Abstract

We consider supervised learning with random decision trees, where the tree construction is completely random. The method was used as a heuristic working well in practice despite the simplicity of the setting, but with almost no theoretical guarantees. The goal of this paper is to shed new light on the entire paradigm. We provide strong theoretical guarantees regarding learning with random decision trees. We present and compare three different variants of the algorithm that have minimal memory requirements: majority voting, threshold averaging and probabilistic averaging. The random structure of the tree enables us to adapt our setting to the differentially-private scenario thus we also propose differentially-private versions of all three schemes. We give upper bounds on the generalization error and mathematically explain how the accuracy depends on the number of random decision trees. Furthermore, we prove that only logarithmic number of independently selected random decision trees suffice to correctly classify most of the data, even when differential-privacy guarantees must be maintained. Such an analysis has never been done before. We empirically show that majority voting and threshold averaging give the best accuracy, also for conservative users requiring high privacy guarantees. In particular, this happens when majority voting and threshold averaging give the best accuracy, also for conservative users requiring high privacy guarantees. In particular, this happens when machine learning computations access data frequently during the entire execution of the algorithm and output structures that are very sensitive to the data. This is also an obstacle for proposing a scheme that computes an optimal decision tree in a differentially-private way. In such a scenario the attribute chosen in every node and any additional information stored there depends on the data and that is why it must be perturbed in order to keep the desired level of differential privacy. Big perturbation added in this setting leads to the...
Differentially- and non-differentially-private random decision trees

In this paper we consider a different approach where, instead of constructing one differentially-private decision tree, we construct a random forest. Random forests constitute an important member of the family of decision tree-based algorithms due to their effectiveness and excellent performance [Biau et al., 2008]. They are also the most accurate general-purpose classifiers available [Biau et al., 2008; Breiman, 2001]. In this paper we construct a forest consisting of $O(\log(n))$ random decision trees ($n$ is the size of the dataset). An attribute according to which the selection is performed in any given node is chosen uniformly at random from all the attributes, independently from the dataset in that node. In the continuous case, the threshold value for the chosen attribute is then also chosen uniformly at random from the range of all possible values. That simple rule enables us to construct each decision tree very fast since the choice of nodes’ attributes does not depend on the data at all. We therefore obtain fast and scalable algorithm with minimal memory requirements which, contrary to most learning algorithms dealing with random decision trees, takes only one pass over the data to construct the classifier. Since most of the structure of each random decision tree is constructed without examining the data, the algorithm suits the differentially-private scenario very well.

After a sufficient number of random decision trees is constructed, the classification of every point from a dataset takes place. Classification is done according to one of the three schemes: majority voting, threshold averaging or probabilistic averaging. Performing voting/averaging instead of just taking the best tree for a given dataset is important since it enables us to add smaller perturbation error to obtain the same level of differential privacy (see: [Schapire et al., 1998] for applications of the voting methods). For each scheme we analyze both: non-differentially-private and differentially-private setting. We are the first to give a comprehensive and unified theoretical analysis of all three models in both settings. We relate the empirical error and the generalization error of the classifier to the average tree accuracy and explain quantitatively how the quality of the system depends on the number of chosen trees. Furthermore, we show that, even in case of large datasets, the random forest need not many trees to achieve good accuracy. This finding runs counter to the conventional wisdom. In particular, we prove both theoretically and empirically that in practice the logarithmic number of random decision trees suffices to get good quality guarantees. Also we show that not only do they exist parameters of the setting (such as: the number of random trees in the forest, the height of the tree, etc.) under which one can effectively learn, but the setting is very robust.

To be more precise, we empirically show that the parameters do not need to be chosen in the optimal way, in example one can choose far fewer trees, to achieve good performance. We also show that majority voting and threshold averaging are good candidates for the differentially-private classifiers. Surprisingly, a simple majority voting rule, that was not considered before in this context, is competitive with the threshold averaging rule that has been empirically studied before. Furthermore, majority voting rule is much less sensitive to the choice of the parameters of the random forest (such as the number of the trees and the height of the tree).

To conclude, random decision trees were considered before as important heuristics often used for supervised learning due to their simplicity and somewhat unreasonable effectiveness in practice. Although their mechanism is simple, they are very difficult to analyze (which results in very little theoretical analysis available in the literature) and not well understood [Biau et al., 2008]. In this paper we provide a careful theoretical analysis of these methods. Our analysis is novel both in the non-differentially-private and differentially-private setting, where in case of the latter to the best of our knowledge no theoretical analysis was ever provided. The differentially-private setting is especially difficult to analyze since increasing the number of trees does not necessarily decrease the training (and test) error in this setting. Thus by giving strong theoretical guarantees regarding learning with random decision trees, also under very restrictive privacy assumptions, we make them a provable machine learning method.

In Section 2 we describe previous work regarding random decision trees. We then introduce our model and the notion of differential privacy in Section 3. In Section 4 we present a differentially-private supervised algorithm that uses random decision trees. Section 5 contains our theoretical analysis. We conclude the paper with experimental results (Section 6). The Supplementary material contains the proofs of our theorems as well as additional experiments.

2 Previous work versus our approach

As mentioned in the previous section, the statistical mechanism of random decision trees is still not well-understood [Biau et al., 2008]. Our paper is inspired by two previous works regarding random decision trees [Fan et al., 2003; Jagannathan et al., 2012]. The first work [Fan et al., 2003] introduces the random decision tree approach. The authors of this work focused exclusively on a very specific non-differentially-private scenario and proved that when increasing the number of random decision trees the accuracy of the
prediction is increasing too. They also showed experimentally that the approach based on random decision trees is surprisingly effective in practice. Their work however does not provide any upper bounds on the generalization error or the empirical error and in particular it does not explain how the error depends on the data. Contrary to their work, our analysis provides such error bounds. In the context of non-differentially-private learning studied in [Fan et al., 2003], we instead consider a broader range of possible variants of the algorithm. We furthermore extend the analysis to differentially-private setting and discover that a simple majority voting scheme, not studied before in this setting, significantly outperforms the probabilistic averaging rule that was already studied.

It was first observed in [Jagannathan et al., 2012] that random decision trees may turn out to be an effective tool for constructing a differentially-private decision tree classifier. The authors showed a very efficient heuristic that averages over random decision trees and gives good practical results. Their work however lacks theoretical results regarding the quality of the differentially-private algorithm that is using random decision trees. The authors also did not show how many random decision trees should be chosen to get reasonable accuracy. This is a fundamental problem in the differentially-private setting. As mentioned before, having more random decision trees require adding bigger perturbation error that may decrease the efficiency of the learning algorithm. In this paper we thoroughly investigate this phenomenon. The dependence of the quality of the random decision tree methods on the chosen level of differential privacy, the height of the tree and the number of trees in the forest is in the central focus of our analysis which in this respect is very novel. These dependencies were not explored before (some existing analysis of random decision tree approach can be found in [Fan, 2004] and [Fan et al., 2005]) but understanding them is crucial while applying those methods in practice.

For completeness we should notice that several machine learning random structures were considered before. For instance, in [Breiman, 2001] a random forest approach was presented where a computation involving the dataset is performed to choose an attribute for every node. This algorithm however is not as simple and fast as the one considered in this paper. Other examples of the results regarding random trees and their potential applications can also be found for instance in [Ho, 1995], [Choromanska et al., 2013]. Among those, a notable line of works provide an elegant analysis of their consistency. [Biau et al., 2008], [Meinshausen, 2006], [Lin and Jeon, 2002], [Breiman, 2000], [Breiman, 2004].

3 Preliminaries

3.1 Differential privacy

Differential privacy is a model of privacy for database access mechanism. It guarantees that small changes in a database (removal or addition of an element) does not change substantially the output of the mechanism.

Definition 3.1 ([Dwork et al., 2006]). A randomized algorithm \( M \) satisfies \( \epsilon \)-differential-privacy if for all databases \( D_1 \) and \( D_2 \) differing on at most one element, and all measurable sets \( S \),

\[
\Pr(M(D_1) = S) \leq \exp(\epsilon) \cdot \Pr(M(D_2) = S).
\]  

(1)

The probability is taken over the coin tosses of \( M \).

The smaller \( \epsilon \), the stronger level of differential privacy is obtained. Assume that the non-perturbed output of the mechanism can be encoded by the function \( f \). A mechanism \( M \) can compute a differentially-private noisy version of \( f \) over a database \( D \) by adding noise with magnitude calibrated to the sensitivity of \( f \).

Definition 3.2 ([Dwork et al., 2006]). The global sensitivity \( S(f) \) of a function \( f \) is the smallest number \( s \) such that for all \( D_1 \) and \( D_2 \) which differ on at most one element, \( |f(D_1) - f(D_2)| \leq s \).

Let \( \text{Lap}(0,\lambda) \) denote the Laplace distribution with mean 0 and standard deviation \( \lambda \).

Theorem 3.1 ([Dwork et al., 2006]). Let \( f \) be a function on databases with range \( \mathbb{R}^m \), where \( m \) is the number of rows of databases. Then, the mechanism that outputs \( f(D) + (Y_1, \ldots, Y_m) \), where \( Y_i \) are drawn i.i.d from \( \text{Lap}(0, S(f)/\epsilon) \), satisfies \( \epsilon \)-differential-privacy.

Stronger privacy guarantees and more sensitive functions need bigger variance of the Laplacian noise being added. Differential privacy is preserved under composition, but with an extra loss of privacy for each conducted query.

