9081	0.79	-61	0.9093	0.72	+47	0.9095	0.78	+13
5	0.9275	0.64	479	0.9257	0.67	-30	0.9272	0.55	+276	0.9275	0.64	+39
7	0.9517	0.48	1045	0.9455	0.48	+17	0.9512	0.43	+163	0.9512	0.48	+30
10	0.9667	0.62	570	0.9583	0.53	+66	0.9656	0.43	+63	0.9665	0.60	+8.0

Table 2: Factor analysis

k	Blossom			Murtree^P			CART
	first	≤10s	≤1m	first	≤10s	≤1m	first
3	795	734	729	2101	895	891	808
4	688	650	642	2101	792	786	694
5	607	575	569	2101	718	710	609
7	499	471	467	2101	566	558	507
10	407	382	374	2101	448	441	412

Table 3: Anytime comparison with state the of the art

(cpu*), on runs where an optimal tree has been found.

The variant "No heuristic" disables the Gini impurity heuristic described in Section 4.1, and replaces it by simply selecting first the feature with minimum error. For shallow trees (depth 3 or 4), since in many cases the search space is completely exhausted, not computing the slightly more costly Gini impurity score may actually be a good choice and we observe run time reductions of about 15% to 20%. However, the accuracy of the trees decreases rapidly for larger maximum depth. As a results, less optimality proofs are obtained, and they take longer to compute.

The variant "No preprocessing" disables the preprocessing described in Section 4.3. The feature ordering is impacted by the removal of datapoints, and hence a more acurate tree might be found without preprocessing. However, in most cases, preprocessing does pay off, yielding more optimality proofs, better accuracy, and shorter runtimes. We estimate that most of the gain is due to the removal of redundant features, and of inconsistent datapoints, whereas the fusion of datapoints accounts for only a slight speed-up.

The variant "No lower bound" disables the bound described in (4.2). We observe a slight increase in CPU time in average (but up to 200% for some data sets) but it only slightly negatively affects the accuracy and the number of proofs.

6. Conclusion

We have introduced a simple, exact, iterative, memory-efficient and anytime algorithm for computing optimally-accurate tree classifiers of bounded depth. This algorithm scales significantly better than state-of-the-art exact algorithms to high depths while having no significant time nor memory overhead with respect to greedy heuristic methods.

Our experiments provide further evidence that trees of bounded depth and optimal training accuracy generalize better to unseen data than heuristic trees. However they also indicate that using the size of tree in the objective is beneficial as well, raising the question of the tradeoff between adequacy and complexity in the objectives and models.

Acknowledgements

This work was supported by the AI Interdisciplinary Institute ANITI, funded by the French program "Investing for the Future – PIA3" under Grant agreement no. ANR-19-PI3A-0004; and by the European Union's Horizon Europe Research and Innovation program under the grant agreement TUPLES No101070149.

References

Aghaei, S., Gomez, A., and Vayanos, P. Learning Optimal Classification Trees: Strong Max-Flow Formulations. *arXiv preprint arXiv:2002.09142*, 2020.

Aglin, G., Nijssen, S., and Schaus, P. Learning Optimal Decision Trees Using Caching Branch-and-Bound Search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 3146–3153, 2020.

Avellaneda, F. Efficient Inference of Optimal Decision Trees. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 3195–3202, 2020.

Bertsimas, D. and Dunn, J. Optimal classification trees. *Machine Learning*, 106(7):1039–1082, 2017.

Bertsimas, D. and Shioda, R. Classification and regression via integer optimization. *Operations Research*, 55(2): 252–271, 2007.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. *Classification and Regression Trees*. Wadsworth & Brooks, 1984.

Demirović, E., Lukina, A., Hebrard, E., Chan, J., Bailey, J., Leckie, C., Ramamohanarao, K., and Stuckey, P. J. MurTree: Optimal Decision Trees via Dynamic Programming and Search. *Journal of Machine Learning Research*, 23(26):1–47, 2022.

Harvey, W. D. and Ginsberg, M. L. Limited Discrepancy Search. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 607—-613, 1995.

Hu, H., Siala, M., Hebrard, E., and Huguet, M. Learning Optimal Decision Trees with MaxSAT and its Integration in AdaBoost. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1170–1176, 2020.

Hu, X., Rudin, C., and Seltzer, M. Optimal Sparse Decision Trees. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 7265–7273, 2019.

Kiossou, H. S., Schaus, P., Nijssen, S., and Houndji, V. R. Time Constrained DL8.5 Using Limited Discrepancy Search. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, pp. 443–459, 2022.

Laurent, H. and Rivest, R. L. Constructing Optimal Binary Decision Trees is NP-complete. *Information Processing Letters*, 5(1):15–17, 1976.

Lin, J., Zhong, C., Hu, D., Rudin, C., and Seltzer, M. Generalized and Scalable Optimal Sparse Decision Trees. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pp. 6150–6160, 2020.

Mazumder, R., Meng, X., and Wang, H. Quant-BnB: A Scalable Branch-and-Bound Method for Optimal Decision Trees with Continuous Features. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, pp. 15255–15277, 2022.

Narodytska, N., Ignatiev, A., Pereira, F., and Marques-Silva, J. Learning Optimal Decision Trees with SAT. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1362–1368, 2018.

Nijssen, S. and Fromont, E. Mining Optimal Decision Trees from Itemset Lattices. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 530–539, 2007.

Quinlan, J. R. Induction of Decision Trees. *Mach. Learn.*, 1 (1):81—-106, 1986.

Quinlan, R. *C4.5: Programs for Machine Learning*. Kaufmann, 1993.

Shati, P., Cohen, E., and McIlraith, S. SAT-Based Approach for Learning Optimal Decision Trees with Non-Binary Features. In *Proceedings of the 27th International Conference on Principles and Practice of Constraint Programming (CP)*, pp. 50–50, 2021.

Verhaeghe, H., Nijssen, S., Pesant, G., Quimper, C.-G., and Schaus, P. Learning Optimal Decision Trees Using Constraint Programming. *Constraints*, 25(3–4):226–250, 2020.

Verwer, S. and Zhang, Y. Learning Optimal Classification Trees using a Binary Linear Program Formulation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 1625–1632, 2019.

A. Appendix

Section A.1 provides information about the chosen data sets. Section A.2, provides the raw data from all our experiments (for every method and every data set).

A.1. Information about the data sets

The benchmark of classification data set we used is described in Table 4. It consists of 50 data sets commonly used in related work articles (namely, (Aglin et al., 2020; Narodytska et al., 2018; Verwer & Zhang, 2019)), to which we added the following large data sets in order to stress how well the different approaches scale.

The taiwan_binarised and adult_discretized data sets come from the UCI repository and were discretized using ad-hoc threshold on continuous or large numerical features, and one-hot encoding for symbolic features.

The compas_discretized, bank, titanic, surgical-deepnet and weather-aus data sets come from Kaggle and were binarized using the one-hot encoding.

The data set mnist_0 is the well known data set on hand written digits binarized as follows: every pixel is a binary attribute whose value is 1 if its greyscale is larger than 0.5 and 0 otherwise. The data point is positive if it is the digit "0" and negative otherwise.

We report the number of data points ($|\mathcal{D}|$), the number of features ($|\mathcal{F}|$), the same parameters after preprocessing (respectively $|\mathcal{D}|^*$ and $|\mathcal{F}|^*$), and the "noise" ratio, that is: $2|\mathcal{P} \cap \mathcal{N}|/(|\mathcal{P}| + |\mathcal{N}|)$.

A.2. Full experimental results

A.2.1. Test Accuracy

We report here the raw data from our experimental evaluation of the test accuracy of Blossom's trees in the following tables:

$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$	$k = 8$	$k = 9$	$k = 10$
Tab. 5	Tab. 6	Tab. 7	Tab. 8	Tab. 9	Tab. 10	Tab. 11	Tab. 12

In Tables 5 to 12, data sets filtered-out according to the condition defined in (5.1) are given between parenthesis.

A.2.2. Comparison with the state of the art

We report here the raw data from our experimental comparison with the state of the art for $k = 3, 4, 5, 7, 10$ and for the four size catagories in the following tables:

m	$k = 3$	$k = 4$	$k = 5$	$k = 7$	$k = 10$
< 100	Table 13	Table 15	Table 17	Table 19	Table 21
≥ 100	Table 16	Table 14	Table 18	Table 20	Table 22

For every instance, we give the classification error of the best tree found within a time limit of 1h for every method. Moreover, we give the CPU time taken by each method to prove optimality when optimality is proven (in which case we mark it by a "*"), and to find the best solution otherwise. Notice that CP and DL8.5 are not anytime and hence only report a solution at the end of the time limit when optimality is not proven. In this case, we write $\geq 1h$.

Every process was first run with a memory limit of 3.5GB. Many runs of DL8.5, CP and binOCT went well over that limit and were rerun with a limit of 50GB. Still, 138 runs of binOCT and 164 runs of DL8.5 (out of 460) went over the limit. As binOCT can output trees anytime, the data for these runs (up until the memory blow-out) are in the tables. For DL8.5, however, this is marked as a "-" since there was no output.

set	$\lvert\mathcal{D}\rvert$	$\lvert\mathcal{F}\rvert$	$\lvert\mathcal{D}\rvert^*$	$\lvert\mathcal{F}\rvert^*$	noise
monk3	122	11	122	11	0.0000
monk1	124	11	124	11	0.0000
hepatitis	137	68	136	34	0.0000
lymph	148	68	148	47	0.0000
iris	150	12	28	10	0.0133
monk2	169	11	169	11	0.0000
heart-cleveland	296	95	296	54	0.0000
primary-tumor	336	31	240	17	0.0893
vote	435	48	342	48	0.0000
balance-scale	625	16	625	16	0.0000
soybean	630	50	502	43	0.0063
breast-cancer	683	89	449	89	0.0000
IndiansDiabetes	768	43	723	40	0.0208
anneal	812	93	495	49	0.0837
tic-tac-toe	958	27	958	27	0.0000
messidor	1151	86	1124	82	0.0104
banknote	1372	28	358	26	0.0029
yeast	1484	89	1418	46	0.0067
wine1	178	1276	178	646	0.0000
wine2	178	1276	178	646	0.0000
wine3	178	1276	178	646	0.0000
audiology	216	148	186	84	0.0000
ionosphere	351	445	350	222	0.0000
forest-fires	517	989	504	656	0.0155
australian-credit	653	125	653	74	0.0000
breast-wisconsin	683	120	449	60	0.0000
diabetes	768	112	768	56	0.0000
vehicle	846	252	846	126	0.0000
titanic	887	333	803	333	0.0361
german-credit	1000	112	998	86	0.0000
biodeg	1055	304	1017	291	0.0019
winequality-red	1599	42	1073	39	0.0025
car	1728	21	1728	21	0.0000
car_evaluation	1728	14	1072	14	0.0926
seismic_bumps	2584	91	1663	87	0.0255
chess	3196	73	3196	39	0.0000
kr-vs-kp	3196	73	3196	38	0.0000
hypothyroid	3247	88	2527	44	0.0105
compas_discretized	6167	25	4181	20	0.5928
HTRU_2	17898	70	12710	70	0.0193
magic04	19020	86	18174	86	0.0083
adult_discretized	30299	59	17804	56	0.2149
segment	2310	235	2027	114	0.0000
splice-1	3190	287	3005	255	0.0006
Statlog_satellite	4435	539	4432	531	0.0000
bank_conv	4521	212	4520	207	0.0000
spambase	4601	386	3294	378	0.0252
pendigits	7494	216	7415	108	0.0000
mushroom	8124	119	8124	100	0.0000
surgical-deepnet	14635	6047	11733	6046	0.0000
letter	20000	224	18200	112	0.0000
default_credit	30000	307	29823	304	0.0025
taiwan_binarised	30000	205	29112	198	0.0253
Statlog_shuttle	43500	746	23352	727	0.0000
bank	45211	9531	45211	9530	0.0000
mnist_0	60000	784	59987	671	0.0000
hand_posture	78095	3864	72896	3860	0.0000
weather-aus	142193	4759	142151	4756	0.0000

Table 4: Benchmark and preprocessing data

		CART	Blossom	Blossom $^{1\%}$	Blossom $^{3\%}$	Blossom *
	single test	test acc.	test acc.	test acc.	test acc.	test acc.
adult_discretized	0.795	0.8095	**0.8346**	0.8286	0.8195	0.8335
anneal	0.808	0.8092	**0.8639**	0.8475	0.8374	0.8535
audiology	0.874	**0.9541**	0.9473	0.9414	0.9350	0.9423
(australian-credit)	0.871	0.8518	**0.8767**	0.8636	0.8585	0.8639
(balance-scale)	0.921	0.9202	**0.9206**	**0.9206**	**0.9206**	**0.9206**
bank	0.893	**0.9002**	0.8997	0.8979	0.8830	0.8987
(bank_conv)	0.891	**0.8972**	0.8949	0.8945	0.8844	0.8958
banknote	0.843	0.9087	0.9732	0.9734	0.9595	**0.9740**
biodeg	0.760	0.8004	**0.8185**	0.8143	0.8068	0.8180
breast-cancer	0.895	0.9381	0.9409	0.9396	0.9374	**0.9415**
breast-wisconsin	0.923	0.9422	**0.9586**	0.9574	0.9501	0.9585
car	0.714	**0.8758**	0.8717	0.8742	0.8723	0.8753
car_evaluation	0.699	0.8687	0.8617	**0.8768**	0.8761	0.8686
(chess)	1.000	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
compas_discretized	0.644	0.6534	0.6701	**0.6717**	0.6643	0.6704
(default_credit)	0.820	0.8208	**0.8214**	0.8193	0.8190	0.8193
(diabetes)	0.744	**0.7492**	0.7348	0.7381	0.7342	0.7418
(forest-fires)	0.540	0.5367	**0.5427**	0.5381	0.5398	0.5390
(german-credit)	0.701	0.7105	**0.7376**	0.7275	0.7271	0.7273
hand_posture	0.794	0.8923	**0.9021**	0.8993	0.8993	0.8993
(heart-cleveland)	0.745	**0.7943**	0.7707	0.7830	0.7883	0.7787
(hepatitis)	0.826	0.7883	**0.7893**	0.7814	0.7807	0.7759
HTRU_2	0.962	0.9764	**0.9766**	0.9752	0.9671	0.9752
hypothyroid	0.962	0.9790	0.9783	0.9784	0.9750	**0.9794**
(IndiansDiabetes)	0.738	0.7497	**0.7584**	0.7453	0.7404	0.7482
ionosphere	0.813	**0.8989**	0.8622	0.8676	0.8704	0.8670
iris	0.937	0.9807	**0.9893**	0.9827	0.9700	0.9847
kr-vs-kp	0.675	0.9040	**0.9382**	0.9362	0.9351	0.9362
letter	0.959	0.9655	0.9808	**0.9816**	0.9593	**0.9816**
letter_recognition	0.961	0.9912	**0.9924**	0.9876	0.9876	0.9917
lymph	0.799	0.7974	0.8000	**0.8213**	0.8155	0.8194
magic04	0.738	0.8009	0.8164	0.8125	0.8025	**0.8168**
messidor	0.591	**0.6576**	0.6479	0.6476	0.6434	0.6468
mnist_0	0.920	0.9423	**0.9558**	0.9504	0.9501	0.9554
monk1	0.733	0.8331	**0.8957**	0.8908	0.8908	0.8908
monk2	0.618	0.5629	**0.6522**	0.6347	0.6471	0.6341
monk3	0.754	0.9152	**0.9292**	0.9136	0.9272	0.9136
mushroom	0.887	0.9652	**0.9991**	0.9974	0.9852	0.9974
pendigits	0.932	**0.9921**	0.9918	0.9884	0.9795	0.9913
(primary-tumor)	0.789	0.7909	**0.8446**	0.8332	0.8362	0.8338
segment	0.983	0.9960	**0.9979**	0.9904	0.9902	0.9975
(seismic_bumps)	0.934	0.9314	0.9297	**0.9342**	**0.9342**	0.9297
soybean	0.850	0.9024	0.9397	0.9323	0.9304	**0.9413**
spambase	0.786	**0.8427**	0.8358	0.8356	0.8255	0.8403
splice-1	0.822	0.9085	**0.9329**	0.9305	0.9117	0.9305
Statlog_satellite	0.757	0.9165	0.9471	**0.9507**	0.9316	**0.9507**
Statlog_shuttle	0.928	0.9985	**0.9999**	0.9987	0.9987	0.9987
surgical-deepnet	0.786	0.8056	**0.8250**	0.8228	0.8099	0.8240
(taiwan_binarised)	0.820	0.8210	**0.8214**	0.8186	0.8159	0.8197
tic-tac-toe	0.698	0.7319	0.6942	**0.7353**	0.7303	0.7252
titanic	0.780	0.8163	**0.8191**	0.8179	0.8071	0.8163
vehicle	0.769	0.9087	**0.9656**	0.9628	0.9491	0.9622
(vote)	0.955	0.9491	0.9455	0.9452	**0.9498**	0.9395
(wine1)	0.677	**0.6789**	0.6611	0.6606	0.6606	0.6611
wine2	0.604	**0.6303**	0.6168	0.6232	0.6238	0.6232
(wine3)	0.726	**0.7211**	0.7189	0.7178	0.7172	0.7194
(winequality-red)	0.993	0.9922	0.9907	**0.9937**	**0.9937**	**0.9937**
yeast	0.701	**0.7148**	0.6925	0.6930	0.6899	0.6930
Average (all data sets)		0.8516	**0.8616**	0.8603	0.8559	0.8608
Average (filtered data sets)		0.8672	0.8804	0.8795	0.8736	**0.8804**