Theorem 3.2 ([Dwork et al., 2006]). (Composition Theorem) The sequential application of mechanisms \( M_i \), each giving \( \epsilon_i \)-differential privacy, satisfies \( \sum_i \epsilon_i \)-differential-privacy.

More information about differential privacy can be found in the work of [McSherry and Talwar, 2007] and [Hall et al., 2013]. Machine learning algorithms in the context of differential privacy have been considered before, e.g. see: [Mir, 2012], [Jain et al., 2012].

3.2 The model

All data points are taken from \( \mathcal{F}^m \), where \( m \) is the number of the attributes and \( \mathcal{F} \) is either a discrete set or the set of real numbers. We assume that for every attribute \( \text{attr} \) its smallest (\( \min(\text{attr}) \)) and largest
possible value ($\max(\text{attr})$) are publicly available and that the labels are binary. We consider only binary decision trees (all our results can be easily translated to the setting where inner nodes of the tree have more than two children). Therefore, if $\mathcal{F}$ is discrete then we will assume that $\mathcal{F} = \{0, 1\}$, i.e. each attribute is binary. In the continuous setting for each inner node of the tree we store the attribute according to which the selection is done and the threshold value of this attribute. All decision trees considered in this paper are complete and of a fixed height $h$ that does not depend on the data. Let $T$ be a random decision tree and let $l$ be one of its leaves. We denote by $\theta_l$ the fraction of all training points in $l$ with label $+$. If $l$ does not contain any of the training points we choose the value of $\theta_l$ uniformly at random from $[0, 1]$. The set $M$ of all possible decision trees is of size $|M| = n^{2^{h+1} - 1}$ in the binary setting. It should be emphasized that it is true also in the continuous case. In that setting the set of all possible threshold values for a node is infinite but needless to say, the set of all possible partitionings in the node is still finite. Thus without loss of generality, we assume $M$ is finite. It can be very large but it does not matter since we will never need the actual size of $M$ in our analysis. For a given tree $T$ and given data point $d$ denote by $w_T^d$ the fraction of points (from the training set if $d$ is from this set and from the test set otherwise) with the same label as $d$ that end up in the same leaf of $T$ as $d$. We call it the weight of $d$ in $T$. Notice that a training point is classified correctly by $T$ in the single-tree setting iff its weight in $T$ is larger than $\frac{1}{2}$ (for a single decision tree we consider majority voting model for points classification). The average value of $w_T^d$ over all trees of $M$ will be denoted as $w_d$ and called the weight of $d$ in $M$. We denote by $\sigma(d)$ the fraction of trees from $M$ with the property that most of the points of the leaf of the tree containing $d$ have the same label as $d$ (again, the points are taken from the training set if $d$ is from it and from the test set otherwise). We call $\sigma(d)$ the goodness of $d$ in $M$. For a given dataset $\mathcal{D}$ the average tree accuracy $e(\mathcal{D})$ of a random decision tree model is an average accuracy of the random decision tree from $M$, where the accuracy is the fraction of data points that a given tree classifies correctly (accuracy is computed under assumption that the same distribution $\mathcal{D}$ was used in both: the training phase and test phase).

4 Algorithms

The non-differentially-private algorithms for supervised learning with random decision are captured in Algorithm $\textbf{1}$. Their differentially-private counterparts are captured in Algorithm $\textbf{2}$. We consider three versions of each algorithm: majority voting, threshold averaging and probabilistic averaging. Only variables $n_T^\pm$, $n_I^\pm$ stored in leaves depend on the data. This fact will play crucial role in the analysis of the differentially-private version of the algorithm where Laplacian error is added to the point counters at every leaf with variance calibrated to the number of all trees used by the algorithm (we denote by $g(\lambda)$ an independent copy of the Laplacian with pdf: $\frac{\lambda}{2}e^{-|x|/\lambda}$).

**Algorithm 1** Non-differentially-private RTD classifier

**Input:** $\text{Train, Test}$: train and test sets, $h$: height of the tree

**Random forest construction:**
construct $k = \theta(\log(n))$ random decision trees by choosing for each inner node of the tree independently at random its attribute (uniformly from the set of all the attributes); in the continuous case for each chosen attribute attr choose independently at random a threshold value uniformly from $[\min(attr), \max(attr)]$

**Training:**
For $d \in \text{Train}$ 
add $d$ to the forest by updating $\theta_l$ for every leaf corresponding to $d$

**Testing:**
For $d \in \text{Test}$ 
if (majority voting) 
compute $\mathsf{num}^d$ - the number of the trees classifying $d$ as $+$; classify $d$ as $+$ if $\mathsf{num}^d > \frac{n}{2}$ 

if (threshold averaging) 
compute $\theta^d = \frac{1}{|L|}\sum_{l \in L} \theta_l$, where $L$ is a set of all leaves of the forest that correspond to $d$; classify $d$ as $+$ if $\theta^d > \frac{1}{2}$ 

if (probabilistic averaging) 
compute $\theta^d = \frac{1}{2} \sum_{l \in L} \theta_l$, where $L$ is a set of all leaves of the forest that correspond to $d$; classify $d$ as $+$ with probability $\theta^d$
/*random tosses here are done independently from all other previously conducted*/

**Output:** Classification of all $d \in \text{Test}$

5 Theoretical results

In this section we derive the upper bounds on the empirical error (the fraction of the training data misclassified by the algorithm) and the generalization error (the fraction of the test data misclassified by the algorithm where the test data is taken from the same distribution as the training data) for all methods in Algorithm $\textbf{1}$ and $\textbf{2}$. We also show how to find the number of random decision trees to obtain good accuracy and, in the differentially-private setting, good privacy guarantees. All proofs of the theoretical statements of this section are deferred to the Supplementary material.
Then the average weight $w$ of the tree is better than random, then this is also re-

If we know that the average tree accuracy is itself big for some $0 < \epsilon \leq \frac{1}{2}$, then this implies very small em-

We start with two technical results which, as we will see later, give an intuition why the random decision tree approach works very well in practice.

Theorem 5.1. Assume that the average tree accuracy of the set $M$ of all decision trees of height $h$ on the training/test set $D$ is $e = 1 - \epsilon$ for some $0 < \epsilon \leq \frac{1}{2}$. Then the average goodness $\sigma(d)$ of a training/test point $d$ in $M$ is at least $e \geq \frac{1}{2}$.

Theorem 5.2. Assume that the average tree accuracy of the set $M$ of all decision trees of height $h$ on the training/test set $D$ is $e = 1 - \epsilon$ for some $0 < \epsilon \leq \frac{1}{2}$. Then the average weight $w_d$ of a training/test point $d$ in $M$ is at least $e^2 + (1 - e)^2 \geq \frac{1}{2}$.

The theorems above imply that if the average accuracy of the tree is better than random, then this is also reflected by the average values of $w_d$ and $\sigma_d$. This fact is crucial for the theoretical analysis since we will show that if the average values of $w_d$ and $\sigma_d$ are slightly better than random then this implies very small empirical and generalization error. Furthermore, for most of the training/test points $d$ their values of $\sigma_d$ and $w_d$ are well concentrated around those average values and that, in a nutshell, explains why the random decision trees approach works well. Notice that Theorem 5.1 gives better quality guarantees than Theorem 5.2.

5.1 Non-differentially-private setting

We start by providing theoretical guarantees in the non-differentially-private case. Below we consider majority voting and threshold averaging. The results for the probabilistic averaging are stated later in this para-

Theorem 5.3. Let $K > 0$. Assume that the average tree accuracy of the set $M$ of all decision trees of height $h$ on the training/test set $D$ is $e = 1 - \epsilon$ for some $0 < \epsilon \leq \frac{1}{2}$. Then the average goodness $\sigma$ of training/test points with weight in $M$ at least $w = \frac{1}{2} + \delta / w = \frac{1}{2} + \delta + \frac{1}{K}$ for $0 < \delta < \frac{1}{2}$ (in the threshold averaging version). Then Algorithm 1 for every $C > 0$ and $k = \frac{(1+C)\log(n)}{2\delta^2}$ selected random decision trees gives empirical error $\text{err}_1 \leq 1 - \mu$ with probability $p_1 \geq 1 - \frac{1}{n^2}$. The generalization error $\text{err}_2 \leq 1 - \mu$ will be achieved for $k = \frac{(1+C)\log(n)}{2\delta^2}$ trees with probability $p_2 \geq 1 - 2h + 3ke^{-2\phi^2}$, where $\phi = \frac{\delta}{2(1+C)\delta^2K}$. Probabilities $p_1$ and $p_2$ are under random coin losses used to construct the forest.

Note that parameter $\epsilon$ is always in the range $[\frac{1}{2}, 1]$. The more decision trees that classify data in the non-trivial way (i.e. with accuracy greater than $\frac{1}{2}$), the larger the value of $\epsilon$ is. The result above in particular implies that if most of the points have goodness/weight in $M$ a little bit larger than $\frac{1}{2}$ then both errors are very close to 0. This is indeed the case - the average point’s goodness/weight in $M$, as Theorem 5.1 and Theorem 5.2 say, is at least $\epsilon / e^2 + (1 - e)^2$. The latter expression is greater than $\frac{1}{2}$ if the average tree accuracy is slightly bigger than the worst possible. Besides goodness/weight of most of the points, as was tested experimentally, is well concentrated around that average goodness/weight. We conclude that if the average accuracy of the decision tree is separated from $\frac{1}{2}$ (but not necessarily very close to 1) then it suffices to classify most of the data points correctly. The intuition behind this result is as follows: if the constructed forest of the decision trees contains at least few “nontrivial trees” giving better accuracy than random then they guarantee correct classification of most of the points.

If we know that the average tree accuracy is itself big then techniques used to prove Theorem 5.3 give us more direct bounds on the empirical and generalization errors captured in Theorem 5.4.

Theorem 5.4. Let $K > 0$. Assume that the average tree accuracy of the set $M$ of all decision trees of height $h$ on the training/test set $D$ is $e = 1 - \epsilon$ for some $0 < \epsilon \leq \frac{1}{2}$. Then Algorithm 1 for every $C > 0$, $0 < \delta < \frac{1}{2}$ and $k = \frac{(1+C)\log(n)}{2\delta^2}$ selected random decision trees gives empirical error: $\text{err}_1 \leq \frac{\sqrt{e^2 - \frac{1}{2}}}{\sqrt{e} - \frac{1}{2}}$ (in the majority version) or: $\text{err}_1 \leq \frac{2\epsilon(1 - \epsilon)^2}{0.5 - \epsilon}$ (in the threshold averaging version) with probability $p_1 \geq 1 - \frac{1}{n^2}$. The generalization error $\text{err}_2 \leq \frac{\epsilon^2 + \frac{1}{2}}{\epsilon - \frac{1}{2}}$ (in the majority ver-
Then with probability at least $\phi$ (in the threshold averaging version) will be achieved for $k = \frac{(1+C)\log(n)}{2(2-\epsilon)^2}$ trees with probability $p_2 \geq p_1 - 2^{h+3k}\epsilon^{-2n^2}$, where $\phi = \frac{\delta}{2(4+\epsilon^2)2^{2K}}$. Probabilities $p_1$ and $p_2$ are under random coin tosses used to construct the forest.

Theorems 5.3 and 5.4 show that logarithmic number of random decision trees in practice suffices to obtain high prediction accuracy with a very large probability. In particular, the upper bound on the generalization error is about two times the average error of the tree. The existence of the tree with lower accuracy in the forest does not harm the entire scheme since all trees of the forest play role in the final classification.

We now state our results for the probabilistic averaging setting. In this setting the following theorem holds.

**Theorem 5.6.** Assume that the average tree accuracy of the set $M$ of all decision trees of height $h$ on the training/test set $D$ is $e = 1 - \epsilon$ for some $0 < \epsilon < 1$. Let $C > 0$ be a constant. Let $0 < \delta, c < 1$. Then with probability at least $p_1 = \frac{1}{1 - n\epsilon}(1 - e^{-2n^2\epsilon^2})$ the probabilistic averaging version of Algorithm\[ gives empirical error $err_1 \leq 2\epsilon - 2\epsilon^2 + 3\delta + c$ and with probability $p_2 \geq p_1 - 2^{h+3k}\epsilon^{-2n^2}$, where $\phi = \frac{\delta}{2(4+\epsilon^2)2^{2K}}$, it gives generalization error $err_2 \leq 2(\epsilon + \frac{1}{n}) - 2(\epsilon + \frac{1}{n})^2 + \delta + c$. Probabilities $p_1$ and $p_2$ are under random tosses used to construct the forest.

Notice that this result is nontrivial for almost the entire range $[0, \frac{1}{2}]$ of $\epsilon$, and $\delta$ and $c$ close to 0, and large $K$. This is the case since note that $1 - 2\epsilon + 2\epsilon^2 \geq \frac{1}{2}$ and the equality holds only for $\epsilon = \frac{1}{2}$.

5.2 Differentially-private setting

We begin this section with the proof that all three methods captured in Algorithm\[ where the Laplacian noise is added to certain counts, are indeed $\eta$-differentially private.

**Proof.** Notice that in every method to obtain the forest of random decision trees with perturbed counters in leaves we need $k$ queries to the private data (this is true since the structure of the inner nodes of the trees does not depend at all on the data and data subsets corresponding to leaves are pairwise disjoint). Furthermore, the values that are being perturbed by the Laplacian noise are simple counts of global sensitivity 1. Thus we can use use Theorem 3.1 and Theorem 3.2 to conclude that in order to obtain $\eta$-differential privacy of the entire system we need to add a Lap$(0, \frac{k}{\eta})$ to every count in the leaf. This proves that our algorithms are indeed $\eta$-differentially private.

Next we show the theoretical guarantees we obtained in the differentially-private setting. As in the previous section, we first focus on the majority voting and threshold averaging, and then we consider the probabilistic averaging.

**Theorem 5.7.** Assume that we are given a parameter $\eta > 0$. Let $K > 0$. Assume that the average tree accuracy of the set $M$ of all decision trees of height $h$ on the training/test set $D$ is $e = 1 - \epsilon$ for some $0 < \epsilon < 1$. Let $c_0$ be the fraction of training/test points with: goodness in $M$ at least $\sigma = \frac{1}{2} + \delta + \frac{c}{K}$ (in the majority version) or: weight in $M$ at least $w = \frac{1}{2} + \delta + \frac{c}{K}$ (in the threshold averaging version) for $0 < \delta < \frac{1}{2}$. Then Algorithm\[ for $k$ selected random decision trees and differential privacy parameter $\eta$ gives empirical error $err_1 \leq 1 - \mu$ with probability $p_1 \geq 1 - n(e^{-\frac{4\epsilon^2}{K}} + e^{-\frac{2}{K}} + ke^{-\frac{2\eta^2}{K}})$ and generalization error $err_2 \leq 1 - \mu$ with probability $p_2 \geq p_1 - 2^{h+3k}\epsilon^{-2n^2}$, where: $\lambda = \frac{\delta}{24K^2}$ and $\phi = \frac{\delta}{2(4+\epsilon^2)2^{2K}}$. Probabilities $p_1$ and $p_2$ are under random coin tosses used to construct the forest. Furthermore, we always have: $\mu \geq 1 - \frac{e^{-\frac{4\epsilon^2}{K}}}{\frac{1}{2} - \delta - \frac{c}{K}}$ in the majority version and: $\mu \geq 1 - \frac{e^{-\frac{2\eta^2}{K}}}{\frac{1}{2} - \delta - \frac{c}{K}}$ in the threshold averaging version.

Again, as in the non-differentially-private case, we see that if there are many points of goodness/weight in $M$ close to the average goodness/weight then empirical and generalization error are small. Notice also that increasing the number of the trees too much has an impact on the empirical error (term $ke^{-\frac{2\eta^2}{K}}$ in the lower bound on $p_1$). More trees means bigger variance of the single Laplacian used in the leaf of the tree. This affects tree quality. The theorem above describes this phenomenon quantitatively.

If the average tree accuracy is big enough then the following result becomes of its own interest. This result considers in particular the empirical error (similar result holds for the generalization error) of the threshold averaging version of Algorithm\[ (and also similar result holds for majority voting version of Algorithm\[).

**Theorem 5.8.** Assume that we are given a parameter $\eta > 0$. Assume besides that the average tree accuracy of the set $M$ of all decision trees of height $h$ on the training set $D$ is $e = 1 - \epsilon$ for some $0 < \epsilon < 1$. Let $0 < \delta < 1$. Let $c_0 = \frac{1}{2000}$ and let $k_{opt}$ be the integer value for which the value of the function $f(k) = e^{-\frac{k}{1000}} + 2ke^{-\frac{2\eta^2}{K}}$ is smallest possible. Then with probability at least $p = 1 - n(e^{-\frac{4\epsilon^2}{K}} + 2ke^{-\frac{2\eta^2}{K}})$ the $\eta$-differentially-private threshold averaging version of Algorithm\[ gives empirical error at most $\frac{1}{K} + \frac{2}{n} + 2\epsilon - 5\epsilon^2$ for the forest with $k_{opt}$ randomly chosen decision trees. Probability $p$ is under
Algorithm 2 gives empirical error at most $\epsilon$-differentially-private probabilistic averaging version of Theorem 5.9. As in the two previous settings, information about the under random coin tosses used to construct the forest. Both theorems show that logarithmic number of random decision trees in practice suffices to obtain good accuracy and high level of differential privacy.

The next theorem considers the differentially-private probabilistic averaging setting.

**Theorem 5.8.** Assume that we are given a parameter $\eta > 0$. Let $K, c > 0$ and $0 < \delta < 1$. Assume that the average tree accuracy of the set $M$ of all decision trees of height $h$ on the training/test set $D$ is $e = 1 - e$ for some $0 < e \leq \frac{1}{2}$. Let $\lambda = \frac{h}{2^{h+2}\ln 2}$. Then for $k$ selected random decision trees the $\eta$-differentially-private probabilistic averaging version of Algorithm 2 gives empirical error $\text{Err}_1 \leq 2(1 + \frac{e}{2}) - 2(e + \frac{1}{2})^2 + 2h + 3ke^{-2n\phi^2}$ with probability $p_1 \geq 1 - n(e^{\frac{-3h^2}{2k}} + e^{\frac{-h}{2}} + ke^{-\frac{2h}{2n\phi}})(1 - e^{\frac{-h}{2n\phi}})$ and generalization error $\text{Err}_2 \leq 2(1 + \frac{e}{2}) - 2(e + \frac{1}{2})^2 + 2h + 3ke^{-2n\phi^2}$ with probability $p_2 \geq p_1 = \frac{1}{2} - \frac{e}{2\theta}$. Probabilities $p_1$ and $p_2$ are under random coin tosses used to construct the forest.