Table 5: Depth 3 – Test accuracy of CART and Blossom, without and with pruning (up to 1%, 3% or $0.3 \cdot k^2\%$ loss)

		CART	Blossom	Blossom $^{1\%}$	Blossom $^{3\%}$	Blossom *
	single test	test acc.	test acc.	test acc.	test acc.	test acc.
adult_discretized	0.795	0.8269	**0.8477**	0.8412	0.8411	0.8452
anneal	0.808	0.8011	**0.8722**	0.8591	0.8502	0.8616
audiology	0.874	**0.9541**	0.9518	0.9418	0.9509	0.9436
(australian-credit)	0.871	0.8611	**0.8697**	0.8665	0.8605	0.8655
(balance-scale)	0.921	0.9125	0.9073	**0.9206**	**0.9206**	0.9133
bank	0.893	0.9006	**0.9016**	0.8959	0.8881	0.8994
(bank_conv)	0.891	**0.8999**	0.8939	0.8960	0.8953	0.8966
banknote	0.843	0.9631	**0.9877**	0.9821	0.9632	0.9842
biodeg	0.760	0.8191	**0.8253**	0.8207	0.8175	0.8206
breast-cancer	0.895	**0.9391**	0.9307	0.9334	0.9293	0.9343
breast-wisconsin	0.923	0.9476	0.9525	**0.9553**	0.9534	0.9530
car	0.714	0.8819	**0.9073**	0.8988	0.8920	0.9013
car_evaluation	0.699	**0.9210**	0.9164	0.9180	0.9186	0.9182
(chess)	1.000	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
compas_discretized	0.644	0.6638	0.6696	0.6726	**0.6739**	0.6719
(default_credit)	0.820	**0.8209**	0.8209	0.8196	0.8196	0.8196
(diabetes)	0.744	0.7416	**0.7439**	0.7375	0.7355	0.7374
(forest-fires)	0.540	0.5438	**0.5646**	0.5425	0.5435	0.5425
(german-credit)	0.701	0.7090	0.7230	**0.7241**	0.7240	0.7193
hand_posture	0.794	0.9217	**0.9360**	0.9352	0.9228	0.9352
(heart-cleveland)	0.745	0.7640	0.7820	0.7777	**0.7887**	0.7757
(hepatitis)	0.826	**0.7766**	0.7587	0.7628	0.7745	0.7566
HTRU_2	0.962	**0.9765**	0.9764	0.9729	0.9651	0.9741
hypothyroid	0.962	0.9810	0.9796	0.9786	0.9660	**0.9812**
(IndiansDiabetes)	0.738	**0.7568**	0.7371	0.7277	0.7252	0.7288
ionosphere	0.813	0.8704	0.8578	0.8758	**0.8806**	0.8721
iris	0.937	0.9807	**0.9893**	0.9827	0.9700	0.9847
kr-vs-kp	0.675	0.9391	**0.9561**	0.9489	0.9361	0.9512
letter	0.959	0.9771	**0.9857**	0.9853	0.9593	0.9853
letter_recognition	0.961	0.9922	**0.9933**	0.9889	0.9881	0.9916
lymph	0.799	0.8258	0.8325	0.8342	**0.8484**	0.8277
magic04	0.738	0.8221	**0.8316**	0.8267	0.8131	0.8294
messidor	0.591	**0.6460**	0.6340	0.6360	0.6397	0.6348
mnist_0	0.920	0.9599	**0.9606**	0.9561	0.9431	0.9571
monk1	0.733	0.8462	0.9557	**0.9592**	0.9569	**0.9592**
monk2	0.618	0.5906	0.6367	0.6388	**0.6476**	0.6288
monk3	0.754	**0.9152**	0.8708	0.8832	0.8640	0.8832
mushroom	0.887	0.9991	**1.0000**	0.9941	0.9852	0.9989
pendigits	0.932	**0.9957**	0.9950	0.9899	0.9803	0.9924
(primary-tumor)	0.789	0.8074	**0.8571**	0.8450	0.8412	0.8444
segment	0.983	0.9975	**0.9984**	0.9956	0.9830	0.9960
(seismic_bumps)	0.934	0.9316	0.9303	0.9276	**0.9342**	0.9266
soybean	0.850	0.9266	0.9569	0.9573	0.9461	**0.9609**
spambase	0.786	0.8538	0.8585	0.8536	0.8383	**0.8586**
splice-1	0.822	0.9529	**0.9571**	0.9468	0.9457	0.9535
Statlog_satellite	0.757	0.9378	0.9614	0.9606	0.9509	**0.9618**
Statlog_shuttle	0.928	0.9989	**0.9999**	0.9961	0.9961	0.9969
surgical-deepnet	0.786	0.8137	**0.8447**	0.8338	0.8194	0.8402
(taiwan_binarised)	0.820	**0.8209**	0.8206	0.8196	0.8193	0.8198
tic-tac-toe	0.698	**0.8147**	0.7856	0.8045	0.8101	0.8008
titanic	0.780	0.8210	**0.8296**	0.8175	0.8133	0.8180
vehicle	0.769	0.9367	**0.9659**	0.9579	0.9473	0.9609
(vote)	0.955	0.9427	**0.9455**	0.9425	0.9425	0.9400
(wine1)	0.677	**0.6789**	0.6600	0.6578	0.6583	0.6583
wine2	0.604	0.6470	0.6274	0.6465	**0.6476**	0.6465
(wine3)	0.726	0.7217	**0.7300**	0.7167	0.7150	0.7167
(winequality-red)	0.993	0.9909	0.9917	**0.9937**	**0.9937**	0.9937
yeast	0.701	**0.7168**	0.6795	0.6900	0.6895	0.6896
Average (all data sets)		0.8613	**0.8682**	0.8663	0.8625	0.8665
Average (filtered data sets)		0.8819	**0.8905**	0.8891	0.8833	0.8901

Table 6: Depth 4 – Test accuracy of CART and Blossom, without and with pruning (up to 1%, 3% or $0.3 \cdot k^2\%$ loss)

		CART	Blossom	Blossom $^{1\%}$	Blossom $^{3\%}$	Blossom *
	single test	test acc.	test acc.	test acc.	test acc.	test acc.
adult_discretized	0.795	0.8401	**0.8520**	0.8459	0.8398	0.8474
anneal	0.808	0.8226	**0.8776**	0.8677	0.8629	0.8682
audiology	0.874	**0.9582**	0.9373	0.9418	0.9445	0.9418
(australian-credit)	0.871	**0.8573**	0.8342	0.8397	0.8482	0.8365
(balance-scale)	0.921	0.8987	0.8857	0.8887	**0.9206**	0.8867
bank	0.893	0.9008	**0.9024**	0.9002	0.8994	0.9004
(bank_conv)	0.891	0.8968	0.8946	0.8966	0.8953	**0.8969**
banknote	0.843	0.9826	**0.9896**	0.9820	0.9688	0.9825
biodeg	0.760	0.8250	0.8275	0.8280	0.8251	**0.8288**
breast-cancer	0.895	**0.9390**	0.9357	0.9331	0.9334	0.9331
breast-wisconsin	0.923	0.9491	0.9443	0.9514	**0.9521**	0.9502
car	0.714	0.9349	**0.9397**	0.9289	0.9283	0.9284
car_evaluation	0.699	0.9250	**0.9460**	0.9410	0.9372	0.9431
(chess)	1.000	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
compas_discretized	0.644	0.6731	0.6697	0.6711	**0.6732**	0.6709
(default_credit)	0.820	0.8203	**0.8213**	0.8196	0.8196	0.8200
(diabetes)	0.744	**0.7368**	0.7140	0.7271	0.7270	0.7266
(forest-fires)	0.540	0.5462	**0.5585**	0.5365	0.5356	0.5367
(german-credit)	0.701	0.6988	0.7144	0.7205	**0.7272**	0.7191
hand_posture	0.794	0.9559	**0.9581**	0.9572	0.9368	0.9572
(heart-cleveland)	0.745	0.7487	**0.7727**	0.7563	0.7593	0.7527
(hepatitis)	0.826	**0.7614**	0.7400	0.7441	0.7545	0.7448
HTRU_2	0.962	0.9767	**0.9772**	0.9736	0.9637	0.9742
hypothyroid	0.962	**0.9803**	0.9781	0.9783	0.9715	0.9792
(IndiansDiabetes)	0.738	**0.7304**	0.7086	0.7138	0.7175	0.7118
ionosphere	0.813	0.8769	0.8689	0.8738	**0.8772**	0.8738
iris	0.937	0.9807	**0.9893**	0.9827	0.9700	0.9847
kr-vs-kp	0.675	0.9391	**0.9712**	0.9633	0.9499	0.9653
letter	0.959	0.9814	**0.9886**	0.9845	0.9845	0.9848
letter_recognition	0.961	0.9931	**0.9945**	0.9887	0.9867	0.9898
lymph	0.799	**0.8161**	0.8150	0.8097	0.8142	0.8071
magic04	0.738	0.8281	**0.8410**	0.8370	0.8223	0.8374
messidor	0.591	0.6432	**0.6452**	0.6314	0.6346	0.6322
mnist_0	0.920	0.9649	**0.9688**	0.9622	0.9500	0.9667
monk1	0.733	0.8415	**1.0000**	**1.0000**	0.9915	**1.0000**
monk2	0.618	0.5900	**0.7033**	0.6824	0.6806	0.6824
monk3	0.754	**0.9048**	0.8246	0.8184	0.8232	0.8184
mushroom	0.887	0.9995	**1.0000**	0.9941	0.9852	0.9947
pendigits	0.932	0.9966	**0.9970**	0.9915	0.9811	0.9930
(primary-tumor)	0.789	0.8003	0.8103	0.8047	**0.8221**	0.8009
segment	0.983	0.9975	**0.9982**	0.9955	0.9830	0.9956
(seismic_bumps)	0.934	0.9296	0.9151	0.9223	**0.9342**	0.9215
soybean	0.850	0.9435	0.9459	**0.9546**	0.9534	0.9520
spambase	0.786	0.8652	**0.8741**	0.8701	0.8596	0.8716
splice-1	0.822	0.9527	0.9519	0.9561	0.9409	**0.9563**
Statlog_satellite	0.757	0.9593	0.9657	0.9665	0.9528	**0.9671**
Statlog_shuttle	0.928	0.9995	**0.9999**	0.9961	0.9961	0.9961
surgical-deepnet	0.786	0.8342	**0.8499**	0.8467	0.8350	0.8481
(taiwan_binarised)	0.820	**0.8202**	0.8194	0.8195	0.8196	0.8202
tic-tac-toe	0.698	**0.8994**	0.8581	0.8830	0.8738	0.8827
titanic	0.780	**0.8164**	0.8131	0.8093	0.8111	0.8089
vehicle	0.769	0.9459	0.9553	**0.9564**	0.9478	0.9547
(vote)	0.955	0.9427	0.9291	0.9395	**0.9473**	0.9373
(wine1)	0.677	**0.6800**	0.6678	0.6578	0.6572	0.6578
wine2	0.604	0.6519	0.6379	0.6595	**0.6605**	0.6595
(wine3)	0.726	**0.7200**	0.7133	0.7056	0.7050	0.7056
(winequality-red)	0.993	0.9895	0.9884	**0.9937**	**0.9937**	**0.9937**
yeast	0.701	**0.7136**	0.6943	0.6980	0.7014	0.6979
Average (all data sets)		0.8651	**0.8685**	0.8672	0.8653	0.8672
Average (filtered data sets)		0.8900	**0.8972**	0.8953	0.8901	0.8957

Table 7: Depth 5 – Test accuracy of CART and Blossom, without and with pruning (up to 1%, 3% or $0.3 \cdot k^2\%$ loss)

		CART	Blossom	Blossom $^{1\%}$	Blossom $^{3\%}$	Blossom *
	single test	test acc.	test acc.	test acc.	test acc.	test acc.
adult_discretized	0.795	0.8473	**0.8540**	0.8491	0.8385	0.8483
anneal	0.808	0.8394	**0.8729**	0.8550	0.8523	0.8545
audiology	0.874	**0.9555**	0.9382	0.9514	0.9473	0.9514
(australian-credit)	0.871	**0.8468**	0.8176	0.8277	0.8395	0.8277
(balance-scale)	0.921	**0.8865**	0.8390	0.8556	0.8819	0.8592
bank	0.893	0.9005	**0.9023**	0.9016	0.8984	0.9010
(bank_conv)	0.891	0.8964	0.8937	0.8956	**0.8970**	0.8958
banknote	0.843	0.9865	**0.9899**	0.9826	0.9707	0.9819
biodeg	0.760	0.8250	0.8311	**0.8364**	0.8358	0.8363
breast-cancer	0.895	**0.9371**	0.9258	0.9327	0.9349	0.9327
breast-wisconsin	0.923	0.9448	0.9458	0.9485	**0.9561**	0.9485
car	0.714	0.9348	**0.9710**	0.9663	0.9552	0.9658
car_evaluation	0.699	**0.9403**	0.9296	0.9379	0.9325	0.9372
(chess)	1.000	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
compas_discretized	0.644	**0.6694**	0.6654	0.6684	0.6694	0.6686
(default_credit)	0.820	**0.8199**	0.8193	0.8197	0.8196	0.8198
(diabetes)	0.744	0.7199	0.6891	0.7149	**0.7209**	0.7149
(forest-fires)	0.540	0.5515	**0.5596**	0.5390	0.5388	0.5390
(german-credit)	0.701	0.6962	0.6998	0.7017	**0.7078**	0.7009
hand_posture	0.794	0.9773	**0.9790**	0.9784	0.9554	0.9784
(heart-cleveland)	0.745	0.7423	0.7440	0.7553	**0.7643**	0.7570
(hepatitis)	0.826	**0.7628**	0.7360	0.7503	0.7566	0.7503
HTRU_2	0.962	**0.9768**	0.9767	0.9759	0.9710	0.9759
hypothyroid	0.962	**0.9796**	0.9778	0.9763	0.9684	0.9760
(IndiansDiabetes)	0.738	**0.7279**	0.7026	0.6981	0.7045	0.6981
ionosphere	0.813	0.8777	0.8806	0.8814	**0.8887**	0.8868
iris	0.937	0.9807	**0.9893**	0.9827	0.9700	0.9827
kr-vs-kp	0.675	0.9363	**0.9734**	0.9680	0.9518	0.9672
letter	0.959	0.9866	**0.9894**	0.9858	0.9839	0.9849
letter_recognition	0.961	0.9938	**0.9939**	0.9893	0.9892	0.9893
lymph	0.799	**0.8277**	0.8125	0.8058	0.8123	0.8058
magic04	0.738	0.8371	**0.8467**	0.8423	0.8313	0.8418
messidor	0.591	**0.6306**	0.6250	0.6270	0.6296	0.6274
mnist_0	0.920	0.9678	**0.9719**	0.9669	0.9612	0.9662
monk1	0.733	0.8423	**1.0000**	**1.0000**	0.9915	**1.0000**
monk2	0.618	0.6606	**0.8800**	0.8488	0.8276	0.8488
monk3	0.754	**0.8928**	0.8277	0.8544	0.8712	0.8680
mushroom	0.887	0.9997	**1.0000**	0.9941	0.9852	0.9941
pendigits	0.932	**0.9974**	0.9970	0.9914	0.9847	0.9911
(primary-tumor)	0.789	**0.8065**	0.7886	0.7821	0.7941	0.7821
segment	0.983	0.9972	**0.9982**	0.9955	0.9830	0.9955
(seismic_bumps)	0.934	**0.9269**	0.9127	0.9163	0.9227	0.9163
soybean	0.850	**0.9469**	0.9413	0.9457	0.9452	0.9457
spambase	0.786	0.8753	**0.8801**	0.8786	0.8717	0.8784
splice-1	0.822	**0.9552**	0.9436	0.9537	0.9513	0.9536
Statlog_satellite	0.757	0.9633	0.9659	**0.9661**	0.9559	0.9656
Statlog_shuttle	0.928	0.9996	**0.9999**	0.9961	0.9961	0.9961
surgical-deepnet	0.786	0.8510	**0.8606**	0.8555	0.8462	0.8554
(taiwan_binarised)	0.820	0.8196	0.8175	0.8196	0.8196	**0.8199**
tic-tac-toe	0.698	0.9231	0.9223	**0.9321**	0.9211	0.9320
titanic	0.780	0.8120	0.8164	0.8092	**0.8174**	0.8103
vehicle	0.769	0.9489	**0.9659**	0.9579	0.9551	0.9576
(vote)	0.955	0.9380	0.9436	0.9455	**0.9509**	0.9455
(wine1)	0.677	**0.6694**	0.6678	0.6461	0.6444	0.6461
wine2	0.604	0.6497	0.6411	0.6562	**0.6573**	0.6562
(wine3)	0.726	**0.7183**	0.6956	0.7022	0.7017	0.7022
(winequality-red)	0.993	0.9878	0.9897	**0.9937**	**0.9937**	**0.9937**
yeast	0.701	0.7074	**0.7129**	0.7054	0.7086	0.7054
Average (all data sets)		0.8671	0.8709	0.8709	0.8695	**0.8712**
Average (filtered data sets)		0.8944	**0.9049**	0.9038	0.8993	0.9041