As in the two previous settings, information about the average accuracy of just a single tree gives strong guarantees regarding the classification quality achieved by the differentially-private version of the forest. The next result shows how to choose the optimal number of trees and that this number is in fact small.

**Theorem 5.9.** Assume that we are given a parameter $\eta > 0$. Assume besides that the average tree accuracy of the set $M$ of all decision trees of height $h$ on the training set $D$ is $e = 1 - e$ for some $0 < e \leq \frac{1}{2}$. Let $\gamma = \frac{1}{2^{h+9\ln 2}}$ and let $k_{\text{opt}}$ be the integer value for which the value of the function $f(k) = e^{-\frac{k}{2}} + 2e^{-\frac{3k}{2n\phi^2}}$ is smallest possible. Then with probability at least $p = 1 - n(e^{\frac{-3h^2}{2k_{\text{opt}}}} + 2k_{\text{opt}}e^{-\frac{3k_{\text{opt}}}{2n\phi^2}} + e^{\frac{-h}{2}})(1 - e^{\frac{-h}{2n\phi}})$ the $\eta$-differentially-private probabilistic averaging version of Algorithm 2 gives empirical error at most $\frac{1}{2} + \frac{19}{10}e - 2\epsilon^2$ for the forest with $k_{\text{opt}}$ randomly chosen decision trees. Probability $p$ is under random coin tosses used to construct the forest.

### 6 Experiments

The experiments were performed on the benchmark datasets\footnote{downloaded from http://osmot.cs.cornell.edu/kddcup/, http://archive.ics.uci.edu/ml/datasets.html, and http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets} banknote authentication, Blood Transfusion Service Center, Congressional Voting Records, Mammographic Mass, Mushroom, adult, covertype and quantum. 90% of each dataset was used for training and the remaining part for testing. Furthermore, 10% of the training dataset was used as a validation set. All codes for our experiments are publicly released.

We first compare the test error (%) obtained using five different methods: open-source implementation of CART called rpart \cite{Therneau et al.} non-differentially-private (n-dp) and differentially-private (dp) random forest with majority voting (RFMV) and with threshold averaging (RFTA). For all methods except rpart we also report the number of trees in the forest ($k$) and the height of the tree ($h$) for which the smallest validation error was obtained, where we explored: $h \in \{1, 2, 3, \ldots, 15\}$ and $k \in \{1, 3, 5, \ldots, 21\}$. In all the experiments the differential privacy parameter $\eta$ was set to $\eta = 1000/n_{tr}$, where $n_{tr}$ is the number of training examples. Table 1 captures the results (for each experiment we report average test error over 10 runs). The performance of random forest with probabilistic averaging (RFPA) was significantly worse than the competitive methods (RFMV, RFTA, rpart) and is not reported in the table. The performance of RFPA will however be shown in the next set of results.

Next set of results\footnote{All figures in this section and in the Supplementary material should be read in color.} (Figure 1 and 2) are reported for an exemplary dataset (Mushrooms) and for the following methods: dpRFMV, dpRFTA and dpRFPA. Note that similar results were obtained for other datasets and they are deferred to the Supplementary material. In Figure 1 we report the test error vs. $h$ for selected settings of $k$. In Figure 2 we also show minimal,
Differentially- and non-differentially-private random decision trees

Figure 1: Comparison of dpRFMV, dpRFTA and dpRFPA. η = 1000/n_t, = 0.137. Test error resp. vs. a) h across various settings of k and vs. c) k across various settings of h; Minimal, average and maximal test error resp. vs. h (b)) and vs. k (d)) for dpRFMV.

Figure 2: Comparison of dpRFMV, dpRFTA and dpRFPA. a) Test error vs. η for two settings of h. Test error vs. k for fixed h and across different settings of η. b) b) Test error vs. k for fixed h and different levels of η.

average and maximal test error vs. h for dpRFMV, whose performance was overall the best. Similarly, in Figure 1c we report the test error vs. h (b)) and vs. k (d)) for dpRFMV.

Finally, in Figure 2a we report test error for various settings of η and two selected settings of h. For each experiment k was chosen from the set {1, 2, ..., 101} to give the smallest validation error. Additionally, in Figure 2b, we show how the test error changes with k for a fixed h and different levels of η.

Figure 2a shows that in most cases dpRFTA outperforms remaining differentially-private classifiers, however it requires careful selection of the forest parameters (h and k) in order to obtain the optimal performance as is illustrated on Figure 1c and 2b. This problem can be overcome by using dpRFMV which has comparable performance to dpRFTA but is much less sensitive to the setting of the forest parameters. Therefore dpRFMV is much easier to use in the differentially-private setting.

7 Conclusion

In this paper we first provide novel theoretical analysis of supervised learning with non-differentially-private random decision trees. Secondly we show that, contrary to most of the related techniques, the algorithms we consider here can be easily adapted to the setting where high privacy guarantees must be achieved. We furthermore provide both theoretical and experimental evaluation of the differentially-private random decision trees approach. To the best of our knowledge, the theoretical analysis of the differentially-private setting was never done before. Our experiments reveal that majority voting and threshold averaging are good differentially private classifiers and that in particular majority voting, which was not studied before in the context of differentially-private learning, exhibits less sensitivity to forest parameters.
References

[Agrawal and Srikant, 2000] Agrawal, R. and Srikant, R. (2000). Privacy-preserving data mining. *ACM SIGMOD*, pages 439–450.

[Biau et al., 2008] Biau, G., Devroye, L., and Lugosi, G. (2008). Consistency of random forests and other averaging classifiers. *J. Mach. Learn. Res.*, 9:2015–2033.

[Breiman, 2000] Breiman, L. (2000). Some infinite theory for predictor ensembles. *Technical Report 577, Statistics Department, UC Berkeley*. [http://www.stat.berkeley.edu/~breiman](http://www.stat.berkeley.edu/~breiman).

[Breiman, 2001] Breiman, L. (2001). Random forests. *Machine Learning*, 45:5-32.

[Breiman, 2004] Breiman, L. (2004). Consistency for a simple model of random forests. *Technical Report 670, Statistics Department, UC Berkeley*.

[Chaudhuri and Monteleoni, 2008] Chaudhuri, K. and Monteleoni, C. (2008). Privacy-preserving logistic regression. In *NIPS*.

[Choromanska et al., 2013] Choromanska, A., Choromanski, K., Jagannathan, G., and Monteleoni, C. (2013). Differentially-private learning of low dimensional manifolds. In *ALT*.

[Choromanski et al., 2013] Choromanski, K., Jebara, T., and Tang, K. (2013). Adaptive anonymity via $b$-matching. In *NIPS*.

[Du and Zhan, 2003] Du, W. and Zhan, J. (2003). Using randomized response techniques for privacy-preserving data mining. In *KDD*.

[Dwork et al., 2006] Dwork, C., McSherry, F., Nissim, K., and Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In *TCC*.

[Fan, 2004] Fan, W. (2004). On the optimality of probability estimation by random decision trees. In *AAAI*.

[Fan et al., 2005] Fan, W., Greengrass, E., McCloskey, J., Yu, P., and Drummey, K. (2005). Effective estimation of posterior probabilities: Explaining the accuracy of randomized decision tree approaches. In *ICDM*.

[Fan et al., 2003] Fan, W., Wang, H., Yu, P., and Ma, S. (2003). Is random model better? on its accuracy and efficiency. *ICDM*.

[Hall et al., 2013] Hall, R., Rinaldo, A., and Wasserman, L. A. (2013). Differential privacy for functions and functional data. *Journal of Machine Learning Research*, 14(1):703–727.

[Ho, 1995] Ho, T. (1995). Random decision forest. In *ICDAR*.

[Jagannathan et al., 2012] Jagannathan, G., Pillai, K., and Wright, R. N. (2012). A practical differentially private random decision tree classifier. *Trans. Data Privacy*, 5(1):273–295.

[Jain et al., 2012] Jain, P., Kothari, P., and Thakurta, A. (2012). Differentially private online learning. In *COLT*.

[Lin and Jeon, 2002] Lin, Y. and Jeon, Y. (2002). Random forests and adaptive nearest neighbors. *Journal of the American Statistical Association*, 101:578-590.

[McSherry and Talwar, 2007] McSherry, F. and Talwar, K. (2007). Mechanism design via differential privacy. In *FOCS*.

[Meinshausen, 2006] Meinshausen, N. (2006). Quantile regression forests. *Journal of Machine Learning Research*, 7:983–999.

[Mir, 2012] Mir, D. (2012). Differentially-private learning and information theory. In *EDBT/ICDT Workshops*.

[Schapire et al., 1998] Schapire, R. E., Freund, Y., Bartlett, P., and Lee, W. S. (1998). Boosting the margin: A new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 26:1651–1686.

[Therneau et al., 2011] Therneau, T. M., Atkinson, B., and Ripley, B. (2011). rpart: Recursive partitioning. [http://CRAN.R-project.org/package=rpart](http://CRAN.R-project.org/package=rpart).
8 Empirical and generalization errors

8.1 Preliminaries

We will prove here results regarding empirical and generalization errors of all the variants of the algorithm mentioned in the paper as well as Theorem 5.1 and Theorem 5.2. Without loss of generality we will assume that all attributes are binary (taken from the set \{0, 1\}). It can be easily noticed that the proofs can be directly translated to the continuous case. We leave this simple exercise to the reader.

Let us introduce first some useful notation that will be very helpful in the proofs we present next.

We denote by \( n \) the size of the dataset (training or test) \( \mathcal{T} \). Let us remind that \( m \) is the number of attributes of any given data point, \( h \) is the height of the random decision tree and \( M \) is the set of all random decision trees under consideration.

We focus first on classifying with just one decision tree. Fix some decision tree \( T_j \) and one of its leaves. Assume that it contains \( a \) points with label: \(-\) and \( b \) points with label: \(+\). We associate label \(-\) with that leaf if \( a > b \) and label 1 otherwise. To classify a given point using that tree we feed our tree with that point and assign to a point a label of the corresponding leaf. Denote by \( m^i \) the number of data points that were correctly classified by a tree \( T_i \). Denote \( e^i = \frac{m^i}{n} \). We call \( e^i \) the quality (or accuracy) of the tree \( T_i \). Note that obviously we always have: \( e^i \geq \frac{1}{2} \), since for every leaf of any given tree the majority of the data points from that leaf are classified correctly. Denote: \( e = \frac{1}{|M|} \sum_{i=1}^{|M|} e^i \). We call \( e \) the average tree accuracy. This parameter measures how well data points are classified on average by a complete decision tree of a given height \( h \). Note that \( e \geq \frac{1}{2} \). Denote \( t = 2^h \). Parameter \( t \) is the number of leaves of the decision tree.

For \( i = 1, 2, \ldots, |M| \) and \( j = 1, 2, \ldots, t \) denote by \( n^i_j \) the number of points from the dataset in the \( j^{th} \) leaf of a decision tree \( T_i \). Denote by \( m^i_j \) the number of points from the dataset in the \( j^{th} \) leaf of the decision tree \( T_i \) that were classified correctly. Denote \( e^i_j = \frac{m^i_j}{n^i_j} \) for \( n^i_j > 0 \) and \( e^i_j = 1 \) for \( n^i_j = 0 \). Note that \( e^i_j \geq \frac{1}{2} \) for every \( i, j \). Note also that we have: \( n = n^1_1 + \ldots + n^i_1 \) and \( m^i = m^1_1 + \ldots + m^i_1 \). Denote by \( a^i_j \) the number of data points in the \( j^{th} \) leaf of the decision tree \( T_i \) that are of label 0. Denote by \( b^i_j \) the number of data points in the \( j^{th} \) leaf of the decision tree \( T_i \) that are of label 1.

We will use frequently the following structure in the proofs. Let \( G \) be a bipartite graph with color classes: \( \mathcal{A} \), \( \mathcal{B} \) and weighted edges. Color class \( \mathcal{A} \) consists of \( n \) points from the dataset. Color class \( \mathcal{B} \) consists of \( 2t|M| \) elements of the form \( y^i_j, \) where \( i \in \{1, 2, \ldots, |M|\} \), \( b \in \{0, 1\} \) and \( j \in \{1, 2, \ldots, t\} \).

Data point \( x \in \mathcal{A} \) is adjacent to \( y^{i,1}_j \) iff it belongs to larger of the two groups (these with labels: 0 and 1) of the data points that are in the \( j^{th} \) leaf of the decision tree \( T_i \). An edge joining \( x \) with \( y^{i,1}_j \) has weight \( e^i_j \). Data point \( x \in \mathcal{A} \) is adjacent to \( y^{i,0}_j \) iff it belongs to smaller of the two groups of the data points that are in the \( j^{th} \) leaf of the decision tree \( T_i \). An edge joining \( x \) with \( y^{i,0}_j \) has weight \( 1 - e^i_j \). Note that the degree of a vertex \( y^{i,1}_j \) is \( m^i_j \) and the degree of a vertex \( y^{i,0}_j \) is \( n^i_j - m^i_j \).

In the proofs we will refer to the size of the set of decision trees under consideration as: \(|M| \) or \( k \) (note that \( k \) is used in the main body of the paper).

We are ready to prove Theorem 5.1 and Theorem 5.2.
Proof. We start with the proof of Theorem 5.2 Note that from the definition of \( w_d \) we get:

\[
\sum_{d \in T} w_d = \frac{1}{|M|} \sum_{i=1}^{|M|} \sum_{j=1}^{t} (m_i^j \epsilon_j^i + (n_i^j - m_i^j)(1 - \epsilon_j^i)).
\]

Therefore, using formula on \( m_i^j \), we get:

\[
\sum_{d \in T} w_d = \frac{1}{|M|} \sum_{i=1}^{|M|} \sum_{j=1}^{t} (n_i^j(\epsilon_j^i)^2 + n_i^j(1-\epsilon_j^i)^2).
\]

Note that we have: \( \sum_{i=1}^{|M|} \sum_{j=1}^{t} n_i^j = n|M| \). From Jensen’s inequality, applied to the function \( f(x) = x^2 \), we get:

\[
\sum_{i=1}^{|M|} \sum_{j=1}^{t} \frac{n_i^j}{|M|n}(\epsilon_j^i)^2 \geq (\sum_{i=1}^{|M|} \sum_{j=1}^{t} \frac{n_i^j}{|M|n})^2 = (\frac{1}{|M|n})^2 = e^2,
\]

where \( e \) is the average quality of the system of all complete decision trees of height \( h \) (the average tree accuracy). Similarly, \( \sum_{i=1}^{|M|} \sum_{j=1}^{t} n_i^j (1-e_j^i)^2 \geq (1-e)^2 \). Thus we get:

\[
\sum_{d \in T} w_d \geq n(e^2 + (1 - e)^2).
\]

That completes the proof of Theorem 5.2. The proof of Theorem 5.1 is even simpler. Notice that for any data point \( d \) the expression \( \sigma(d) \cdot |M| \) counts the number of decision trees from \( M \) that classified \( d \) correctly (follows directly from the definition of \( \theta \)). Thus we have:

\[
\sum_{d \in T} \sigma(d) \cdot |M| = \sum_{i=1}^{|M|} m^i.
\]

Therefore \( \frac{1}{n} \sum_{d \in T} \sigma(d) = \frac{1}{|M|} \sum_{i=1}^{|M|} e^i \) and we are done.

We need one more technical result, the Azuma’s inequality:

**Lemma 8.1.** Let \( \{W_n, n \geq 1\} \) be a martingale with mean 0 and suppose that for some non-negative constants: \( \alpha_i, \beta_i \) we have: \(-\alpha_i \leq W_i - W_{i-1} \leq \beta_i \) for \( i = 2, 3, \ldots \). Then for any \( n \geq 0, a > 0 \):

\[
P(W_n \geq a) \leq e^{-\frac{2a^2}{\sum_{i=1}^n (\alpha_i + \beta_i)^2}} \text{ and } P(W_n \leq -a) \leq e^{-\frac{2a^2}{\sum_{i=1}^n (\alpha_i + \beta_i)^2}}.
\]

### 8.2 Majority voting and threshold averaging setting - empirical error

We will now prove parts of theorems 5.3 and 5.4 regarding empirical errors.

**Proof.** Again, we start with the analysis of the threshold averaging. Take \( i^{th} \) random decision tree \( T_i^R \), where \( i \in \{1, 2, \ldots, k\} \). For a given data point \( d \) from the training set let \( X_d^i \) be a random variable defined as follows. If \( d \) does not belong to any leaf of \( T_i^R \) then \( X_d^i = 0 \). Otherwise let \( a^R_i \) be the number of points from the training set with label 0 in that leaf and let \( b^R_i \) be the number of points from the training set with label 1 in that leaf.

If \( d \) has label 0 then we take \( X_d^i = \frac{a^R_i}{a^R_i + b^R_i} \). Otherwise we take \( X_d^i = \frac{b^R_i}{a^R_i + b^R_i} \). Denote \( X^d = \frac{X_1^d + \ldots + X_k^d}{k} \). When from the context it is clear to which data point we refer to we will skip upper index and simply write \( X \) or \( X_i \) respectively.

Fix some point \( d \) from the training set. Note that if \( X > \frac{1}{2} \) then point \( d \) is correctly classified. Notice that the weight of the point \( d \) denoted as \( w_d \) is nothing else but the sum of weights of all the edges of \( G \) incident to \( d \) divided by the number of all trees (or the average weight of an edge indecent to \( d \) if we consider real-valued attributes). Note that we have \( EX = w_d \) and that from Theorem 5.2 we get:

\[
\sum_{d \in T} w_d \geq n(e^2 + (1 - e)^2).
\]

Take \( 0 < \delta < \frac{1}{2} \). Denote by \( \mu \) the fraction of points \( d \) from the training data such that \( w_d \geq \frac{1}{2} + \delta \). From the lower bound on \( \sum_{d \in T} w_d \), we have just derived, we get: \((\frac{1}{2} + \delta)(1-\mu)n + \mu n \geq n(e^2 + (1 - e)^2) \), which gives us:

\[
\mu \geq 1 - \frac{2\epsilon - 2\epsilon^2}{0.5 - \delta},
\]
Differentially- and non-differentially-private random decision trees

where $\epsilon = 1 - \epsilon$.