Table 8: Depth 6 – Test accuracy of CART and Blossom, without and with pruning (up to 1%, 3% or $0.3 \cdot k^2\%$ loss)

		CART	Blossom	Blossom $^{1\%}$	Blossom $^{3\%}$	Blossom *
	single test	test acc.	test acc.	test acc.	test acc.	test acc.
adult_discretized	0.795	0.8490	**0.8535**	0.8511	0.8426	0.8489
anneal	0.808	0.8480	**0.8724**	0.8601	0.8618	0.8611
audiology	0.874	**0.9555**	0.9382	0.9505	0.9473	0.9505
(australian-credit)	0.871	**0.8371**	0.8018	0.8189	0.8291	0.8215
(balance-scale)	0.921	**0.8600**	0.7844	0.8019	0.8308	0.8116
bank	0.893	0.9009	0.9018	**0.9027**	0.8987	0.9020
(bank_conv)	0.891	0.8931	0.8920	0.8952	**0.8973**	0.8967
banknote	0.843	**0.9894**	0.9890	0.9817	0.9729	0.9801
biodeg	0.760	0.8248	0.8221	0.8282	**0.8327**	0.8287
breast-cancer	0.895	0.9339	0.9328	0.9368	**0.9439**	0.9374
breast-wisconsin	0.923	0.9416	0.9414	0.9476	**0.9509**	0.9486
car	0.714	0.9646	**0.9724**	0.9665	0.9555	0.9639
car_evaluation	0.699	**0.9376**	0.9146	0.9349	0.9299	0.9349
(chess)	1.000	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
compas_discretized	0.644	0.6666	0.6633	0.6672	**0.6688**	0.6680
(default_credit)	0.820	0.8188	0.8177	0.8188	**0.8196**	0.8192
(diabetes)	0.744	**0.7179**	0.6655	0.6791	0.6919	0.6819
(forest-fires)	0.540	0.5458	**0.5638**	0.5321	0.5319	0.5321
(german-credit)	0.701	**0.6946**	0.6834	0.6843	0.6911	0.6862
hand_posture	0.794	0.9871	**0.9884**	0.9860	0.9629	0.9860
(heart-cleveland)	0.745	0.7283	0.7527	0.7617	**0.7733**	0.7647
(hepatitis)	0.826	**0.7607**	0.7307	0.7545	0.7600	0.7545
HTRU_2	0.962	**0.9768**	0.9757	0.9761	0.9700	0.9759
hypothyroid	0.962	**0.9784**	0.9756	0.9728	0.9703	0.9720
(IndiansDiabetes)	0.738	**0.7164**	0.6751	0.6757	0.6886	0.6792
ionosphere	0.813	0.8789	0.8694	0.8820	**0.8932**	0.8896
iris	0.937	0.9807	**0.9893**	0.9827	0.9700	0.9827
kr-vs-kp	0.675	0.9628	**0.9848**	0.9780	0.9718	0.9765
letter	0.959	0.9893	**0.9908**	0.9893	0.9840	0.9886
letter_recognition	0.961	0.9941	**0.9943**	0.9905	0.9892	0.9892
lymph	0.799	**0.8252**	0.8137	0.8142	0.8194	0.8142
magic04	0.738	0.8404	0.8454	**0.8457**	0.8381	0.8445
messidor	0.591	**0.6317**	0.6191	0.6171	0.6228	0.6184
mnist_0	0.920	0.9742	**0.9750**	0.9709	0.9603	0.9687
monk1	0.733	0.8569	**1.0000**	**1.0000**	0.9915	**1.0000**
monk2	0.618	0.6900	**0.8722**	0.8512	0.8306	0.8512
monk3	0.754	**0.8896**	0.8569	0.8728	0.8872	0.8880
mushroom	0.887	1.0000	**1.0000**	0.9941	0.9852	0.9932
pendigits	0.932	0.9976	**0.9978**	0.9918	0.9886	0.9886
(primary-tumor)	0.789	**0.7988**	0.7829	0.7679	0.7865	0.7694
segment	0.983	0.9974	**0.9982**	0.9955	0.9830	0.9954
(seismic_bumps)	0.934	**0.9233**	0.8985	0.9062	0.9176	0.9078
soybean	0.850	0.9482	0.9425	0.9520	0.9496	**0.9534**
spambase	0.786	0.8816	**0.8826**	0.8809	0.8752	0.8794
splice-1	0.822	0.9515	0.9488	0.9567	0.9577	**0.9582**
Statlog_satellite	0.757	0.9669	0.9672	**0.9679**	0.9600	0.9668
Statlog_shuttle	0.928	0.9997	**0.9999**	0.9961	0.9961	0.9961
surgical-deepnet	0.786	0.8606	**0.8679**	0.8647	0.8604	0.8634
(taiwan_binarised)	0.820	0.8186	0.8149	0.8177	**0.8196**	0.8181
tic-tac-toe	0.698	0.9339	**0.9753**	0.9645	0.9450	0.9609
titanic	0.780	0.8051	0.8031	0.7912	**0.8124**	0.7980
vehicle	0.769	0.9452	0.9475	**0.9558**	0.9535	0.9535
(vote)	0.955	0.9366	0.9427	0.9468	**0.9520**	0.9486
(wine1)	0.677	**0.6722**	0.6689	0.6561	0.6539	0.6561
wine2	0.604	0.6546	0.6484	0.6627	**0.6638**	0.6632
(wine3)	0.726	**0.7100**	0.6944	0.7050	0.7033	0.7044
(winequality-red)	0.993	0.9863	0.9895	0.9936	**0.9937**	**0.9937**
yeast	0.701	**0.7119**	0.6883	0.7019	0.7080	0.7013
Average (all data sets)		0.8679	0.8686	0.8698	0.8697	**0.8705**
Average (filtered data sets)		0.8980	0.9055	0.9058	0.9026	**0.9060**

Table 9: Depth 7 – Test accuracy of CART and Blossom, without and with pruning (up to 1%, 3% or $0.3 \cdot k^2\%$ loss)

		CART	Blossom	Blossom $^{1\%}$	Blossom $^{3\%}$	Blossom *
	single test	test acc.	test acc.	test acc.	test acc.	test acc.
adult_discretized	0.795	0.8502	**0.8528**	0.8515	0.8450	0.8488
anneal	0.808	0.8520	**0.8763**	0.8648	0.8694	0.8671
audiology	0.874	**0.9523**	0.9382	0.9505	0.9473	0.9518
(australian-credit)	0.871	0.8279	0.8185	0.8268	**0.8420**	0.8380
(balance-scale)	0.921	**0.8478**	0.7530	0.7884	0.8156	0.8005
bank	0.893	0.9002	0.9009	**0.9029**	0.8995	0.9021
(bank_conv)	0.891	0.8921	0.8916	0.8929	**0.8958**	0.8952
banknote	0.843	**0.9914**	0.9888	0.9827	0.9749	0.9793
biodeg	0.760	**0.8284**	0.8234	0.8206	0.8231	0.8227
breast-cancer	0.895	0.9350	0.9316	0.9320	**0.9391**	0.9345
breast-wisconsin	0.923	0.9425	0.9441	0.9447	**0.9479**	0.9474
car	0.714	0.9649	**0.9910**	0.9790	0.9673	0.9748
car_evaluation	0.699	0.9288	0.9126	0.9340	0.9336	**0.9363**
(chess)	1.000	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
compas_discretized	0.644	0.6649	0.6604	0.6628	**0.6673**	0.6658
(default_credit)	0.820	0.8165	0.8141	0.8163	**0.8196**	0.8184
(diabetes)	0.744	**0.7113**	0.6727	0.6921	0.7064	0.6986
(forest-fires)	0.540	0.5477	**0.5546**	0.5433	0.5440	0.5419
(german-credit)	0.701	**0.6916**	0.6846	0.6775	0.6846	0.6803
hand_posture	0.794	0.9892	**0.9908**	0.9878	0.9765	0.9878
(heart-cleveland)	0.745	0.7430	0.7280	0.7490	**0.7573**	0.7520
(hepatitis)	0.826	0.7579	0.7467	0.7545	**0.7607**	0.7593
HTRU_2	0.962	0.9761	0.9743	**0.9769**	0.9752	0.9759
hypothyroid	0.962	**0.9781**	0.9730	0.9749	0.9762	0.9755
(IndiansDiabetes)	0.738	**0.7123**	0.6597	0.6556	0.6708	0.6613
ionosphere	0.813	0.8848	0.8661	0.8837	**0.8927**	0.8913
iris	0.937	0.9807	**0.9893**	0.9827	0.9700	0.9720
kr-vs-kp	0.675	0.9823	**0.9843**	0.9819	0.9669	0.9765
letter	0.959	0.9906	**0.9917**	0.9900	0.9834	0.9855
letter_recognition	0.961	**0.9942**	0.9941	0.9913	0.9892	0.9892
lymph	0.799	**0.8213**	0.8163	0.8142	0.8194	0.8168
magic04	0.738	0.8410	0.8446	**0.8466**	0.8423	0.8454
messidor	0.591	**0.6275**	0.6133	0.6158	0.6235	0.6173
mnist_0	0.920	0.9776	**0.9794**	0.9757	0.9671	0.9705
monk1	0.733	0.8731	**1.0000**	**1.0000**	0.9915	**1.0000**
monk2	0.618	0.6953	**0.8722**	0.8500	0.8294	0.8429
monk3	0.754	**0.9000**	0.8600	0.8712	0.8824	0.8832
mushroom	0.887	**1.0000**	**1.0000**	0.9942	0.9852	0.9852
pendigits	0.932	0.9975	**0.9977**	0.9918	0.9888	0.9888
(primary-tumor)	0.789	**0.7894**	0.7657	0.7718	0.7838	0.7774
segment	0.983	0.9971	**0.9982**	0.9955	0.9830	0.9925
(seismic_bumps)	0.934	**0.9178**	0.8900	0.8993	0.9145	0.9070
soybean	0.850	0.9490	0.9484	0.9545	0.9546	**0.9569**
spambase	0.786	0.8847	**0.8869**	0.8839	0.8793	0.8818
splice-1	0.822	0.9495	0.9508	0.9541	**0.9592**	0.9581
Statlog_satellite	0.757	**0.9695**	0.9673	0.9681	0.9627	0.9668
Statlog_shuttle	0.928	0.9998	**0.9999**	0.9961	0.9961	0.9961
surgical-deepnet	0.786	0.8769	**0.8858**	0.8804	0.8758	0.8792
(taiwan_binarised)	0.820	0.8169	0.8132	0.8145	**0.8190**	0.8173
tic-tac-toe	0.698	0.9398	**0.9734**	0.9654	0.9485	0.9606
titanic	0.780	0.8046	0.7967	0.7926	**0.8082**	0.7990
vehicle	0.769	0.9442	0.9494	0.9544	**0.9548**	0.9538
(vote)	0.955	0.9357	0.9450	0.9464	**0.9525**	0.9486
(wine1)	0.677	**0.6800**	0.6678	0.6639	0.6617	0.6639
wine2	0.604	0.6600	0.6516	0.6714	**0.6719**	**0.6719**
(wine3)	0.726	0.7050	0.6900	0.7083	0.7061	**0.7089**
(winequality-red)	0.993	0.9864	0.9895	0.9936	**0.9937**	**0.9937**
yeast	0.701	**0.7072**	0.6829	0.6928	0.6990	0.6950
Average (all data sets)		0.8686	0.8680	0.8700	0.8706	**0.8708**
Average (filtered data sets)		0.9001	0.9065	**0.9066**	0.9042	0.9062

Table 10: Depth 8 – Test accuracy of CART and Blossom, without and with pruning (up to 1%, 3% or $0.3 \cdot k^2\%$ loss)