Take point $d$ from the training set such that $w_d \geq \frac{1}{2} + \delta$. Denote by $p_d$ the probability that $d$ is misclassified. We have:

$$p_d \leq \mathbb{P}(\frac{X_1 + \ldots + X_k}{k} \leq w_d - \delta).$$

Denote: $Z_i = X_i - w_d$ for $i = 1, 2, \ldots, k$. We have:

$$p_d \leq \mathbb{P}(Z_1 + \ldots + Z_k \leq -k\delta).$$

Note that, since $w_d = EX$ and random variables $X_i$ are independent, we can conclude that $\{Z_1, Z_1 + Z_2, \ldots, Z_1 + Z_2 + \ldots + Z_k\}$ is a martingale. Note also that $-\alpha_i \leq Z_i \leq \beta_i$ for some $\alpha_i, \beta_i > 0$ such that $\alpha_i + \beta_i = 1$.

Using Lemma 8.1, we get:

$$\mathbb{P}(Z_1 + \ldots + Z_k \leq -k\delta) \leq e^{-\frac{2(k+1)^2}{k}}.$$

Therefore the probability that at least one of $\mu n$ points $d$ for which $w_d \geq \frac{1}{2} + \delta$ will be misclassified by the set of $k$ random decision trees is, by union bound, at most: $\mu ne^{-2k\delta^2} \leq ne^{-2k\delta^2}$. That, for $k = \frac{(1+C)\log(n)}{2\delta^2}$, completes the proof of the upper bound on the empirical error from theorems 5.3 and 5.4 since we have already proved that $\mu \geq 1 - \frac{2\epsilon - 2\delta^2}{0.5 - 2\delta}$. The proof of the majority voting version goes along exactly the same lines. This time, instead of Theorem 5.2, we use Theorem 5.1. We know that $\sum_{d \in \mathcal{T}} \sigma(d) \geq ne$, where $e = 1 - \epsilon$. Denote the fraction of points $d$ with $\sigma(d) \geq \frac{1}{2} + x$ for $0 < x < \frac{1}{2}$ by $\mu^x$. Then, by the argument similar to the one presented above, we have:

$$\mu^x \geq 1 - \frac{\epsilon}{0.5 - x}. \quad (2)$$

All other details of the proof for the majority voting are exactly the same as for the threshold averaging scheme.

Proof. Let $K > 0$ be a constant. We first consider the threshold averaging scheme. Take a decision tree $T_i$. Denote by $S_{T_i}$ the set of points $d$ from the training set with the following property: point $d$ belongs to $T_i$ do the leaf that contains at least $\frac{n}{K^2}$ points. Note that since each $T_i$ has exactly $2^h$ leaves, we can conclude that $|S_{T_i}| \geq n(1 - \frac{1}{K})$. In this proof and proof of theorems 5.8 and 5.8 (presented in the next section) we will consider graph $\tilde{G}$ that is obtained from $G$ by deleting edges adjacent to those vertices of the color class $B$ that correspond to leaves containing less than $\frac{n}{K^2}$ points from the training set. Take point $d$ from the training set with $w_d' \geq \frac{1}{2} + \delta$, where $w_d'$ is the average weight of an edge incident to $d$ in $G$. Notice that $w_d \geq \frac{1}{2} + \delta + \frac{1}{K}$ implies: $w_d' \geq \frac{1}{2} + \delta$. We say that a decision tree $T_i$ is $d$-good if the leaf of $T_i$ to which $d$ belongs contains at least $\frac{n}{K^2}$ points from the training set. Let us now define $X_i^d$. If $i$th chosen random decision tree is $d$-good then $X_i^d$ is defined as in the proof of Theorem 5.3. Otherwise we put $X_i^d = 0$. Denote $Z_i = X_i^d - w_d^d$. Note that the probability $p_d$ that point $d$ is misclassified by selected random decision trees is $p_d \leq \mathbb{P}(\frac{Z_1 + \ldots + Z_k}{k} + \sum_{j \in \mathcal{I}} \frac{R_j}{|\mathcal{I}|} \leq -\delta)$, where $\mathcal{I}$ is the set of indices corresponding to those chosen random decision trees that are d-good and random variables $R_j$ are correction terms for d-good random decision trees that must be introduced in order to take into account added Laplacians (if $\mathcal{I} = \emptyset$ then we assume that the value of the expression $\sum_{j \in \mathcal{I}} \frac{R_j}{|\mathcal{I}|}$ is 0). Note also that set $\{R_j, Z_j : j = 1, 2, \ldots, k\}$ is a set of independent random variables. We get:

$$p_d \leq \mathbb{P}(\frac{Z_1 + \ldots + Z_k}{k} \leq -\frac{\delta}{2}) + \mathbb{P}(\sum_{j \in \mathcal{I}} R_j \leq -\frac{\delta}{2}).$$

Since from the Azuma’s inequality we get: $\mathbb{P}(\frac{Z_1 + \ldots + Z_k}{k} \leq -\frac{\delta}{2}) \leq e^{-\frac{\delta^2}{2}}$, we have:

$$p_d \leq e^{-\frac{\delta^2}{2}} + \mathbb{P}(\sum_{j \in \mathcal{I}} \frac{R_j}{|\mathcal{I}|} \leq -\frac{\delta}{2}). \quad (3)$$

We will now estimate the expression $p_r = \mathbb{P}(\sum_{j \in \mathcal{I}} R_j \leq -\frac{\delta}{2})$. 


For \( i \in \mathcal{I} \) denote by \( \mathcal{A}_i \) an event that each of the two perturbation errors added to the leaf containing point \( d \) was of magnitude at most \( \frac{\sqrt{\eta n}}{K} \delta_1 \), where \( \delta_1 = \frac{\delta}{24} \). Denote \( \mathcal{A} = \bigcap_{i \in \mathcal{I}} \mathcal{A}_i \). Denote by \( \mathcal{A}^c \) the complement of \( \mathcal{A} \).

We have: 
\[
\mathbb{P}(\sum_{i \in \mathcal{I}} R_j \leq -\frac{\delta}{2}) \leq \mathbb{P}(\sum_{i \in \mathcal{I}} R_j \leq -\frac{\delta}{2}|\mathcal{A}) \mathbb{P}(\mathcal{A}) + \mathbb{P}(\sum_{i \in \mathcal{I}} R_j \leq -\frac{\delta}{2}|\mathcal{A}^c)(1 - \mathbb{P}(\mathcal{A})).
\]

Thus we get:
\[
p_r \leq \mathbb{P}(\sum_{i \in \mathcal{I}} R_j \leq -\frac{\delta}{2}|\mathcal{A}) \leq \frac{\delta}{2} (1 - \mathbb{P}(\mathcal{A})).
\] (4)

Now take one of the chosen random decision trees \( T_i \) with \( i \in \mathcal{I} \). Take its leaf that contains given point \( d \) from the training set. Assume that this leaf contains \( r \) points from the training set with some fixed label \( l \in \{-, +\} \) and that it altogether contains \( n_a \) points. Note that from the definition of \( \mathcal{I} \) we have: \( n_a \geq \frac{n}{K^2} \).

Let \( g_1, g_2 \) be two independent Laplacian random variables, each of density function \( \frac{1}{2\pi e^{\frac{|x|}{2}}} \). We would like to estimate the following random variable \( \Theta = \frac{r+g_1}{n_a+g_1+g_2} - \frac{r}{n_a} \) for an event \( \mathcal{A} \). Note that in particular we know that \( |g_1|, |g_2| \leq \frac{\sqrt{n_a}}{\sqrt{2}} \).

Simple calculation gives us:
\[
|\Theta| \leq \frac{\delta}{4\sqrt{n}}.
\] (5)

Now consider truncated probability space \( \Theta|_{\mathcal{A}} \) and truncated random variables \( R_i^t = R_i|_{\mathcal{A}} \) for \( i \in \mathcal{I} \). We have:
\[
\mathbb{P}(\sum_{i \in \mathcal{I}} R_i^t \leq -\frac{|\mathcal{I}|}{2} |\mathcal{A}) \leq \mathbb{P}(\sum_{i \in \mathcal{I}} R_i^t \leq -\frac{|\mathcal{I}|}{2}).
\]

Using inequality (5) we get:
\[
|R_i^t| \leq \frac{\delta}{4\sqrt{n}}, \quad \mathbb{E}|R_i^t| \leq \frac{\delta}{4\sqrt{n}}.
\] (6)

Thus we can use Azuma’s inequality once more, this time to find the upper bound on the expression: \( \mathbb{P}(\sum_{i \in \mathcal{I}} R_i^t \leq -\frac{|\mathcal{I}|}{2}) \) (we assume here that the random decision trees have been selected thus \( \mathcal{I} \) is given). Without loss of generality we can assume that \( \mathcal{I} \neq \emptyset \). We have:
\[
\mathbb{P}(\sum_{i \in \mathcal{I}} R_i^t \leq -\frac{|\mathcal{I}|}{2}) = \mathbb{P}(\sum_{i \in \mathcal{I}} (R_i^t - ER_i^t) \leq -\frac{|\mathcal{I}|}{2} - \sum_{i \in \mathcal{I}} E R_i^t) \leq \mathbb{P}(\sum_{i \in \mathcal{I}} (R_i^t - ER_i^t) \leq -\frac{|\mathcal{I}|}{4}) \leq e^{-\frac{|\mathcal{I}|}{4\sqrt{n}} + \frac{\delta^2}{4\sqrt{n}}}. \]

Therefore we get:
\[
p_r \leq e^{-\frac{|\mathcal{I}|}{4\sqrt{n}}} + (1 - \mathbb{P}(\mathcal{A})).
\] (7)

It remains to bound the expression: \( (1 - \mathbb{P}(\mathcal{A})) \). Let \( g \) be a Laplacian random variable with density function \( \frac{1}{2\pi e^{\frac{|x|}{2}}} \). Note that from the union bound we get: \( 1 - \mathbb{P}(\mathcal{A}) \leq 2k\mathbb{P}(|g| > \frac{\sqrt{n_a}}{2K^2}) \), where factor 2 in the expression \( 2k\mathbb{P}(g > \frac{\sqrt{n_a}}{2K^2}) \) comes from the fact that for a given data point \( d \) we need to add perturbation error in two places in the leaf of the chosen random decision tree corresponding to \( d \).

Denote \( \gamma = \frac{\delta}{24K^2} \). We have:
\[
p_r \leq e^{-\frac{|\mathcal{I}|}{4\sqrt{n}}} + 4k \int_{\gamma}^{\infty} \frac{\eta}{2k} e^{-\frac{\eta x}{4\sqrt{n}}} dx.
\] (8)

Evaluation of the RHS-expression gives us:
\[
p_r \leq e^{-\frac{\gamma}{2}} + 2ke^{-\frac{\lambda \sqrt{n}}{4}}, \quad \text{where} \quad \lambda = \frac{\delta}{24K^2}.
\] (9)

Thus we can conclude that the probability \( p_d \) that the fixed point \( d \) from the training set will be misclassified by the set of \( k \) randomly chosen random decision trees satisfies:
\[
p_d \leq e^{-\frac{\gamma}{2}} + e^{-\frac{\lambda \sqrt{n}}{4}} + 2ke^{-\frac{\lambda \sqrt{n}}{2}}.
\] (10)

Note that by the similar argument to the one presented in the proof of Theorem 5.3 and Theorem 5.4 we can conclude that at least \( n(1 - \frac{2(\epsilon + \frac{1}{2})^2}{\eta} - 2(\epsilon + \frac{1}{2})^2) \) points \( d \) from the training data satisfy: \( w_d' \geq \epsilon + \frac{1}{2} + \delta \). Let \( \mu' \) be a fraction of points with this property. As we observed earlier, if the points \( d \) satisfies: \( w_d \geq \epsilon + \frac{1}{2} + \delta + \frac{1}{K} \) then it
also satisfies: \( w_d' \geq \frac{1}{2} + \delta \). Thus \( \mu \geq \mu' \). We also have: \( \mu' \geq 1 - \frac{2(\epsilon + \frac{1}{10}) - 2(\epsilon + \frac{1}{10})^2}{\frac{1}{2} + \delta} \). Thus \( \mu \geq 1 - \frac{2(\epsilon + \frac{1}{10}) - 2(\epsilon + \frac{1}{10})^2}{\frac{1}{2} + \delta} \).

We replace \( \epsilon \) by \( \epsilon + \frac{1}{K} \) in the formula derived in the proof of Theorem 5.3, since now for any fixed decision tree we do not take into account points that belong to leaves with less than \( \frac{n}{K} \) points from the training set. For every given decision tree \( T_i \) there are at most \( \frac{n}{K} \) points \( d \) from the training set such that \( T_i \) is not \( d \)-good. Note that, by union bound, the probability that at least one from the \( n \mu \) points \( d \) with \( w_d' \geq \frac{1}{2} + \delta \) is misclassified is at most \( n \mu p_d \). To see how Theorem 5.7 and the part of Theorem 5.6 regarding empirical error follow now, take \( K = 40 \) and \( \delta = \frac{1}{10} \). The proof of the majority voting version is very similar. We use inequality 2 (that was derived from Theorem 5.4) but all other details are exactly the same. Therefore we will not give it in details here since it would basically mean copying almost exactly the proof that we have just showed.

8.3 Probabilistic averaging setting - empirical error

Let us switch now to the probabilistic averaging setting. In practice, as was shown in the experimental section, it is the least effective method. However for the completeness of our theoretical analysis and since for very large datasets theoretical guarantees regarding also this setting can be obtained, we focus on it now.

We will first focus on the part of Theorem 5.5 regarding empirical error.

**Proof.** We already know that: \( \sum_{d \in T} w_d \geq n(\epsilon^2 + (1 - \epsilon)^2) \), where \( \epsilon \) is the average quality. Assume that \( k \) random decision trees have been selected. Denote by \( Y_d \) the indicator of the event that a fixed data point \( d \) from the training set will be correctly classified. We have:

\[
Y_d = \begin{cases} 
1 & \text{with probability } X^d \\
0 & \text{with probability } 1 - X^d,
\end{cases}
\]

where \( X^d \) is random variable defined in the proof of theorems 5.3 and 5.4. Note that after random decision trees have been selected, \( X^d \) has a deterministic value. Note also that random variables \( Y_d \) are independent and \( EY_d = X^d \). Thus, we can use Lemma 8.1 in the very similar way as in the proof of theorems 5.3 and 5.4 to get that for any given \( c > 0 \):

\[
P(\sum_{d \in T} (Y_d - X^d) \leq -nc) \leq e^{-2nc^2}.
\]

(11)

Let us focus now on the process of choosing random decision trees. Fix parameter \( \delta > 0 \). Fix some point \( d \) from the training set. Using Lemma 8.1 in exactly the same way as in the proof of theorems 5.3 and 5.4 we conclude that \( P(X^d < w_d - \delta) \leq e^{-2\delta k^2} \). Therefore, by the union bound, with probability at least \( (1 - ne^{-2\delta k^2}) \) we have: \( \sum_{d \in T} X^d \geq \sum_{d \in T} (w_d - \delta) \). Thus, according to the lower bound for \( \sum_{d \in T} w_d \) presented at the beginning of the proof, we get that with probability at least \( (1 - ne^{-2\delta k^2}) \) the following holds: \( \sum_{d \in T} X^d \geq n(1 - 2\epsilon + 2\epsilon^2 - \delta) \), where \( \epsilon = 1 - c \). Note that random variables \( Y_d \) are independent from random variables \( X_d \). We can conclude, using inequality 11 that with probability at least \( (1 - ne^{-2\delta k^2}) (1 - e^{-2nc^2}) \) at least \( n(1 - 2\epsilon + 2\epsilon^2 - \delta - c) \) points will be correctly classified. Now we can take \( k = \frac{(1+c)\log(n)}{2d^2} \) and that completes the proof. Again, as in the previous proof, the majority voting scheme requires only minor changes in the presented proof so we will leave to the reader.

Let us focus now on parts of theorems 5.8 and 5.9 regarding empirical errors.

**Proof.** Proofs of statements regarding empirical errors go along exactly the same lines as presented proof of the part of Theorem 5.5 (regarding empirical error). The changes in the statement, due to the added perturbation error, follow from the proof of bounds on the empirical error from theorems 5.6 and 5.7. Therefore we will not give the entire proof but only mention few things.

In comparison with the statement of Theorem 5.5 in the expression on the upper bound on empirical error the term \( \epsilon \) is replaced by \( \epsilon + \frac{1}{K} \). This is, as explained in the proof of Theorem 5.6 (regarding empirical error), due to the fact that while dealing with weights of edges in graph \( G_d^D \) we do not take into account points from the training set corresponding to leaves with too few data points. To see how Theorem 5.9 can be derived, take \( K = 40, \delta = \frac{1}{10}, c = \frac{1}{20} \). Again, as for Theorem 5.7 Theorem 5.9 follows now by simple calculations.
8.4 Generalization error

We will now prove upper bounds regarding generalization error for all the theorems presented in the previous paragraphs. We do it for all of them in the same section since all the proofs are very similar. Besides, right now, when we have already developed tools for obtaining upper bounds on the empirical error, we can use them to simplify our analysis regarding generalization error. Random decision trees give strong bounds on the generalization error since they do not lead to data overfitting. The internal structure of each constructed tree (i.e. the set of its inner nodes) does not depend at all on the data. This fact is crucial in obtaining strong guarantees on the generalization error. All the experiments presented in the main body of the paper measured generalization error of the random tree approach and stand for the empirical verification that this method is a good learning technique in the setting requiring high privacy guarantees. Below is the proof of the presented upper bounds on the generalization error.