		CART	Blossom	Blossom $^{1\%}$	Blossom $^{3\%}$	Blossom *
	single test	test acc.	test acc.	test acc.	test acc.	test acc.
adult_discretized	0.795	**0.8531**	0.8522	0.8529	0.8489	0.8497
anneal	0.808	0.8582	**0.8678**	0.8568	0.8620	0.8610
audiology	0.874	**0.9541**	0.9382	0.9505	0.9473	0.9477
(australian-credit)	0.871	0.8218	0.8197	0.8305	**0.8448**	0.8406
(balance-scale)	0.921	**0.8371**	0.7451	0.7649	0.7857	0.7833
bank	0.893	0.8998	0.9005	**0.9028**	0.9009	0.9025
(bank_conv)	0.891	0.8897	0.8906	0.8922	**0.8963**	0.8956
banknote	0.843	**0.9912**	0.9888	0.9826	0.9748	0.9775
biodeg	0.760	**0.8258**	0.8202	0.8186	0.8249	0.8247
breast-cancer	0.895	0.9320	0.9322	0.9349	**0.9391**	0.9380
breast-wisconsin	0.923	0.9418	0.9446	0.9439	**0.9470**	0.9457
car	0.714	0.9803	**0.9953**	0.9831	0.9701	0.9737
car_evaluation	0.699	0.9218	0.9126	**0.9392**	0.9357	0.9378
(chess)	1.000	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
compas_discretized	0.644	0.6622	0.6607	0.6623	**0.6661**	0.6653
(default_credit)	0.820	0.8146	0.8106	0.8138	**0.8182**	0.8170
(diabetes)	0.744	0.7030	0.6816	0.6943	**0.7100**	0.7062
(forest-fires)	0.540	0.5438	**0.5658**	0.5350	0.5360	0.5352
(german-credit)	0.701	0.6827	0.6694	0.6790	**0.6841**	0.6823
hand_posture	0.794	0.9904	**0.9925**	0.9890	0.9849	0.9890
(heart-cleveland)	0.745	0.7323	0.7260	0.7530	**0.7587**	0.7567
(hepatitis)	0.826	**0.7641**	0.7440	0.7545	0.7607	0.7593
HTRU_2	0.962	0.9755	0.9730	**0.9775**	0.9759	0.9759
hypothyroid	0.962	0.9770	0.9741	0.9760	**0.9782**	0.9772
(IndiansDiabetes)	0.738	**0.7038**	0.6657	0.6708	0.6804	0.6769
ionosphere	0.813	0.8772	0.8617	0.8851	**0.8918**	0.8896
iris	0.937	0.9807	**0.9893**	0.9827	0.9700	0.9720
kr-vs-kp	0.675	**0.9898**	0.9890	0.9845	0.9753	0.9783
letter	0.959	0.9916	**0.9920**	0.9906	0.9839	0.9844
letter_recognition	0.961	0.9944	**0.9947**	0.9917	0.9892	0.9892
lymph	0.799	**0.8284**	0.8163	0.8135	0.8187	0.8161
magic04	0.738	0.8412	0.8433	0.8445	0.8446	**0.8452**
messidor	0.591	**0.6273**	0.6048	0.6087	0.6171	0.6143
mnist_0	0.920	0.9801	**0.9821**	0.9784	0.9692	0.9700
monk1	0.733	0.8654	**1.0000**	**1.0000**	0.9915	0.9915
monk2	0.618	0.7071	**0.8733**	0.8500	0.8294	0.8429
monk3	0.754	**0.8968**	0.8631	0.8736	0.8848	0.8848
mushroom	0.887	**1.0000**	**1.0000**	0.9942	0.9852	0.9852
pendigits	0.932	0.9975	**0.9977**	0.9918	0.9888	0.9888
(primary-tumor)	0.789	**0.7847**	0.7663	0.7700	0.7750	0.7726
segment	0.983	0.9974	**0.9982**	0.9955	0.9830	0.9830
(seismic_bumps)	0.934	**0.9122**	0.8836	0.8958	0.9118	0.9075
soybean	0.850	0.9463	0.9434	**0.9513**	0.9480	0.9509
spambase	0.786	0.8859	0.8877	**0.8879**	0.8847	0.8862
splice-1	0.822	0.9480	0.9447	0.9547	**0.9605**	0.9590
Statlog_satellite	0.757	0.9692	0.9668	**0.9696**	0.9653	0.9671
Statlog_shuttle	0.928	0.9998	**0.9999**	0.9961	0.9961	0.9961
surgical-deepnet	0.786	0.8900	0.8937	**0.8950**	0.8927	0.8944
(taiwan_binarised)	0.820	0.8148	0.8097	0.8122	**0.8171**	0.8161
tic-tac-toe	0.698	0.9402	0.9623	0.9610	0.9582	**0.9629**
titanic	0.780	**0.8020**	0.7898	0.7842	0.8001	0.7939
vehicle	0.769	0.9440	0.9506	**0.9554**	0.9532	0.9538
(vote)	0.955	0.9375	0.9450	0.9466	**0.9539**	0.9511
(wine1)	0.677	0.6650	**0.6667**	0.6600	0.6578	0.6606
wine2	0.604	0.6681	0.6474	0.6670	**0.6697**	0.6670
(wine3)	0.726	0.7039	0.6800	**0.7089**	0.7006	0.7050
(winequality-red)	0.993	0.9856	0.9892	**0.9937**	**0.9937**	**0.9937**
yeast	0.701	**0.7082**	0.6819	0.6825	0.6917	0.6879
Average (all data sets)		0.8679	0.8670	0.8696	**0.8704**	0.8704
Average (filtered data sets)		0.9010	0.9057	**0.9065**	0.9050	0.9055

Table 11: Depth 9 – Test accuracy of CART and Blossom, without and with pruning (up to 1%, 3% or $0.3 \cdot k^2\%$ loss)

		CART	Blossom	Blossom $^{1\%}$	Blossom $^{3\%}$	Blossom *
	single test	test acc.	test acc.	test acc.	test acc.	test acc.
adult_discretized	0.795	0.8532	0.8523	**0.8541**	0.8519	0.8519
anneal	0.808	0.8579	0.8768	0.8748	0.8784	**0.8785**
audiology	0.874	**0.9527**	0.9382	0.9505	0.9473	0.9473
(australian-credit)	0.871	0.8203	0.8200	0.8291	**0.8415**	**0.8415**
(balance-scale)	0.921	**0.8306**	0.7479	0.7681	0.7897	0.7897
bank	0.893	0.8990	0.8991	0.9022	**0.9027**	0.9027
(bank_conv)	0.891	0.8886	0.8898	0.8906	**0.8950**	0.8949
banknote	0.843	**0.9913**	0.9888	0.9826	0.9748	0.9748
biodeg	0.760	0.8211	0.8175	0.8185	**0.8249**	**0.8249**
breast-cancer	0.895	0.9336	0.9313	0.9352	**0.9396**	**0.9396**
breast-wisconsin	0.923	0.9412	0.9455	0.9438	**0.9472**	0.9467
car	0.714	0.9790	**0.9963**	0.9831	0.9706	0.9706
car_evaluation	0.699	0.9176	0.9126	**0.9392**	0.9374	0.9374
(chess)	1.000	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
compas_discretized	0.644	0.6594	0.6601	0.6625	**0.6668**	**0.6668**
(default_credit)	0.820	0.8116	0.8061	0.8109	0.8155	**0.8156**
(diabetes)	0.744	0.7018	0.6865	0.6943	0.7081	**0.7084**
(forest-fires)	0.540	0.5402	**0.5769**	0.5317	0.5329	0.5333
(german-credit)	0.701	0.6844	0.6638	0.6796	0.6852	**0.6863**
hand_posture	0.794	0.9911	**0.9939**	0.9892	0.9886	0.9886
(heart-cleveland)	0.745	0.7347	0.7227	0.7513	**0.7583**	**0.7583**
(hepatitis)	0.826	0.7552	0.7440	0.7545	**0.7607**	**0.7607**
HTRU_2	0.962	0.9744	0.9712	**0.9775**	0.9759	0.9759
hypothyroid	0.962	0.9761	0.9729	0.9768	**0.9786**	**0.9786**
(IndiansDiabetes)	0.738	**0.6964**	0.6636	0.6678	0.6821	0.6821
ionosphere	0.813	0.8769	0.8650	0.8820	**0.8924**	**0.8924**
iris	0.937	0.9807	**0.9893**	0.9827	0.9700	0.9700
kr-vs-kp	0.675	**0.9927**	0.9904	0.9848	0.9747	0.9747
letter	0.959	0.9919	**0.9920**	0.9906	0.9838	0.9838
letter_recognition	0.961	0.9949	**0.9955**	0.9924	0.9893	0.9893
lymph	0.799	**0.8316**	0.8163	0.8135	0.8168	0.8168
magic04	0.738	0.8393	0.8352	0.8384	**0.8425**	0.8424
messidor	0.591	**0.6287**	0.6002	0.6094	0.6138	0.6154
mnist_0	0.920	0.9827	**0.9838**	0.9812	0.9702	0.9702
monk1	0.733	0.8692	**1.0000**	**1.0000**	0.9915	0.9915
monk2	0.618	0.6965	**0.8722**	0.8494	0.8294	0.8294
monk3	0.754	**0.8992**	0.8631	0.8736	0.8848	0.8848
mushroom	0.887	**1.0000**	**1.0000**	0.9942	0.9852	0.9852
pendigits	0.932	0.9976	**0.9977**	0.9918	0.9888	0.9888
(primary-tumor)	0.789	0.7741	0.7760	0.7753	**0.7803**	**0.7803**
segment	0.983	0.9974	**0.9982**	0.9955	0.9830	0.9830
(seismic_bumps)	0.934	0.9040	0.8815	0.8955	0.9115	**0.9117**
soybean	0.850	0.9430	0.9428	**0.9498**	0.9488	0.9488
spambase	0.786	0.8884	0.8890	**0.8907**	0.8887	0.8888
splice-1	0.822	0.9466	0.9441	0.9533	**0.9608**	0.9607
Statlog_satellite	0.757	0.9692	0.9670	**0.9694**	0.9659	0.9659
Statlog_shuttle	0.928	0.9998	**0.9999**	0.9961	0.9961	0.9961
surgical-deepnet	0.786	0.8903	0.8919	0.8946	**0.8967**	**0.8967**
(taiwan_binarised)	0.820	0.8117	0.8069	0.8097	**0.8144**	0.8144
tic-tac-toe	0.698	0.9415	**0.9619**	0.9613	0.9581	0.9588
titanic	0.780	**0.7954**	0.7884	0.7785	0.7898	0.7898
vehicle	0.769	0.9440	0.9494	**0.9545**	0.9535	0.9535
(vote)	0.955	0.9370	0.9445	0.9475	0.9539	**0.9541**
(wine1)	0.677	0.6700	**0.6733**	0.6572	0.6561	0.6561
wine2	0.604	0.6546	0.6516	0.6665	**0.6676**	**0.6676**
(wine3)	0.726	**0.7094**	0.6833	0.7089	0.7044	0.7044
(winequality-red)	0.993	0.9851	0.9892	**0.9937**	**0.9937**	**0.9937**
yeast	0.701	**0.7037**	0.7015	0.6940	0.6989	0.6991
Average (all data sets)		0.8665	0.8676	0.8697	0.8708	**0.8709**
Average (filtered data sets)		0.9001	0.9061	**0.9070**	0.9056	0.9057

Table 12: Depth 10 – Test accuracy of CART and Blossom, without and with pruning (up to 1%, 3% or $0.3 \cdot k^2\%$ loss)

	Blossom		Murtree		Murtree^P		DL8.5		DL8.5^P		CP		CP^P		binOCT		CART	
	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu
adult_discretized	**5020**	0.43*	**5020**	0.84*	**5020**	0.51*	**5020**	8.4*	**5020**	3.6*	**5020**	6.4*	**5020**	3.8*	5600	3503	5758	0.05
anneal	**112**	0.03*	**112**	0.14*	**112**	0.06*	**112**	2.1*	**112**	0.40*	**112**	6.0*	**112**	3.0*	123	3042	149	0.00
balance-scale	**49**	0.00*	**49**	0.01*	**49**	0.00*	**49**	0.02*	**49**	0.02*	**49**	0.55*	**49**	1.0*	-	-	**49**	0.00
banknote	**36**	0.01*	**36**	0.02*	**36**	0.02*	**36**	0.09*	**36**	0.06*	**36**	0.88*	**36**	1.4*	-	-	118	0.00
breast-cancer	**24**	0.16*	**24**	0.07*	**24**	0.06*	**24**	0.89*	**24**	0.78*	**24**	5.7*	**24**	4.4*	25	3131	28	0.00
car	**192**	0.01*	**192**	0.01*	**192**	0.01*	**192**	0.03*	**192**	0.03*	**192**	1.7*	**192**	1.4*	**192**	1141	202	0.00
car_evaluation	**202**	0.00*	**202**	0.01*	**202**	0.01*	**202**	0.02*	**202**	0.01*	**202**	0.44*	**202**	0.62*	-	-	226	0.00
chess	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.04*	**0**	0.14*	-	-	**0**	0.00
compas_discretized	**2004**	0.00*	**2004**	0.06*	**2004**	0.02*	**2004**	0.21*	**2004**	0.04*	**2004**	1.8*	**2004**	1.5*	2032	806	2072	0.01
heart-cleveland	**41**	0.05*	**41**	0.12*	**41**	0.04*	**41**	3.5*	**41**	0.60*	**41**	6.8*	**41**	4.5*	42	870	43	0.00
hepatitis	**10**	0.00*	**10**	0.03*	**10**	0.01*	**10**	1.1*	**10**	0.10*	**10**	3.9*	**10**	2.5*	**10**	2314	16	0.00
HTRU_2	**401**	1.2*	**401**	3.6*	**401**	2.8*	**401**	12*	**401**	7.3*	**401**	5.7*	**401**	4.9*	-	-	422	0.05
hypothyroid	**61**	0.07*	**61**	0.41*	**61**	0.19*	**61**	3.8*	**61**	0.52*	**61**	6.6*	**61**	2.8*	62	2662	62	0.01
IndiansDiabetes	**166**	0.02*	**166**	0.07*	**166**	0.05*	**166**	0.31*	**166**	0.29*	**166**	1.6*	**166**	4.0*	-	-	180	0.00
iris	**1**	0.00*	**1**	0.00*	**1**	0.00*	**1**	0.00*	**1**	0.00*	**1**	0.15*	**1**	0.07*	-	-	**1**	0.00
kr-vs-kp	**198**	0.09*	**198**	0.22*	**198**	0.04*	**198**	2.3*	**198**	0.17*	**198**	4.8*	**198**	3.0*	375	2200	306	0.01
lymph	**12**	0.01*	**12**	0.03*	**12**	0.01*	**12**	0.56*	**12**	0.19*	**12**	3.7*	**12**	3.1*	14	2298	17	0.00
magic04	**3446**	3.8*	**3446**	7.6*	**3446**	7.5*	**3446**	26*	**3446**	20*	**3446**	11*	**3446**	10*	-	-	3788	0.06
messidor	**366**	0.25*	**366**	0.63*	**366**	0.50*	**366**	4.9*	**366**	3.6*	**366**	5.3*	**366**	5.4*	-	-	384	0.00
monk1	**11**	0.00*	**11**	0.00*	**11**	0.00*	**11**	0.00*	**11**	0.00*	**11**	0.27*	**11**	0.44*	-	-	**11**	0.00
monk2	**42**	0.00*	**42**	0.00*	**42**	0.00*	**42**	0.01*	**42**	0.01*	**42**	0.37*	**42**	0.60*	-	-	57	0.00
monk3	**6**	0.00*	**6**	0.00*	**6**	0.00*	**6**	0.00*	**6**	0.00*	**6**	0.32*	**6**	0.41*	-	-	7	0.00
primary-tumor	**46**	0.00*	**46**	0.01*	**46**	0.00*	**46**	0.12*	**46**	0.02*	**46**	2.0*	**46**	0.95*	**46**	2722	53	0.00
seismic_bumps	**160**	0.28*	**160**	1.1*	**160**	0.84*	**160**	5.3*	**160**	4.1*	**160**	7.3*	**160**	7.1*	-	-	170	0.01
soybean	**29**	0.01*	**29**	0.02*	**29**	0.01*	**29**	0.23*	**29**	0.17*	**29**	2.3*	**29**	2.4*	31	3098	47	0.00
tic-tac-toe	**216**	0.01*	**216**	0.02*	**216**	0.01*	**216**	0.11*	**216**	0.10*	**216**	1.8*	**216**	2.6*	232	1794	236	0.00
vote	**12**	0.02*	**12**	0.02*	**12**	0.02*	**12**	0.29*	**12**	0.32*	**12**	2.6*	**12**	2.7*	13	2763	14	0.00
winequality-red	**8**	0.02*	**8**	0.10*	**8**	0.11*	**8**	0.37*	**8**	0.23*	**8**	1.2*	**8**	3.5*	-	-	9	0.00
yeast	**403**	0.07*	**403**	0.34*	**403**	0.10*	**403**	6.1*	**403**	0.66*	**403**	7.7*	**403**	1.6*	434	2683	418	0.00

Table 13: Comparison with state of the art: $m < 100$, depth 3

	Blossom		Murtree		Murtree^P		DL8.5		DL8.5^P		CP		CP^P		binOCT		CART	
	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu
audiology	**5**	0.06*	**5**	0.13*	**5**	0.06*	**5**	4.1*	**5**	0.85*	**5**	9.1*	**5**	5.0*	6	508	6	0.00
australian-credit	**73**	0.14*	**73**	0.35*	**73**	0.12*	**73**	9.7*	**73**	1.8*	**73**	14*	**73**	4.6*	87	192	87	0.00
bank_conv	**428**	9.9*	**428**	16*	**428**	15*	**428**	112*	**428**	71*	**428**	73*	**428**	63*	-	-	438	0.02
biodeg	**164**	5.4*	**164**	12*	**164**	12*	**164**	141*	**164**	98*	**164**	90*	**164**	103*	-	-	184	0.01
breast-wisconsin	**15**	0.05*	**15**	0.20*	**15**	0.07*	**15**	5.6*	**15**	0.76*	**15**	11*	**15**	2.0*	18	1858	26	0.00
default_credit	**5327**	232*	**5327**	451*	**5327**	450*	**5327**	1730*	**5327**	1389*	**5327**	510*	**5327**	462*	-	-	5349	0.48
diabetes	**162**	0.09*	**162**	0.37*	**162**	0.12*	**162**	10*	**162**	1.00*	**162**	12*	**162**	4.6*	165	3501	177	0.00
forest-fires	**193**	20*	**193**	9.6*	**193**	4.3*	-	-	**193**	301*	**193**	2836*	**193**	895*	198	3501	198	0.01
german-credit	**236**	0.26*	**236**	0.38*	**236**	0.20*	**236**	9.4*	**236**	3.1*	**236**	13*	**236**	6.9*	244	2329	251	0.00
hand_posture	**7645**	181	**7645**	1134	**7645**	1110	10461	≥1h	10461	≥1h	8432	≥1h	8432	≥1h	-	-	8382	22
ionosphere	**22**	3.8*	**22**	12*	**22**	2.6*	**22**	397*	**22**	50*	**22**	460*	**22**	69*	27	3268	29	0.01
letter	**369**	10*	**369**	34*	**369**	11*	**369**	462*	**369**	44*	**369**	158*	**369**	19*	813	0.00	677	0.17
mnist_0	**2557**	1994*	**2557**	568*	**2557**	597*	3366	≥1h	3294	≥1h	**2557**	≥1h	**2557**	≥1h	-	-	3329	2.5
mushroom	**8**	0.79*	**8**	0.53*	**8**	0.40*	**8**	6.8*	**8**	3.8*	**8**	8.4*	**8**	5.6*	180	2728	280	0.02
pendigits	**47**	3.3*	**47**	11*	**47**	3.4*	**47**	126*	**47**	13*	**47**	70*	**47**	12*	477	2663	51	0.05
segment	**0**	0.03*	**0**	0.13*	**0**	0.04*	**0**	2.0*	**0**	0.27*	**0**	4.1*	**0**	1.8*	4	2865	5	0.01
spambase	**694**	11*	**694**	38*	**694**	35*	**694**	291*	**694**	286*	**694**	203*	**694**	211*	-	-	704	0.05
splice-1	**224**	9.8*	**224**	5.3*	**224**	4.8*	**224**	108*	**224**	90*	**224**	173*	**224**	145*	453	3502	279	0.03
Statlog_satellite	**187**	79*	**187**	118*	**187**	116*	**187**	998*	**187**	905*	**187**	703*	**187**	865*	-	-	345	0.08
Statlog_shuttle	**0**	6.7*	**0**	81*	**0**	70*	1	≥1h	1	≥1h	**0**	39*	**0**	39*	-	-	58	1.7
surgical-deepnet	**2512**	953	**2512**	3523	2524	605	-	-	2768	≥1h	**2512**	≥1h	**2512**	≥1h	-	-	2924	5.7
taiwan_binarised	**5326**	48*	**5326**	45*	**5326**	35*	**5326**	511*	**5326**	328*	**5326**	190*	**5326**	176*	6636	0.00	5346	0.26
titanic	**143**	6.7*	**143**	11*	**143**	9.6*	**143**	135*	**143**	107*	**143**	173*	**143**	182*	150	3362	148	0.01
vehicle	**26**	0.93*	**26**	2.2*	**26**	0.58*	**26**	63*	**26**	6.5*	**26**	66*	**26**	11*	42	3374	66	0.01
weather-aus	**1756**	14	**1756**	611	**1756**	687	-	-	-	-	**1756**	≥1h	-	-	-	-	1761	20
wine1	**43**	16*	**43**	9.0*	**43**	2.0*	-	-	**43**	200*	**43**	≥1h	**43**	703*	44	3507	45	0.00
wine2	**49**	17*	**49**	5.8*	**49**	1.5*	-	-	**49**	189*	**49**	≥1h	**49**	680*	57	3207	52	0.00
wine3	**33**	16*	**33**	8.4*	**33**	2.3*	-	-	**33**	228*	**33**	≥1h	**33**	649*	35	2814	35	0.00

Table 14: Comparison with state of the art: $m \geq 100$, depth 3

	Blossom		Murtree		Murtree^P		DL8.5		DL8.5^P		CP		CP^P		binOCT		CART	
	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu
adult_discretized	**4609**	14*	**4609**	30*	**4609**	15*	**4609**	271*	**4609**	93*	**4609**	246*	**4609**	156*	5659	3392	5022	0.06
anneal	**91**	1.5*	**91**	5.0*	**91**	1.5*	**91**	102*	**91**	9.6*	**91**	193*	**91**	30*	108	2954	135	0.00
balance-scale	**48**	0.04*	**48**	0.05*	**48**	0.05*	**48**	0.22*	**48**	0.16*	**48**	1.8*	**48**	2.2*	-	-	49	0.00
banknote	**13**	0.08*	**13**	0.27*	**13**	0.22*	**13**	0.78*	**13**	0.59*	**13**	4.2*	**13**	2.9*	-	-	38	0.00
breast-cancer	**16**	9.6*	**16**	2.9*	**16**	2.7*	**16**	28*	**16**	30*	**16**	219*	**16**	241*	22	2746	21	0.00
car	**136**	0.19*	**136**	0.16*	**136**	0.16*	**136**	0.36*	**136**	0.38*	**136**	2.8*	**136**	3.3*	178	871	178	0.00
car_evaluation	**130**	0.02*	**130**	0.07*	**130**	0.06*	**130**	0.13*	**130**	0.07*	**130**	1.3*	**130**	2.1*	-	-	**130**	0.00
chess	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.07*	**0**	0.14*	-	-	**0**	0.00
compas_discretized	**1954**	0.07*	**1954**	1.0*	**1954**	0.19*	**1954**	3.5*	**1954**	0.45*	**1954**	6.3*	**1954**	2.2*	1991	3390	1997	0.01
heart-cleveland	**25**	3.1*	**25**	4.8*	**25**	1.1*	**25**	154*	**25**	17*	**25**	391*	**25**	47*	37	2750	38	0.00
hepatitis	**3**	0.32*	**3**	0.73*	**3**	0.14*	**3**	28*	**3**	1.2*	**3**	70*	**3**	8.4*	11	510	12	0.00
HTRU_2	**385**	74*	**385**	122*	**385**	101*	**385**	450*	**385**	276*	**385**	295*	**385**	253*	-	-	409	0.05
hypothyroid	**53**	2.9*	**53**	16*	**53**	3.4*	**53**	181*	**53**	12*	**53**	254*	**53**	24*	55	3071	**53**	0.01
IndiansDiabetes	**149**	0.90*	**149**	1.3*	**149**	1.3*	**149**	7.3*	**149**	6.6*	**149**	16*	**149**	16*	-	-	166	0.00
iris	**1**	0.00*	**1**	0.00*	**1**	0.00*	**1**	0.00*	**1**	0.00*	**1**	0.92*	**1**	0.05*	-	-	**1**	0.00
kr-vs-kp	**144**	2.8*	**144**	6.9*	**144**	0.93*	**144**	88*	**144**	3.6*	**144**	141*	**144**	15*	189	2850	189	0.01
lymph	**3**	0.74*	**3**	0.63*	**3**	0.25*	**3**	14*	**3**	3.0*	**3**	64*	**3**	19*	7	2987	10	0.00
magic04	**3112**	232*	**3112**	328*	**3112**	308*	**3112**	1296*	**3112**	1021*	**3112**	800*	**3112**	701*	-	-	3350	0.07
messidor	**332**	21*	**332**	27*	**332**	25*	**332**	245*	**332**	219*	**332**	269*	**332**	318*	-	-	364	0.00
monk1	**2**	0.00*	**2**	0.01*	**2**	0.01*	**2**	0.01*	**2**	0.01*	**2**	1.5*	**2**	0.93*	-	-	11	0.00
monk2	**31**	0.01*	**31**	0.01*	**31**	0.01*	**31**	0.04*	**31**	0.03*	**31**	2.1*	**31**	0.39*	-	-	50	0.00
monk3	**4**	0.00*	**4**	0.01*	**4**	0.01*	**4**	0.01*	**4**	0.01*	**4**	1.0*	**4**	1.1*	-	-	5	0.00
primary-tumor	**34**	0.03*	**34**	0.11*	**34**	0.03*	**34**	2.0*	**34**	0.13*	**34**	5.6*	**34**	2.2*	38	3132	44	0.00
seismic_bumps	**148**	22*	**148**	56*	**148**	44*	**148**	290*	**148**	175*	**148**	303*	**148**	307*	-	-	158	0.01
soybean	**14**	0.62*	**14**	0.46*	**14**	0.31*	**14**	5.1*	**14**	2.8*	**14**	22*	**14**	15*	22	2906	32	0.00
tic-tac-toe	**137**	0.38*	**137**	0.26*	**137**	0.23*	**137**	1.8*	**137**	1.4*	**137**	7.2*	**137**	6.5*	162	2511	150	0.00
vote	**5**	1.2*	**5**	0.50*	**5**	0.48*	**5**	7.6*	**5**	6.8*	**5**	21*	**5**	26*	12	3311	8	0.00
winequality-red	**4**	0.62*	**4**	1.2*	**4**	0.94*	**4**	4.3*	**4**	2.4*	**4**	12*	**4**	11*	-	-	8	0.00
yeast	**366**	3.4*	**366**	18*	**366**	2.9*	**366**	257*	**366**	21*	**366**	386*	**366**	34*	438	888	394	0.01

Table 15: Comparison with state of the art: $m < 100$, depth 4

	Blossom		Murtree		Murtree^P		DL8.5		DL8.5^P		CP		CP^P		binOCT		CART	
	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu
audiology	**1**	4.0*	**1**	6.4*	**1**	1.6*	**1**	128*	**1**	20*	**1**	773*	**1**	113*	2	2687	3	0.00
australian-credit	**56**	10*	**56**	24*	**56**	4.7*	**56**	470*	**56**	77*	**56**	1170*	**56**	179*	83	3258	74	0.00
bank_conv	**392**	1963*	**392**	1651*	**392**	1621*	-	-	404	≥1h	**392**	≥1h	**392**	≥1h	-	-	408	0.04
biodeg	**128**	1511*	**128**	1436*	**128**	1323*	-	-	-	-	129	≥1h	129	≥1h	-	-	148	0.01
breast-wisconsin	**7**	3.1*	**7**	9.3*	**7**	1.8*	**7**	245*	**7**	24*	**7**	662*	**7**	58*	15	3460	16	0.00
default_credit	**5270**	209	**5270**	430	**5270**	430	5306	≥1h	5306	≥1h	**5270**	≥1h	**5270**	≥1h	-	-	5306	0.69
diabetes	**137**	5.7*	**137**	22*	**137**	3.5*	**137**	550*	**137**	35*	**137**	1001*	**137**	73*	180	2663	166	0.00
forest-fires	173	15	**171**	2907*	**171**	879*	-	-	**171**	≥1h	179	≥1h	173	≥1h	196	3356	186	0.01
german-credit	**204**	28*	**204**	27*	**204**	11*	**204**	423*	**204**	200*	**204**	1008*	**204**	453*	236	3306	231	0.00
hand_posture	**4896**	976	5778	1432	5778	1368	11021	≥1h	11021	≥1h	16265	≥1h	16265	≥1h	-	-	6098	27
ionosphere	**7**	730*	**7**	1683*	**7**	233*	-	-	**7**	≥1h	8	≥1h	8	≥1h	24	751	27	0.01
letter	**261**	1185*	**261**	2956*	**261**	540*	335	≥1h	**261**	2517*	**261**	≥1h	**261**	2076*	813	0.00	462	0.20
mnist_0	2173	2158	**1951**	3542	2077	3152	3319	≥1h	3294	≥1h	5923	≥1h	5923	≥1h	-	-	2311	3.8
mushroom	**0**	0.00*	**0**	0.03*	**0**	0.02*	**0**	41*	**0**	25*	**0**	0.07*	**0**	0.29*	192	3354	4	0.02
pendigits	**13**	230*	**13**	833*	**13**	129*	-	-	**13**	704*	14	≥1h	**13**	923*	780	0.00	25	0.07
segment	**0**	0.00*	**0**	0.02*	**0**	0.01*	**0**	1.6*	**0**	0.20*	**0**	2.5*	**0**	0.99*	1	3501	1	0.01
spambase	**590**	7.7	**590**	3295*	**590**	3117*	-	-	-	-	**590**	≥1h	**590**	≥1h	-	-	624	0.06
splice-1	**141**	3241*	**141**	644*	**141**	525*	-	-	212	≥1h	**141**	≥1h	**141**	≥1h	568	3416	**141**	0.03
Statlog_satellite	**111**	3571	116	1306	116	1322	-	-	187	≥1h	136	≥1h	136	≥1h	-	-	204	0.08
Statlog_shuttle	**0**	0.64*	**0**	41*	**0**	41*	1	≥1h	2	≥1h	**0**	42*	**0**	30*	-	-	36	2.4
surgical-deepnet	**2269**	49	2506	489	2506	498	-	-	2767	≥1h	3690	≥1h	3690	≥1h	-	-	2704	6.2
taiwan_binarised	**5273**	6.2	**5273**	37	**5273**	34	5307	≥1h	5307	≥1h	**5273**	≥1h	**5273**	≥1h	6521	75	5306	0.27
titanic	**119**	1604*	**119**	2104*	**119**	1874*	-	-	-	-	**119**	≥1h	**119**	≥1h	135	3501	134	0.01
vehicle	**12**	71*	**12**	172*	**12**	26*	-	-	**12**	427*	**12**	≥1h	**12**	919*	30	3410	28	0.01
weather-aus	**1749**	2525	1750	1243	1750	1252	-	-	-	-	1752	≥1h	-	-	-	-	1761	20
wine1	**37**	1674	**37**	1831*	**37**	265*	-	-	-	-	39	≥1h	39	≥1h	45	3506	42	0.01
wine2	**43**	17	**43**	1833*	**43**	218*	-	-	-	-	46	≥1h	**43**	≥1h	57	3232	47	0.01
wine3	**28**	33	**28**	2537*	**28**	320*	-	-	-	-	30	≥1h	30	≥1h	32	3388	32	0.01