Proof. Consider test set of $n$ points. Whenever we refer to the weight or goodness of the test point $d$, this is in respect to the test set (see: definition of goodness and other terms in the description of the model). Let $\phi > 0$ be a small constant and denote by $E_\phi$ an event that for the selected forest $F$ of random decision trees the non-perturbed counts in all leafs (for each leaf we count points with label + and − in that leaf) for the test set and training set differ by at most $2\phi n$. We start by finding a lower bound on $P(E_\phi)$. Let us fix a forest, a particular tree of that forest and a particular leaf of that tree. Denote by $X_t$ a random variable that takes value 1 if $i$th point of the training set corresponds to that leaf and 0 otherwise. Similarly, denote by $Y_t$ a random variable that takes value 1 if $i$th point of the test set corresponds to that leaf and 0 otherwise. Denote by $X_t^+$ a random variable that takes value 1 if $i$th point of the training set corresponds to that leaf and has label + and is 0 otherwise. Similarly, denote by $Y_t^+$ a random variable that takes value 1 if $i$th point of the test set corresponds to that leaf and has label + and is 0 otherwise. Denote by $p_1$ the probability that $i$th point of the training/test set corresponds to that leaf and by $p_2$ the probability that $i$th point of the training/test set corresponds to that leaf and has label +. Notice that $p_1$, $p_2$ are the same for the training and test set since we assume that training and test set are taken from the same distribution. Since all the random variables introduced above are independent, we can conclude using Azuma’s inequality that

$$P(X_1 + ... + X_n \in [n(p_1 - \phi), n(p_1 + \phi))] \geq 1 - 2e^{-2n\phi^2}.$$ 

Similarly,

$$P(Y_1 + ... + Y_n \in [n(p_1 - \phi), n(p_1 + \phi))] \geq 1 - 2e^{-2n\phi^2}.$$ 

Therefore, by the union bound

$$P((X_1 + ... + X_n) - (Y_1 + ... + Y_n) \leq 2\phi n) \geq 1 - 4e^{-2n\phi^2}.$$ 

By the same analysis we can show that

$$P(X_1^+ + ... + X_n^+ \in [n(p_2 - \phi), n(p_2 + \phi)]) \geq 1 - 2e^{-2n\phi^2}$$ 

and

$$P(Y_1^+ + ... + Y_n^+ \in [n(p_2 - \phi), n(p_2 + \phi)]) \geq 1 - 2e^{-2n\phi^2}.$$ 

Thus we also have

$$P((X_1^+ + ... + X_n^+) - (Y_1^+ + ... + Y_n^+) \leq 2\phi n) \geq 1 - 4e^{-2n\phi^2}.$$ 

We can conclude that the probability of the following event:

$$|(X_1 + ... + X_n) - (Y_1 + ... + Y_n)| \leq 2\phi n \quad \text{and} \quad |(X_1^+ + ... + X_n^+) - (Y_1^+ + ... + Y_n^+)| \leq 2\phi n$$

is at least $1 - 8e^{-2n\phi^2}$. If we now take the union bound over all $2^hK$ leafs of the forest then we obtain: $P(E_\phi) \geq 1 - 2^{h+3K}e^{-2n\phi^2}$. We will now consider average weights $w_d$ of the test points. The analysis for the majority voting uses $\sigma(d)$ and is completely analogous. Assume now that all the counts for all the leaves for the test and training set differ by at most $2\phi$. As in the analysis of the empirical error in the differentially-private setting, lets focus on those leaves of the forest that contain at least $\frac{n}{2^K}$ of the test points each, for a constant $K > 0$. Take a leaf $l$ with this property. Denote by $x^1$ the number of test points corresponding to that leaf and with label +. Denote by $x^2$ the number of training points corresponding to that leaf and with label +. Denote by $y^1$ the number of
all test points corresponding to that leaf and by \( y^2 \) the number of all training points corresponding to that leaf. We want to find an upper bound on the expression \( q = |x_1^1 - x_2^2| \). Simple algebra gives us: \( q \leq \frac{2\phi n(x_1^1 + y_1)}{y_1^1 - 2\phi n} \). If we now take \( \zeta = \frac{2\phi}{\theta} \), where \( \theta = \frac{1}{2\phi n} \), then we get: \( q \leq \frac{2\phi}{y_1^1 - \theta} \). Let us take \( \zeta \) such that: \( \frac{2\phi}{y_1^1 - \theta} \leq \frac{\delta}{2} \), where \( \delta > 0 \) is a positive constant. Thus we want: \( \zeta \leq \frac{\delta}{2\phi n} \). Take \( \phi = \frac{\delta}{2\phi n + \delta} \). We can conclude that with probability at least \( P(\mathcal{E}_0) \) the difference between ratios of counts in leaves containing at least \( n_\theta \) test points for the test and training set is at most \( \frac{\delta}{2} \). This in particular implies that if we consider test point \( d \) and a truncated bipartite graph \( G^d \) (but this time with respect to the test set, not training set) then weights of \( d \) in \( G^d \) and its corresponding version for the training set differ by at most \( \frac{\delta}{2} \).

We are almost done. Consider first majority voting/threshold averaging scheme. The only changes we need to introduce in the statement of Theorem 5.3 for the empirical error is to subtract from \( p_1 \) the probability that \( \mathcal{E}_0 \) does not hold to obtain a lower bound on \( p_2 \), add factor \( \frac{1}{K} \) to the expression on \( w \) (since we are using the truncated model) and change \( \delta \) by \( \frac{\delta}{2} \) in the expression on number of random decision trees used. Similarly, in the statement of Theorem 5.4 we need to replace \( \epsilon \) in the expression on \( err_1 \) by \( \epsilon + \frac{1}{K} \) to obtain an upper bound on \( err_2 \) (again, because we are using truncation argument) and make the same change in the number of decision trees as the one above. To obtain a lower bound on \( p_2 \) it suffices to subtract the probability that \( \mathcal{E}_0 \) does not hold. Let us focus now on Theorem 5.6. Again we need to add extra factor \( \frac{1}{K} \) to the expression on \( w \) and subtract probability that \( \mathcal{E}_0 \) does not hold to obtain a lower bound on \( p_2 \).

Now lets consider probabilistic averaging scheme. Take the statement of Theorem 5.5 first. We make similar correction to those mentioned earlier to get a lower bound on \( p_2 \). Besides in the upper bound on \( err_1 \) we need to replace \( \epsilon \) by \( \epsilon + \frac{1}{K} \) to obtain an upper bound on \( err_2 \). In Theorem 5.8 we need to add one extra term \( \frac{1}{K} \) in the upper bound on \( err_1 \) to obtain an upper bound on \( err_2 \) and again modify \( p_1 \) in the same way as before to obtain a lower bound on \( p_2 \).

9 Experiments on all datasets

In this section we enclose the experimental results we obtained for all benchmark datasets. The plots have similar form to the ones shown on Mushrooms dataset in the main body of the paper. The presented results further support the claims of the paper.

Figure 3: adult dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. \( \eta = 1000/n_{tr} \). Test error resp. vs. \( h \) across various settings of \( k \) and vs. \( k \) across various settings of \( h \); Minimal, average and maximal test error resp. vs. \( h \) (b)) and vs. \( k \) (d)) for dpRFMV.
Differentially- and non-differentially-private random decision trees

Figure 4: *adult* dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. a) Test error vs. $\eta$ for two settings of $h$. b) Test error vs. $k$ for fixed $h$ and across different settings of $\eta$.

Figure 5: *ban_aut* dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. $\eta = 1000/n_{tr}$. Test error resp. vs. $h$ across various settings of $k$ and vs. $c$) $k$ across various settings of $h$; Minimal, average and maximal test error resp. vs. $h$ (b)) and vs. $k$ (d)) for dpRFMV.

Figure 6: *ban_aut* dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. a) Test error vs. $\eta$ for two settings of $h$. b) Test error vs. $k$ for fixed $h$ and across different settings of $\eta$. 

Differentially- and non-differentially-private random decision trees

Figure 7: BTSC dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. $\eta = 1000/n_{tr}$. Test error resp. vs. a) $h$ across various settings of $k$ and vs. c) $k$ across various settings of $h$; Minimal, average and maximal test error resp. vs. $h$ (b)) and vs. $k$ (d)) for dpRFMV.

Figure 8: BTSC dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. a) Test error vs. $\eta$ for two settings of $h$. b) Test error vs. $k$ for fixed $h$ and across different settings of $\eta$.

Figure 9: CVR dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. $\eta = 1000/n_{tr}$. Test error resp. vs. a) $h$ across various settings of $k$ and vs. c) $k$ across various settings of $h$; Minimal, average and maximal test error resp. vs. $h$ (b)) and vs. $k$ (d)) for dpRFMV.
Differentially- and non-differentially-private random decision trees

Figure 10: CVR dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. a) Test error vs. \( \eta \) for two settings of \( h \). b) Test error vs. \( k \) for fixed \( h \) and across different settings of \( \eta \).

Figure 11: Mam\(\_M \) dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. \( \eta = 1000/\eta_{\text{tr}} \). Test error resp. vs. a) \( h \) across various settings of \( k \) and vs. c) \( k \) across various settings of \( h \); Minimal, average and maximal test error resp. vs. \( h \) (b)) and vs. \( k \) (d)) for dpRFMV.

Figure 12: Mam\(\_M \) dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. a) Test error vs. \( \eta \) for two settings of \( h \). b) Test error vs. \( k \) for fixed \( h \) and across different settings of \( \eta \).
Differentially- and non-differentially-private random decision trees

Figure 13: Covertype dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. \( \eta = 1000/n_{tr} \). Test error resp. vs. a) \( h \) across various settings of \( k \) and vs. c) \( k \) across various settings of \( h \); Minimal, average and maximal test error resp. vs. \( h \) (b)) and vs. \( k \) (d)) for dpRFMV.

Figure 14: Covertype dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. a) Test error vs. \( \eta \) for two settings of \( h \). b) Test error vs. \( k \) for fixed \( h \) and across different settings of \( \eta \).

Figure 15: Quantum dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. \( \eta = 1000/n_{tr} \). Test error resp. vs. a) \( h \) across various settings of \( k \) and vs. c) \( k \) across various settings of \( h \); Minimal, average and maximal test error resp. vs. \( h \) (b)) and vs. \( k \) (d)) for dpRFMV.
Figure 16: Quantum dataset. Comparison of dpRFMV, dpRFTA and dpRFPA. a) Test error vs. $\eta$ for two settings of $h$. b) Test error vs. $k$ for fixed $h$ and across different settings of $\eta$. 