Table 16: Comparison with state of the art: $m \geq 100$, depth 4

	Blossom		Murtree		Murtree^P		DL8.5		DL8.5^P		CP		CP^P		binOCT		CART	
	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu
adult_discretized	**4423**	725*	**4423**	794*	**4423**	420*	4442	≥1h	**4423**	2184*	**4423**	≥1h	**4423**	≥1h	7157	20	4728	0.08
anneal	**70**	44*	**70**	148*	**70**	22*	-	-	**70**	156*	75	≥1h	**70**	790*	101	2995	123	0.00
balance-scale	**45**	0.46*	**45**	0.44*	**45**	0.40*	**45**	1.0*	**45**	0.76*	**45**	7.9*	**45**	6.3*	-	-	49	0.00
banknote	**3**	0.88*	**3**	2.2*	**3**	1.4*	**3**	3.9*	**3**	2.2*	**3**	34*	**3**	27*	-	-	15	0.00
breast-cancer	**6**	725*	**6**	72*	**6**	72*	**6**	438*	**6**	478*	**6**	≥1h	**6**	≥1h	14	2894	16	0.00
car	**86**	2.4*	**86**	1.2*	**86**	1.2*	**86**	2.7*	**86**	2.5*	**86**	21*	**86**	27*	138	3379	106	0.01
car_evaluation	**90**	0.13*	**90**	0.37*	**90**	0.18*	**90**	0.49*	**90**	0.15*	**90**	4.9*	**90**	3.7*	-	-	116	0.00
chess	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.08*	**0**	0.04*	-	-	**0**	0.00
compas_discretized	**1919**	1.1*	**1919**	11*	**1919**	1.7*	**1919**	26*	**1919**	3.3*	**1919**	77*	**1919**	20*	1952	3153	1968	0.01
heart-cleveland	**7**	93*	**7**	101*	**7**	17*	-	-	**7**	236*	**7**	≥1h	**7**	1731*	26	3288	26	0.00
hepatitis	**0**	0.05*	**0**	0.18*	**0**	0.05*	**0**	71*	**0**	1.8*	**0**	12*	**0**	3.8*	6	3026	8	0.00
HTRU_2	**361**	98	**361**	2724*	**361**	2302*	369	≥1h	362	≥1h	**361**	≥1h	**361**	≥1h	-	-	394	0.06
hypothyroid	**44**	87*	**44**	343*	**44**	50*	-	-	**44**	172*	45	≥1h	**44**	684*	64	3324	50	0.01
IndiansDiabetes	**125**	30*	**125**	19*	**125**	17*	**125**	125*	**125**	122*	**125**	410*	**125**	375*	-	-	162	0.00
iris	**1**	0.00*	**1**	0.00*	**1**	0.00*	**1**	0.01*	**1**	0.00*	**1**	1.2*	**1**	0.06*	-	-	**1**	0.00
kr-vs-kp	**81**	65*	**81**	150*	**81**	13*	-	-	**81**	37*	**81**	≥1h	**81**	399*	189	3502	189	0.01
lymph	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	14*	**0**	2.6*	**0**	2.7*	**0**	2.8*	7	3380	4	0.00
magic04	2882	756	2882	873	2882	831	2910	≥1h	**2880**	≥1h	2882	≥1h	2882	≥1h	-	-	3179	0.11
messidor	**281**	1522*	**281**	855*	**281**	790*	-	-	**281**	≥1h	292	≥1h	292	≥1h	-	-	345	0.00
monk1	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.23*	**0**	0.10*	-	-	9	0.00
monk2	**15**	0.05*	**15**	0.05*	**15**	0.05*	**15**	0.09*	**15**	0.13*	**15**	2.8*	**15**	2.9*	-	-	32	0.00
monk3	**2**	0.03*	**2**	0.03*	**2**	0.02*	**2**	0.03*	**2**	0.02*	**2**	2.2*	**2**	2.7*	-	-	5	0.00
primary-tumor	**26**	0.38*	**26**	1.5*	**26**	0.23*	**26**	24*	**26**	0.41*	**26**	103*	**26**	9.2*	34	3255	35	0.00
seismic_bumps	**132**	1533*	**132**	1617*	**132**	1166*	-	-	133	≥1h	134	≥1h	134	≥1h	-	-	151	0.01
soybean	**8**	20*	**8**	7.6*	**8**	4.5*	**8**	63*	**8**	31*	**8**	752*	**8**	412*	14	3178	23	0.00
tic-tac-toe	**63**	10*	**63**	2.3*	**63**	2.3*	**63**	14*	**63**	11*	**63**	89*	**63**	110*	125	3052	78	0.00
vote	**1**	24*	**1**	6.1*	**1**	5.8*	**1**	45*	**1**	37*	**1**	522*	**1**	599*	8	1319	6	0.00
winequality-red	**3**	16*	**3**	18*	**3**	11*	**3**	39*	**3**	18*	**3**	232*	**3**	204*	-	-	6	0.00
yeast	**313**	139*	**313**	558*	**313**	53*	-	-	**313**	412*	315	≥1h	**313**	1120*	376	3456	367	0.01

Table 17: Comparison with state of the art: $m < 100$, depth 5

	Blossom		Murtree		Murtree^P		DL8.5		DL8.5^P		CP		CP^P		binOCT		CART	
	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu
audiology	**0**	0.00*	**0**	0.02*	**0**	0.01*	**0**	0.05*	**0**	0.02*	**0**	7.0*	**0**	3.8*	1	3083	2	0.00
australian-credit	**39**	658*	**39**	872*	**39**	116*	-	-	**39**	1779*	40	≥1h	40	≥1h	72	3282	64	0.00
bank_conv	**340**	1662	**340**	1636	**340**	1449	-	-	404	≥1h	521	≥1h	521	≥1h	-	-	379	0.04
biodeg	**88**	268	**88**	1141	**88**	1030	-	-	-	-	356	≥1h	356	≥1h	-	-	127	0.01
breast-wisconsin	**0**	20*	**0**	72*	**0**	9.1*	-	-	**0**	83*	1	≥1h	**0**	471*	16	3105	13	0.00
default_credit	**5181**	3202	5251	3121	5251	3130	5334	≥1h	5334	≥1h	6636	≥1h	6636	≥1h	-	-	5273	1.0
diabetes	**106**	312*	**106**	920*	**106**	78*	-	-	**106**	974*	107	≥1h	**106**	2841*	160	3501	141	0.00
forest-fires	156	777	**149**	2977	**149**	857	-	-	172	≥1h	172	≥1h	172	≥1h	207	3386	177	0.01
german-credit	**161**	2741*	**161**	973*	**161**	318*	-	-	**161**	≥1h	**161**	≥1h	**161**	≥1h	221	3504	209	0.01
hand_posture	**3154**	56	4482	1297	4482	1218	11736	≥1h	11736	≥1h	16265	≥1h	16265	≥1h	-	-	3377	42
ionosphere	**0**	506*	**0**	1340*	**0**	192*	-	-	-	-	4	≥1h	2	≥1h	25	3386	17	0.01
letter	**168**	3082	190	549	**168**	2091	352	≥1h	260	≥1h	813	≥1h	201	≥1h	813	0.00	335	0.32
mnist_0	**1714**	284	2066	2149	2066	2013	3319	≥1h	3319	≥1h	5923	≥1h	5923	≥1h	-	-	2021	4.5
mushroom	**0**	0.00*	**0**	0.03*	**0**	0.02*	**0**	36*	**0**	21*	**0**	0.10*	**0**	0.56*	1930	19	3	0.03
pendigits	**0**	284*	**0**	1295*	**0**	196*	-	-	2	≥1h	780	≥1h	**0**	3010*	751	30	11	0.07
segment	**0**	0.00*	**0**	0.02*	**0**	0.01*	**0**	1.0*	**0**	0.15*	**0**	2.0*	**0**	0.26*	41	2839	1	0.01
spambase	**501**	219	**501**	2340	**501**	2150	-	-	615	≥1h	1813	≥1h	1813	≥1h	-	-	571	0.05
splice-1	101	24	**100**	3308	**100**	2808	-	-	561	≥1h	1535	≥1h	1535	≥1h	814	16	117	0.04
Statlog_satellite	**71**	279	98	638	98	604	-	-	187	≥1h	1072	≥1h	1072	≥1h	-	-	128	0.13
Statlog_shuttle	**0**	0.06*	**0**	39*	**0**	38*	1	≥1h	1	≥1h	**0**	34*	**0**	23*	-	-	10	2.8
surgical-deepnet	**2131**	2168	2337	400	2337	381	-	-	2767	≥1h	3690	≥1h	3690	≥1h	-	-	2245	8.4
taiwan_binarised	**5200**	105	5261	38	5261	32	5412	≥1h	5412	≥1h	6636	≥1h	6636	≥1h	6636	0.00	5280	0.37
titanic	**95**	1428	**95**	1371	**95**	1074	-	-	-	-	342	≥1h	342	≥1h	149	3505	130	0.01
vehicle	**1**	690	**1**	1540	**1**	912*	-	-	-	-	218	≥1h	3	≥1h	85	3502	23	0.01
weather-aus	**1735**	419	**1735**	1907	**1735**	1980	-	-	-	-	1761	≥1h	-	-	-	-	1751	26
wine1	**33**	1154	**33**	287	**33**	76	-	-	-	-	38	≥1h	36	≥1h	46	2910	39	0.01
wine2	39	411	**37**	3400	**37**	438	-	-	-	-	42	≥1h	40	≥1h	50	3197	44	0.01
wine3	25	17	25	25	**23**	971	-	-	-	-	28	≥1h	27	≥1h	37	3288	30	0.01

Table 18: Comparison with state of the art: $m \geq 100$, depth 5

	Blossom		Murtree		Murtree^P		DL8.5		DL8.5^P		CP		CP^P		binOCT		CART	
	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu
adult_discretized	**4191**	534	4294	2016	4204	397	4998	≥1h	4539	≥1h	7511	≥1h	7511	≥1h	7511	0.00	4481	0.09
anneal	41	3036	50	836	**36**	1579*	-	-	-	-	187	≥1h	187	≥1h	106	3386	96	0.00
balance-scale	**29**	37*	**29**	17*	**29**	16*	**29**	10*	**29**	8.6*	**29**	228*	**29**	279*	-	-	49	0.00
banknote	**2**	0.00*	**2**	69*	**2**	0.01*	**2**	80*	**2**	0.01*	**2**	≥1h	**2**	0.12*	-	-	5	0.00
breast-cancer	**0**	1007*	**0**	150*	**0**	146*	**0**	450*	**0**	523*	1	≥1h	1	≥1h	19	3313	8	0.00
car	**11**	231*	**11**	27*	**11**	25*	**11**	16*	**11**	17*	**11**	1678*	**11**	1978*	80	3495	50	0.00
car_evaluation	**80**	0.00*	**80**	7.4*	**80**	0.00*	**80**	4.2*	**80**	0.01*	**80**	123*	**80**	0.16*	-	-	**80**	0.00
chess	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.13*	**0**	0.08*	-	-	**0**	0.00
compas_discretized	**1852**	198*	**1852**	569*	**1852**	37*	**1852**	575*	**1852**	34*	1857	≥1h	**1852**	2014*	1940	3504	1941	0.01
heart-cleveland	**0**	0.00*	**0**	0.04*	**0**	0.02*	-	-	**0**	174*	**0**	3.0*	**0**	2.7*	17	3368	6	0.01
hepatitis	**0**	0.00*	**0**	0.01*	**0**	0.00*	**0**	8.9*	**0**	0.46*	**0**	0.49*	**0**	0.56*	1	3436	1	0.00
HTRU_2	297	3334	**293**	2992	**293**	1737	601	≥1h	599	≥1h	1639	≥1h	1639	≥1h	-	-	352	0.08
hypothyroid	**22**	3478	23	590	**22**	2548*	-	-	24	≥1h	277	≥1h	23	≥1h	277	274	42	0.01
IndiansDiabetes	**44**	3343	**44**	1355*	**44**	1138*	-	-	**44**	≥1h	268	≥1h	268	≥1h	-	-	113	0.00
iris	**1**	0.00*	**1**	0.00*	**1**	0.00*	**1**	0.01*	**1**	0.00*	**1**	2.6*	**1**	0.07*	-	-	**1**	0.00
kr-vs-kp	18	2550	21	2051	**17**	1063*	-	-	**17**	1733*	37	≥1h	34	≥1h	1096	43	103	0.01
lymph	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.01*	**0**	0.00*	**0**	0.24*	**0**	0.32*	1	3431	**0**	0.00
magic04	**2488**	2773	2851	1512	2851	1414	3140	≥1h	3129	≥1h	6688	≥1h	6688	≥1h	-	-	2768	0.11
messidor	**179**	2456	203	1842	203	1354	-	-	248	≥1h	540	≥1h	540	≥1h	-	-	305	0.01
monk1	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.17*	**0**	0.08*	-	-	8	0.00
monk2	**0**	0.00*	**0**	0.02*	**0**	0.02*	**0**	0.00*	**0**	0.00*	**0**	0.78*	**0**	0.30*	-	-	5	0.00
monk3	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.45*	**0**	0.09*	-	-	2	0.00
primary-tumor	**16**	18*	**16**	162*	**16**	2.1*	**16**	458*	**16**	0.94*	**16**	≥1h	**16**	410*	24	3432	26	0.00
seismic_bumps	**76**	2389	97	193	82	3565	-	-	129	≥1h	170	≥1h	170	≥1h	-	-	137	0.01
soybean	**2**	19*	**2**	1108*	**2**	6.9*	-	-	**2**	164*	3	≥1h	3	≥1h	13	1579	11	0.00
tic-tac-toe	**0**	32*	**0**	8.4*	**0**	8.2*	**0**	29*	**0**	33*	**0**	764*	**0**	761*	46	3449	22	0.00
vote	**0**	0.00*	**0**	0.00*	**0**	0.01*	**0**	0.17*	**0**	0.14*	**0**	3.2*	**0**	3.7*	2	3348	2	0.00
winequality-red	**2**	0.01*	**2**	1131*	**2**	0.02*	-	-	**2**	0.04*	10	≥1h	**2**	0.93*	-	-	4	0.00
yeast	182	3558	222	1088	**165**	2224	-	-	260	≥1h	463	≥1h	215	≥1h	455	1968	306	0.02

Table 19: Comparison with state of the art: $m < 100$, depth 7

	Blossom		Murtree		Murtree^P		DL8.5		DL8.5^P		CP		CP^P		binOCT		CART	
	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu
audiology	**0**	0.00*	**0**	0.01*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.18*	**0**	0.25*	3	2177	**0**	0.00
australian-credit	**0**	101*	**0**	126*	**0**	30*	-	-	3	≥1h	296	≥1h	296	≥1h	85	3320	43	0.01
bank_conv	**220**	1642	319	1174	319	1090	-	-	405	≥1h	521	≥1h	521	≥1h	-	-	303	0.06
biodeg	**26**	2775	95	876	71	3544	-	-	-	-	356	≥1h	356	≥1h	-	-	86	0.02
breast-wisconsin	**0**	0.02*	**0**	0.09*	**0**	0.02*	-	-	**0**	38*	**0**	2805*	**0**	111*	12	3502	4	0.00
default_credit	**4935**	222	5237	223	5237	222	5412	≥1h	5412	≥1h	6636	≥1h	6636	≥1h	-	-	5153	1.0
diabetes	21	827	81	2000	**12**	498	-	-	86	≥1h	268	≥1h	268	≥1h	179	1988	100	0.01
forest-fires	146	125	139	2254	**138**	2715	-	-	-	-	247	≥1h	247	≥1h	270	0.00	161	0.02
german-credit	**56**	1192	87	341	86	2385	-	-	171	≥1h	300	≥1h	300	≥1h	246	2598	150	0.01
hand_posture	**749**	2684	1418	461	1418	413	14236	≥1h	14236	≥1h	16265	≥1h	16265	≥1h	-	-	962	78
ionosphere	**0**	0.07*	**0**	0.37*	**0**	0.10*	-	-	**0**	63*	**0**	566*	**0**	53*	61	213	7	0.01
letter	68	177	131	86	**66**	1473	488	≥1h	324	≥1h	813	≥1h	813	≥1h	-	-	153	0.31
mnist_0	**1107**	2895	1601	964	1601	948	3296	≥1h	3364	≥1h	5923	≥1h	5923	≥1h	-	-	1323	8.5
mushroom	**0**	0.00*	**0**	0.02*	**0**	0.03*	**0**	10*	**0**	6.6*	**0**	0.15*	**0**	0.35*	4208	0.00	**0**	0.03
pendigits	**0**	0.00*	**0**	0.15*	**0**	0.06*	-	-	**0**	1125*	**0**	8.1*	**0**	2.3*	780	0.00	1	0.07
segment	**0**	0.00*	**0**	0.02*	**0**	0.01*	**0**	0.23*	**0**	0.06*	**0**	0.28*	**0**	0.21*	330	0.00	**0**	0.01
spambase	**352**	3562	495	2265	495	2185	-	-	966	≥1h	1813	≥1h	1813	≥1h	-	-	462	0.08
splice-1	**29**	3484	47	881	47	775	-	-	575	≥1h	1535	≥1h	1535	≥1h	1655	0.00	58	0.05
Statlog_satellite	**14**	2428	89	2540	89	2419	-	-	185	≥1h	1072	≥1h	1072	≥1h	-	-	41	0.12
Statlog_shuttle	**0**	0.04*	**0**	42*	**0**	40*	**0**	3163*	**0**	2500*	**0**	14*	**0**	12*	-	-	4	2.8
surgical-deepnet	**1647**	1248	1890	655	1885	3540	-	-	2767	≥1h	3690	≥1h	3690	≥1h	-	-	1871	9.9
taiwan_binarised	**4896**	1958	5189	3125	5186	2596	5412	≥1h	5412	≥1h	6636	≥1h	6636	≥1h	-	-	5161	0.58
titanic	**72**	442	97	149	97	141	-	-	-	-	342	≥1h	342	≥1h	342	0.00	111	0.01
vehicle	**0**	0.09*	**0**	0.47*	**0**	0.14*	-	-	**0**	90*	**0**	1178*	**0**	88*	210	25	4	0.01
weather-aus	**1685**	2048	1724	3257	1724	3415	-	-	-	-	1761	≥1h	-	-	-	-	1721	27
wine1	28	892	28	325	**27**	2628	-	-	40	≥1h	36	≥1h	35	≥1h	57	122	33	0.01
wine2	31	28	31	25	**30**	1011	-	-	-	-	35	≥1h	32	≥1h	71	0.00	38	0.01
wine3	21	524	20	2925	**17**	2248	-	-	-	-	24	≥1h	23	≥1h	47	142	24	0.01

Table 20: Comparison with state of the art: $m \geq 100$, depth 7

	Blossom		Murtree		Murtree^P		DL8.5		DL8.5^P		CP		CP^P		binOCT		CART	
	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu
adult_discretized	**3841**	2632	4052	1695	4010	2583	6200	≥1h	6200	≥1h	7511	≥1h	7511	≥1h	-	-	4148	0.12
anneal	**34**	23*	39	225	**34**	14*	-	-	**34**	4.0*	187	≥1h	187	≥1h	625	0.00	59	0.00
balance-scale	**0**	19*	**0**	23*	**0**	21*	**0**	1.5*	**0**	1.2*	**0**	16*	**0**	17*	-	-	6	0.00
banknote	**2**	0.00*	**2**	1123*	**2**	0.00*	**2**	738*	**2**	0.00*	610	≥1h	**2**	1.1*	-	-	**2**	0.00
breast-cancer	**0**	0.00*	**0**	0.01*	**0**	0.01*	**0**	0.00*	**0**	0.00*	**0**	2.4*	**0**	2.4*	239	0.00	**0**	0.00
car	**0**	0.26*	**0**	0.48*	**0**	0.45*	**0**	0.03*	**0**	0.03*	**0**	3.3*	**0**	3.2*	518	0.00	11	0.00
car_evaluation	**80**	0.00*	**80**	112*	**80**	0.00*	**80**	9.2*	**80**	0.00*	**80**	≥1h	**80**	0.50*	-	-	**80**	0.00
chess	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.01*	**0**	0.01*	**0**	0.66*	**0**	0.31*	-	-	**0**	0.00
compas_discretized	**1828**	0.73*	1842	2465	**1828**	4.5*	-	-	**1828**	0.52*	2809	≥1h	**1828**	372*	2809	0.00	1871	0.01
heart-cleveland	**0**	0.00*	**0**	0.02*	**0**	0.01*	**0**	0.08*	**0**	0.02*	**0**	1.2*	**0**	0.73*	127	7.6	**0**	0.00
hepatitis	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	1.3*	**0**	1.3*	19	2032	**0**	0.00
HTRU_2	**219**	550	299	1817	254	≥1h	669	≥1h	652	≥1h	1639	≥1h	1639	≥1h	-	-	293	0.08
hypothyroid	**17**	0.96*	**17**	517	**17**	1.3*	-	-	**17**	2439*	277	≥1h	**17**	98*	-	-	31	0.01
IndiansDiabetes	**8**	4.7*	44	1200	**8**	15*	-	-	**8**	2733*	268	≥1h	**8**	3047*	-	-	63	0.00
iris	**1**	0.00*	**1**	0.00*	**1**	0.00*	**1**	0.01*	**1**	0.00*	**1**	21*	**1**	0.83*	-	-	**1**	0.00
kr-vs-kp	**0**	1897*	24	711	**0**	282*	-	-	171	≥1h	784	≥1h	784	≥1h	-	-	12	0.01
lymph	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	1.2*	**0**	1.3*	30	576	**0**	0.00
magic04	**1635**	2746	2429	2759	2426	2710	3839	≥1h	3839	≥1h	6688	≥1h	6688	≥1h	-	-	2145	0.13
messidor	**66**	604	168	456	164	426	-	-	356	≥1h	540	≥1h	540	≥1h	-	-	211	0.03
monk1	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.64*	**0**	1.0*	-	-	**0**	0.00
monk2	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.89*	**0**	0.97*	-	-	**0**	0.00
monk3	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.99*	**0**	1.0*	-	-	**0**	0.00
primary-tumor	**15**	0.00*	**15**	1820	**15**	0.01*	-	-	**15**	0.00*	82	≥1h	**15**	1.3*	31	3329	20	0.00
seismic_bumps	38	2591	88	3021	**33**	2374*	-	-	-	-	170	≥1h	170	≥1h	-	-	101	0.01
soybean	**2**	0.00*	**2**	18	**2**	0.01*	-	-	**2**	0.11*	92	≥1h	**2**	2.6*	84	11	**2**	0.00
tic-tac-toe	**0**	0.00*	**0**	0.01*	**0**	0.01*	**0**	0.03*	**0**	0.04*	**0**	0.81*	**0**	1.4*	332	194	6	0.00
vote	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	2.3*	**0**	3.2*	132	9.9	**0**	0.00
winequality-red	**2**	0.00*	**2**	0.10	**2**	0.00*	-	-	**2**	0.00*	10	≥1h	**2**	1.3*	-	-	**2**	0.00
yeast	**28**	1008	170	962	108	2755	-	-	254	≥1h	463	≥1h	319	≥1h	463	0.00	185	0.01

Table 21: Comparison with state of the art: $m < 100$, depth 10

	Blossom		Murtree		Murtree^P		DL8.5		DL8.5^P		CP		CP^P		binOCT		CART	
	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu	error	cpu
audiology	**0**	0.00*	**0**	0.01*	**0**	0.00*	**0**	0.00*	**0**	0.00*	**0**	1.4*	**0**	1.3*	25	17	**0**	0.00
australian-credit	**0**	0.04*	**0**	0.27*	**0**	0.12*	-	-	**0**	487*	**0**	464*	**0**	47*	357	0.00	12	0.01
bank_conv	**169**	2794	223	2607	223	2315	-	-	406	≥1h	521	≥1h	521	≥1h	-	-	207	0.10
biodeg	**1**	1169*	24	1652	**1**	1059*	-	-	**1**	28*	356	≥1h	356	≥1h	-	-	27	0.02
breast-wisconsin	**0**	0.00*	**0**	0.01*	**0**	0.01*	**0**	3.4*	**0**	0.21*	**0**	7.8*	**0**	1.7*	444	0.00	**0**	0.00
default_credit	**4547**	2019	5046	240	5048	241	5412	≥1h	5412	≥1h	6636	≥1h	6636	≥1h	-	-	4762	1.3
diabetes	**0**	0.67*	**0**	3.7*	**0**	0.88*	-	-	124	≥1h	**0**	463*	**0**	22*	500	0.00	35	0.01
forest-fires	**113**	942	119	458	119	165	-	-	200	≥1h	247	≥1h	247	≥1h	-	-	146	0.02
german-credit	**0**	69*	**0**	74*	**0**	40*	-	-	**0**	2852*	**0**	28*	**0**	11*	700	0.00	66	0.01
hand_posture	**334**	39	450	426	450	407	15187	≥1h	15187	≥1h	16265	≥1h	16265	≥1h	-	-	530	88
ionosphere	**0**	0.00*	**0**	0.14*	**0**	0.04*	**0**	110*	**0**	9.8*	**0**	8.1*	**0**	2.9*	225	0.00	**0**	0.01
letter	**0**	79*	**0**	278*	**0**	44*	725	≥1h	644	≥1h	813	≥1h	813	≥1h	-	-	21	0.31
mnist_0	**383**	413	880	101	822	3583	3314	≥1h	3287	≥1h	5923	≥1h	5923	≥1h	-	-	477	8.5
mushroom	**0**	0.00*	**0**	0.02*	**0**	0.02*	**0**	1.1*	**0**	0.74*	**0**	1.2*	**0**	1.5*	-	-	**0**	0.04
pendigits	**0**	0.00*	**0**	0.11*	**0**	0.05*	**0**	1247*	**0**	67*	**0**	5.3*	**0**	2.3*	-	-	**0**	0.07
segment	**0**	0.00*	**0**	0.01*	**0**	0.01*	**0**	0.08*	**0**	0.02*	**0**	1.9*	**0**	1.8*	-	-	**0**	0.01
spambase	**262**	546	381	440	377	3008	-	-	864	≥1h	1813	≥1h	1813	≥1h	-	-	332	0.09
splice-1	5	1160	13	993	13	1254	-	-	575	≥1h	1535	≥1h	**1**	121*	-	-	12	0.05
Statlog_satellite	**3**	219	11	3349	11	3200	-	-	182	≥1h	1072	≥1h	1072	≥1h	-	-	15	0.13
Statlog_shuttle	**0**	0.02*	**0**	24*	**0**	22*	**0**	99*	**0**	70*	**0**	16*	**0**	11*	-	-	**0**	3.6
surgical-deepnet	**965**	2865	1382	885	1382	902	-	-	2766	≥1h	3690	≥1h	3690	≥1h	-	-	1089	14
taiwan_binarised	**4217**	1001	4993	1437	5024	1421	-	-	5412	≥1h	6636	≥1h	6636	≥1h	-	-	4710	0.54
titanic	**35**	3059	77	1852	75	1402	-	-	-	-	342	≥1h	342	≥1h	-	-	78	0.01
vehicle	**0**	0.00*	**0**	0.08*	**0**	0.03*	**0**	0.37*	**0**	0.06*	**0**	4.2*	**0**	2.1*	-	-	**0**	0.01
weather-aus	**1601**	2591	1675	2076	1675	1821	-	-	-	-	1761	≥1h	-	-	-	-	1642	32
wine1	22	545	22	319	**21**	778	-	-	-	-	27	≥1h	27	≥1h	-	-	25	0.01
wine2	24	399	24	2440	**23**	1714	-	-	-	-	29	≥1h	29	≥1h	-	-	29	0.02
wine3	16	272	**15**	151	**15**	40	-	-	32	≥1h	19	≥1h	18	≥1h	-	-	19	0.01

Table 22: Comparison with state of the art: $m \geq 100$, depth 10

	Blossom		No heuristic		No preprocessing		No lower bound	
	error	cpu	error	cpu	error	cpu	error	cpu
adult_discretized	5020	0.43^*	5020	0.25^*	5020	0.72^*	5020	0.27^*
anneal	112	0.03^*	112	0.02^*	112	0.17^*	112	0.03^*
balance-scale	49	0.00^*	49	0.00^*	49	0.00^*	49	0.00^*
banknote	36	0.01^*	36	0.00^*	36	0.01^*	36	0.01^*
breast-cancer	24	0.16^*	24	0.08^*	24	0.10^*	24	0.10^*
car	192	0.01^*	192	0.00^*	192	0.00^*	192	0.01^*
car_evaluation	202	0.00^*	202	0.00^*	202	0.00^*	202	0.01^*
chess	0	0.00^*	0	0.00^*	0	0.00^*	0	0.00^*
compas_discretized	2004	0.00^*	2004	0.00^*	2004	0.03^*	2004	0.00^*
heart-cleveland	41	0.05^*	41	0.03^*	41	0.20^*	41	0.05^*
hepatitis	10	0.00^*	10	0.00^*	10	0.07^*	10	0.01^*
HTRU_2	401	1.2^*	401	1.2^*	401	2.0^*	401	1.4^*
hypothyroid	61	0.07^*	61	0.07^*	61	0.33^*	61	0.08^*
IndiansDiabetes	166	0.02^*	166	0.02^*	166	0.03^*	166	0.03^*
iris	1	0.00^*	1	0.00^*	1	0.00^*	1	0.00^*
kr-vs-kp	198	0.09^*	198	0.06^*	198	0.22^*	198	0.07^*
lymph	12	0.01^*	12	0.01^*	12	0.06^*	12	0.02^*
magic04	3446	3.8^*	3446	3.5^*	3446	3.3^*	3446	3.2^*
messidor	366	0.25^*	366	0.23^*	366	0.26^*	366	0.28^*
monk1	11	0.00^*	11	0.00^*	11	0.00^*	11	0.00^*
monk2	42	0.00^*	42	0.00^*	42	0.00^*	42	0.00^*
monk3	6	0.00^*	6	0.00^*	6	0.00^*	6	0.00^*
primary-tumor	46	0.00^*	46	0.00^*	46	0.01^*	46	0.00^*
seismic_bumps	160	0.28^*	160	0.28^*	160	0.41^*	160	0.32^*
soybean	29	0.01^*	29	0.01^*	29	0.03^*	29	0.02^*
tic-tac-toe	216	0.01^*	216	0.01^*	216	0.01^*	216	0.01^*
vote	12	0.02^*	12	0.02^*	12	0.03^*	12	0.03^*
winequality-red	8	0.02^*	8	0.02^*	8	0.03^*	8	0.03^*
yeast	403	0.07^*	403	0.07^*	403	0.36^*	403	0.07^*

Table 23: Factor analysis: $m < 100$, depth 3

	Blossom		No heuristic		No preprocessing		No lower bound	
	error	cpu	error	cpu	error	cpu	error	cpu
audiology	5	0.06^*	5	0.04^*	5	0.33^*	5	0.06^*
australian-credit	73	0.14^*	73	0.11^*	73	0.55^*	73	0.13^*
bank	4453	259	4383	84	4453	226	4453	257
bank_conv	428	9.9^*	428	5.6^*	428	6.0^*	428	6.3^*
biodeg	164	5.4^*	164	4.6^*	164	5.7^*	164	6.1^*
breast-wisconsin	15	0.05^*	15	0.04^*	15	0.28^*	15	0.06^*
default_credit	5327	232^*	5327	289^*	5327	256^*	5327	220^*
diabetes	162	0.09^*	162	0.08^*	162	0.50^*	162	0.09^*
forest-fires	193	20^*	193	16^*	193	65^*	193	20^*
german-credit	236	0.26^*	236	0.20^*	236	0.54^*	236	0.26^*
hand_posture	7645	181	9132	189	7645	153	7645	172
ionosphere	22	3.8^*	22	3.0^*	22	22^*	22	4.2^*
letter	369	10^*	369	8.4^*	369	45^*	369	8.2^*
mnist_0	2557	1994^*	2557	1832^*	2557	1792^*	2557	1867^*
mushroom	8	0.79^*	8	0.60^*	8	0.76^*	8	0.68^*
pendigits	47	3.3^*	47	3.1^*	47	13^*	47	3.6^*
segment	0	0.03^*	0	0.03^*	0	0.20^*	0	0.03^*
spambase	694	11^*	694	9.6^*	694	12^*	694	11^*
splice-1	224	9.8^*	224	8.2^*	224	11^*	224	9.8^*
Statlog_satellite	187	79^*	187	63^*	187	67^*	187	85^*
Statlog_shuttle	0	6.7^*	0	84^*	0	13^*	0	6.2^*
surgical-deepnet	2512	953	2524	1304	2512	907	2512	918
taiwan_binarised	5326	48^*	5326	28^*	5326	45^*	5326	33^*
titanic	143	6.7^*	143	5.5^*	143	6.6^*	143	6.7^*
vehicle	26	0.93^*	26	0.59^*	26	3.5^*	26	0.83^*
weather-aus	1756	14	1756	1.3	1756	12	1756	13
wine1	43	16^*	43	14^*	43	120^*	43	17^*
wine2	49	17^*	49	14^*	49	118^*	49	17^*
wine3	33	16^*	33	13^*	33	118^*	33	16^*

Table 24: Factor analysis: $m \geq 100$, depth 3

	Blossom		No heuristic		No preprocessing		No lower bound	
	error	cpu	error	cpu	error	cpu	error	cpu
adult_discretized	4609	14*	4609	14*	4609	43*	4609	14*
anneal	91	1.5*	91	1.0*	91	11*	91	1.3*
balance-scale	48	0.04*	48	0.04*	48	0.04*	48	0.04*
banknote	13	0.08*	13	0.07*	13	0.19*	13	0.10*
breast-cancer	16	9.6*	16	7.6*	16	9.1*	16	8.9*
car	136	0.19*	136	0.16*	136	0.14*	136	0.16*
car_evaluation	130	0.02*	130	0.02*	130	0.04*	130	0.03*
chess	0	0.00*	0	0.00*	0	0.00*	0	0.00*
compas_discretized	1954	0.07*	1954	0.05*	1954	0.69*	1954	0.07*
heart-cleveland	25	3.1*	25	2.3*	25	19*	25	3.3*
hepatitis	3	0.32*	3	0.20*	3	3.0*	3	0.31*
HTRU_2	385	74*	385	77*	385	106*	385	75*
hypothyroid	53	2.9*	53	2.5*	53	23*	53	3.1*
IndiansDiabetes	149	0.90*	149	0.89*	149	0.97*	149	0.98*
iris	1	0.00*	1	0.00*	1	0.00*	1	0.00*
kr-vs-kp	144	2.8*	144	2.4*	144	14*	144	2.5*
lymph	3	0.74*	3	0.57*	3	2.4*	3	0.91*
magic04	3112	232*	3112	259*	3112	290*	3112	265*
messidor	332	21*	332	19*	332	21*	332	23*
monk1	2	0.00*	2	0.00*	2	0.00*	2	0.00*
monk2	31	0.01*	31	0.01*	31	0.01*	31	0.01*
monk3	4	0.00*	4	0.00*	4	0.00*	4	0.00*
primary-tumor	34	0.03*	34	0.02*	34	0.22*	34	0.03*
seismic_bumps	148	22*	148	20*	148	31*	148	25*
soybean	14	0.62*	14	0.50*	14	1.1*	14	0.71*
tic-tac-toe	137	0.38*	137	0.34*	137	0.38*	137	0.38*
vote	5	1.2*	5	0.91*	5	1.2*	5	1.4*
winequality-red	4	0.62*	4	0.62*	4	0.96*	4	0.85*
yeast	366	3.4*	366	3.0*	366	29*	366	3.4*

Table 25: Factor analysis: $m < 100$, depth 4

	Blossom		No heuristic		No preprocessing		No lower bound	
	error	cpu	error	cpu	error	cpu	error	cpu
audiology	1	4.0*	1	3.2*	1	29*	1	4.5*
australian-credit	56	10*	56	8.5*	56	68*	56	11*
bank	4314	290	4326	1102	4314	258	4314	308
bank_conv	392	1963*	392	1150*	392	1288*	392	1291*
biodeg	128	1511*	128	1153*	128	1447*	128	1620*
breast-wisconsin	7	3.1*	7	2.1*	7	33*	7	3.4*
default_credit	5270	209	5270	183	5270	167	5270	199
diabetes	137	5.7*	137	4.8*	137	59*	137	6.0*
forest-fires	173	15	173	11	173	48	173	15
german-credit	204	28*	204	22*	204	66*	204	29*
hand_posture	4896	976	10096	28	4896	872	4896	969
ionosphere	7	730*	7	548*	8	55	7	1026*
letter	261	1185*	261	813*	261	292	261	1407*
mnist_0	2173	2158	2229	3292	2173	1844	2173	2444
mushroom	0	0.00*	0	0.00*	0	0.00*	0	0.00*
pendigits	13	230*	13	237*	13	1871*	13	341*
segment	0	0.00*	0	0.00*	0	0.00*	0	0.00*
spambase	590	7.7	590	3504*	590	7.5	590	7.5
splice-1	141	3241*	141	2519*	141	0.00	141	3563*
Statlog_satellite	111	3571	120	1141	111	3479	114	3294
Statlog_shuttle	0	0.64*	0	0.79*	0	1.6*	0	0.83*
surgical-deepnet	2269	49	2414	1479	2269	46	2269	51
taiwan_binarised	5273	6.2	5273	39	5273	6.2	5273	7.1
titanic	119	1604*	119	1318*	119	1620*	119	1722*
vehicle	12	71*	12	60*	12	706*	12	91*
weather-aus	1749	2525	1750	2646	1749	2142	1749	2638
wine1	37	1674	37	1808	38	2248	37	1617
wine2	43	17	43	0.02	43	110	43	16
wine3	28	33	28	190	28	222	28	33

Table 26: Factor analysis: $m \geq 100$, depth 4

	Blossom		No heuristic		No preprocessing		No lower bound	
	error	cpu	error	cpu	error	cpu	error	cpu
adult_discretized	4423	725*	4423	693*	4423	2388*	4423	755*
anneal	70	44*	70	38*	70	736*	70	50*
balance-scale	45	0.46*	45	0.47*	45	0.50*	45	0.51*
banknote	3	0.88*	3	0.87*	3	2.4*	3	1.6*
breast-cancer	6	725*	6	604*	6	764*	6	764*
car	86	2.4*	86	2.5*	86	2.5*	86	2.9*
car_evaluation	90	0.13*	90	0.13*	90	0.40*	90	0.23*
chess	0	0.00*	0	0.00*	0	0.00*	0	0.00*
compas_discretized	1919	1.1*	1919	1.1*	1919	14*	1919	1.3*
heart-cleveland	7	93*	7	78*	7	1224*	7	156*
hepatitis	0	0.05*	0	0.09*	0	0.44*	0	0.05*
HTRU_2	361	98	361	3301*	361	92	361	73
hypothyroid	44	87*	44	85*	44	1539*	44	103*
IndiansDiabetes	125	30*	125	28*	125	31*	125	37*
iris	1	0.00*	1	0.00*	1	0.00*	1	0.00*
kr-vs-kp	81	65*	81	65*	81	823*	81	81*
lymph	0	0.00*	0	0.02*	0	0.00*	0	0.00*
magic04	2882	756	2882	91	2867	3184	2867	3455
messidor	281	1522*	281	1373*	281	1527*	281	1719*
monk1	0	0.00*	0	0.00*	0	0.00*	0	0.00*
monk2	15	0.05*	15	0.05*	15	0.05*	15	0.07*
monk3	2	0.03*	2	0.02*	2	0.02*	2	0.03*
primary-tumor	26	0.38*	26	0.43*	26	6.7*	26	0.46*
seismic_bumps	132	1533*	132	1243*	132	1914*	132	1708*
soybean	8	20*	8	16*	8	40*	8	26*
tic-tac-toe	63	10*	63	8.7*	63	9.3*	63	11*
vote	1	24*	1	21*	1	26*	1	45*
winequality-red	3	16*	3	15*	3	28*	3	24*
yeast	313	139*	313	123*	313	2348*	313	151*

Table 27: Factor analysis: $m < 100$, depth 5

	Blossom		No heuristic		No preprocessing		No lower bound	
	error	cpu	error	cpu	error	cpu	error	cpu
audiology	0	0.00*	0	0.00*	0	0.00*	0	0.00*
australian-credit	39	658*	39	513*	40	40	39	839*
bank	4187	1152	4309	1113	4187	1073	4187	1205
bank_conv	340	1662	346	2252	340	1059	340	984
biodeg	88	268	88	680	88	271	88	323
breast-wisconsin	0	20*	0	16*	0	478*	0	31*
default_credit	5181	3202	5183	3381	5181	1269	5181	1411
diabetes	106	312*	106	245*	106	1425	106	357*
forest-fires	156	777	157	61	156	2891	156	760
german-credit	161	2741*	161	2037*	161	82	161	2885*
hand_posture	3154	56	9813	11	3154	56	3154	58
ionosphere	0	506*	0	444*	2	1746	0	806*
letter	168	3082	172	2110	192	208	173	2313
mnist_0	1714	284	2075	1862	1714	241	1714	300
mushroom	0	0.00*	0	0.00*	0	0.00*	0	0.00*
pendigits	0	284*	0	725*	2	55	0	447*
segment	0	0.00*	0	0.00*	0	0.00*	0	0.00*
spambase	501	219	501	935	501	274	501	234
splice-1	101	24	101	1861	101	26	101	26
Statlog_satellite	71	279	99	1003	71	255	71	286
Statlog_shuttle	0	0.06*	0	85*	0	0.10*	0	0.06*
surgical-deepnet	2131	2168	2310	2836	2131	1932	2131	2286
taiwan_binarised	5200	105	5201	3306	5200	83	5200	115
titanic	95	1428	95	1057	95	1464	95	1465
vehicle	1	690	1	3525*	3	42	1	1142
weather-aus	1735	419	1749	1835	1735	350	1735	401
wine1	33	1154	33	950	34	1319	33	1158
wine2	39	411	37	13	39	2756	39	409
wine3	25	17	25	90	25	100	25	16

Table 28: Factor analysis: $m \geq 100$, depth 5

	Blossom		No heuristic		No preprocessing		No lower bound	
	error	cpu	error	cpu	error	cpu	error	cpu
adult_discretized	4191	534	4203	686	4162	2418	4191	553
anneal	41	3036	49	2818	58	272	50	232
balance-scale	29	37*	29	39*	29	37*	29	40*
banknote	2	0.00*	2	0.01*	2	283*	2	0.00*
breast-cancer	0	1007*	0	824*	0	1024*	0	1194*
car	11	231*	11	256*	11	233*	11	627*
car_evaluation	80	0.00*	80	0.00*	80	27*	80	0.00*
chess	0	0.00*	0	0.00*	0	0.00*	0	0.00*
compas_discretized	1852	198*	1852	184*	1852	2030	1852	299*
heart-cleveland	0	0.00*	0	3.0*	0	0.03*	0	0.00*
hepatitis	0	0.00*	0	0.00*	0	0.00*	0	0.00*
HTRU_2	297	3334	324	2967	298	3052	297	2460
hypothyroid	22	3478	23	147	27	113	23	171
IndiansDiabetes	44	3343	45	3464	44	3448	47	579
iris	1	0.00*	1	0.00*	1	0.01*	1	0.00*
kr-vs-kp	18	2550	18	1423	34	3090	21	1756
lymph	0	0.00*	0	0.01*	0	0.00*	0	0.00*
magic04	2488	2773	2512	2481	2488	2657	2488	2586
messidor	179	2456	172	3162	179	3217	179	2901
monk1	0	0.00*	0	0.00*	0	0.00*	0	0.00*
monk2	0	0.00*	0	0.00*	0	0.00*	0	0.00*
monk3	0	0.00*	0	0.00*	0	0.00*	0	0.00*
primary-tumor	16	18*	16	17*	16	2866*	16	39*
seismic_bumps	76	2389	96	1453	78	1580	77	1217
soybean	2	19*	2	6.1*	2	729	2	32*
tic-tac-toe	0	32*	0	83*	0	31*	0	100*
vote	0	0.00*	0	0.04*	0	0.00*	0	0.00*
winequality-red	2	0.01*	2	0.13*	2	0.43	2	0.00*
yeast	182	3558	234	1611	210	1191	203	410

Table 29: Factor analysis: $m < 100$, depth 7

	Blossom		No heuristic		No preprocessing		No lower bound	
	error	cpu	error	cpu	error	cpu	error	cpu
audiology	0	0.00*	0	0.00*	0	0.00*	0	0.00*
australian-credit	0	101*	0	477*	0	1002*	0	153*
bank	3844	2369	4303	252	3844	2351	3844	2460
bank_conv	220	1642	288	1459	220	1459	220	1442
biodeg	26	2775	57	3064	26	2548	26	3341
breast-wisconsin	0	0.02*	0	0.23*	0	0.33*	0	0.03*
default_credit	4935	222	5054	557	4935	256	4935	187
diabetes	21	827	27	238	26	3164	21	1324
forest-fires	146	125	142	140	132	1346	146	124
german-credit	56	1192	117	2789	56	2472	56	1446
hand_posture	749	2684	8944	3595	749	2401	749	2702
ionosphere	0	0.07*	0	0.07*	0	0.49*	0	0.07*
letter	68	177	168	2143	70	3525	68	193
mnist_0	1107	2895	1556	1539	1107	2983	1107	2735
mushroom	0	0.00*	0	0.00*	0	0.00*	0	0.00*
pendigits	0	0.00*	0	3.5*	0	0.00*	0	0.00*
segment	0	0.00*	0	0.00*	0	0.00*	0	0.00*
spambase	352	3562	373	2535	357	2501	357	2249
splice-1	29	3484	46	3380	29	3575	29	3408
Statlog_satellite	14	2428	54	308	14	2062	14	2407
Statlog_shuttle	0	0.04*	0	25*	0	0.06*	0	0.04*
surgical-deepnet	1647	1248	2246	3102	1647	1086	1647	1288
taiwan_binarised	4896	1958	5016	2961	4909	1426	4896	2055
titanic	72	442	78	2696	72	471	72	500
vehicle	0	0.09*	0	196*	0	0.66*	0	0.10*
weather-aus	1685	2048	1747	1685	1685	1948	1685	2083
wine1	28	892	28	2666	29	487	28	892
wine2	31	28	31	23	31	168	31	28
wine3	21	524	21	1062	20	296	21	531

Table 30: Factor analysis: $m \geq 100$, depth 7

	Blossom		No heuristic		No preprocessing		No lower bound	
	error	cpu	error	cpu	error	cpu	error	cpu
adult_discretized	3841	2632	4119	3075	3775	2994	3841	2988
anneal	34	23*	36	1986	36	661	34	32*
balance-scale	0	19*	0	45*	0	21*	0	61*
banknote	2	0.00*	2	0.00*	2	0.00	2	0.00*
breast-cancer	0	0.00*	0	0.31*	0	0.00*	0	0.00*
car	0	0.26*	0	21*	0	0.32*	0	0.44*
car_evaluation	80	0.00*	80	0.00*	80	0.00	80	0.00*
chess	0	0.00*	0	0.00*	0	0.00*	0	0.00*
compas_discretized	1828	0.73*	1828	9.1*	1828	323	1828	1.4*
heart-cleveland	0	0.00*	0	0.00*	0	0.00*	0	0.00*
hepatitis	0	0.00*	0	0.00*	0	0.00*	0	0.00*
HTRU_2	219	550	272	340	218	638	219	559
hypothyroid	17	0.96*	17	40*	17	72	17	1.5*
IndiansDiabetes	8	4.7*	13	667	8	1732	8	8.6*
iris	1	0.00*	1	0.00*	1	0.02*	1	0.00*
kr-vs-kp	0	1897*	0	752*	5	86	1	400
lymph	0	0.00*	0	0.00*	0	0.00*	0	0.00*
magic04	1635	2746	2180	1847	1653	1768	1658	143
messidor	66	604	62	1498	66	1515	66	774
monk1	0	0.00*	0	0.00*	0	0.00*	0	0.00*
monk2	0	0.00*	0	0.00*	0	0.00*	0	0.00*
monk3	0	0.00*	0	0.00*	0	0.00*	0	0.00*
primary-tumor	15	0.00*	15	0.00*	15	0.28	15	0.00*
seismic_bumps	38	2591	98	874	45	319	45	1015
soybean	2	0.00*	2	0.43*	2	0.00	2	0.00*
tic-tac-toe	0	0.00*	0	0.00*	0	0.00*	0	0.00*
vote	0	0.00*	0	0.00*	0	0.00*	0	0.00*
winequality-red	2	0.00*	2	0.00*	2	0.00	2	0.00*
yeast	28	1008	68	2610	67	466	28	1633

Table 31: Factor analysis: $m < 100$, depth 10

	Blossom		No heuristic		No preprocessing		No lower bound	
	error	cpu	error	cpu	error	cpu	error	cpu
audiology	0	0.00*	0	0.00*	0	0.00*	0	0.00*
australian-credit	0	0.04*	0	0.15*	0	0.26*	0	0.04*
bank	3242	800	4200	20	3245	851	3242	845
bank_conv	169	2794	262	896	172	2851	174	2555
biodeg	1	1169*	40	1739	1	2928	1	1342*
breast-wisconsin	0	0.00*	0	0.00*	0	0.00*	0	0.00*
default_credit	4547	2019	4954	495	4561	1878	4549	1171
diabetes	0	0.67*	0	3026*	0	11*	0	0.60*
forest-fires	113	942	114	3068	118	3167	113	1003
german-credit	0	69*	62	2594	0	173*	0	96*
hand_posture	334	39	8927	2467	334	34	334	35
ionosphere	0	0.00*	0	0.02*	0	0.00*	0	0.00*
letter	0	79*	88	1825	0	1535*	0	104*
mnist_0	383	413	1721	3235	383	404	383	450
mushroom	0	0.00*	0	0.00*	0	0.00*	0	0.00*
pendigits	0	0.00*	0	0.00*	0	0.00*	0	0.00*
segment	0	0.00*	0	0.00*	0	0.00*	0	0.00*
spambase	262	546	321	2700	272	610	262	562
splice-1	5	1160	12	1676	4	3506	5	1205
Statlog_satellite	3	219	14	1016	3	195	3	215
Statlog_shuttle	0	0.02*	0	0.02*	0	0.02*	0	0.02*
surgical-deepnet	965	2865	1849	3204	965	3133	965	3192
taiwan_binarised	4217	1001	4896	2890	4189	1046	4217	1041
titanic	35	3059	52	943	45	1077	42	180
vehicle	0	0.00*	0	60*	0	0.00*	0	0.00*
weather-aus	1601	2591	1734	2391	1603	1988	1601	2758
wine1	22	545	20	1469	22	3227	22	539
wine2	24	399	21	20	24	2832	24	415
wine3	16	272	17	690	18	1802	16	270

Table 32: Factor analysis: $m \geq 100$, depth 10