Under manipulations, are some AI models harder to audit?

Augustin Godinot
Univ Rennes, Inria, CNRS, IRISA, PEReN
Rennes, France
augustin.godinot@inria.fr

Erwan Le Merrer
Univ Rennes, Inria, CNRS, IRISA
Rennes, France

Gilles Trédan
LAAS/CNRS
Toulouse, France

Camilla Penzo
PEReN
Paris, France

Franois Taïani
Univ Rennes, Inria, CNRS, IRISA
Rennes, France

***Abstract*—Auditors need robust methods to assess the compliance of web platforms with the law. However, since they hardly ever have access to the algorithm, implementation, or training data used by a platform, the problem is harder than a simple metric estimation. Within the recent framework of *manipulation-proof* auditing, we study in this paper the feasibility of robust audits in realistic settings, in which models exhibit large capacities.**

We first prove a constraining result: if a web platform uses models that may fit any data, no audit strategy—whether active or not—can outperform random sampling when estimating properties such as demographic parity. To better understand the conditions under which state-of-the-art auditing techniques may remain competitive, we then relate the difficulty of audits to the capacity of the targeted models, using the Rademacher complexity. We empirically validate these results on popular models of increasing capacities, thus confirming experimentally that large-capacity models, which are commonly used in practice, are particularly hard to audit robustly. These results refine the limits of the auditing problem, and open up enticing questions on the connection between model capacity and the ability of platforms to manipulate audit attempts.

***Index Terms*—Audit, black-box interaction, Rademacher complexity, model capacity.**

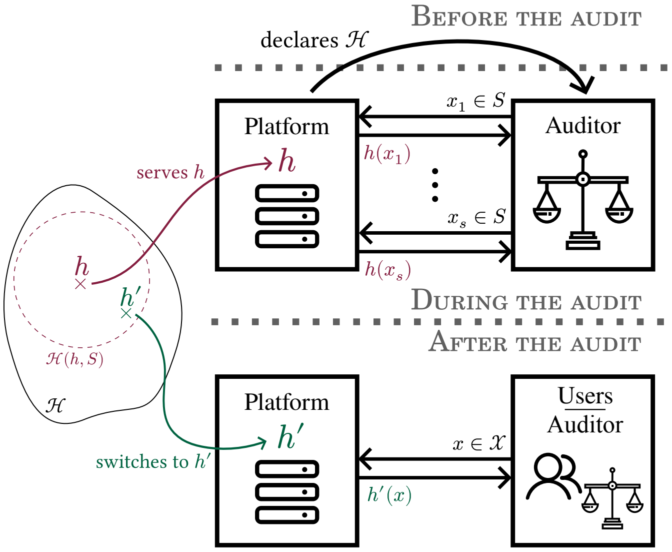

Figure 1. Security game of the manipulation-proof auditing framework. Before the audit, the platform declares the hypothesis space $\mathcal{H}$ to the auditor. During the audit, the platform serves the model $h \in \mathcal{H}$ and the auditor queries h on S. After the audit, the platform can change its model to h' with the constraint that $\forall x \in S, h'(x) = h(x)$ or equivalently, $h' \in \mathcal{H}(h, S)$.

I. Introduction

The pervasive deployment of user-facing automated decision systems raises concerns over their impact on society. The growing number of online platforms and their increasing complexity highlights the need for automated and robust audits to assess their impact on users. The advent of highly publicized audits—such as ProPublica's story on COMPAS [3] or Reuters' study on Amazon's recruiting tool [4]—has brought considerable traction to the AI audit field. For the public to trust Artificial Intelligence (AI) systems, and more broadly algorithmic decision systems, we need methods to explain the decisions of such systems [5], [6], certify their implementation [1], [7] and automatically and robustly detect misconduct [8], [9].

As it is common in the literature (e.g. see [1]), we assume that the system is composed of a *trained machine learning (ML) model* h that the auditor can interact with via a web interface or an Application Programming Interface (API). Similarly to the *honest-but-curious (HBC)* threat model [10], the outputs returned by the API are the actual output of h. However, while the platform cannot arbitrarily directly modify the output of h, it can use the interactions performed by the auditor during the audit process to acquire as much information as possible and modify the model in its favor. In this work, we focus on *external certification audits*. In this type of audit, an external auditor (e.g. a regulator or an auditing company) needs to verify a given property μ (e.g. the absence of bias) of the API provided by the platform. We will refer to this setting as the *remote black-box auditing* problem.

Most of the current audit methods [11], [12] could be referred to as "detection" audits. This is because they seek to detect whether some rule is being violated either to improve the platform itself or to take legal action. A typical methodology of "detection" audits consists in randomly sampling the input space, computing the measure(s) of interest and declaring the audit failed if the measures cross a given threshold. In this case, to prove the platform's misconduct, one must witness it during the audit. As a result, proving the absence of misconduct would require probing the entire input space of the model h. Since the auditor cannot query the model on its entire input space $\mathcal{X}$, they must choose a subset $S \subseteq \mathcal{X}$, and they must have the guarantee that the estimation on the subset S is not "too far" from the value they would find if they could sample the whole input space.

Threat model: We describe the interaction between the auditor and the platform in the threat model diagram Figure 1. Before the audit, the platform discloses the hypothesis space $\mathcal{H}$ they use (decision trees for example) to the auditor. Then, during the auditing phase, the auditor interacts with the (unknown) model $h \in \mathcal{H}$ exposed by the platform to iteratively build an audit set $S \subset \mathcal{X}$. The manipulation-proof framework acknowledges the possibility for a platform to try to *evade* the audit by showing a fair model h to the auditor, then switching to a more accurate but potentially unfair model h'. The only assumption on how the platform may choose the new model h' is that it should be *consistent* with h. The consistency constraint requires h' to have the same outputs as h on the audit set S, otherwise the auditor could easily check that the platform changed its model after the audit by re-querying it on S. We now formalize the capabilities and knowledge of the platform and the auditor in the manipulation proof framework.

- *Auditor capabilities*: The auditor can send adaptive queries to the platform to build an audit set $S \subset \mathcal{X}$.
- *Auditor knowledge*: The auditor knows the hypothesis class $\mathcal{H}$ implemented by the platform and the value of the sensitive attribute x_A of all the points in the input space $\mathcal{X}$. However, the auditor does not know the specific hypothesis $h \in \mathcal{H}$ implemented by the platform.
- *Platform capabilities*: The platform can change its model from $h \in \mathcal{H}$ to $h' \in \mathcal{H}$ after the audit as long as h' respects the consistency constraint $\forall x \in S, h(x) = h'(x)$.
- *Platform knowledge*: The platform knows the property μ (e.g. Demographic Parity) being measured by the auditor. As the auditor, it knows the value of the sensitive attribute x_A of all the points in the input space $\mathcal{X}$.

Problem: Among the attempts at formalizing robust auditing [1], [2], [13], Yan and Zhang [1] have shown that the knowledge of the hypothesis class used by the platform can *potentially* reduce the required number of audit queries to reach a given robustness level. Their method is based on disagreement-based active learning [14] which requires training surrogates of the platform's model. However, they only demonstrated their proposed audit algorithm with linear models on small datasets (`StudentPerf` [15] and `COMPAS` [3]). Furthermore, they prove that quantifying the potential improvement (in terms of query complexity) of their algorithm over a simple random baseline is computationally intractable. Thus, whether it is possible or not to devise practical robust auditing methods still remains an open question.

Our exploration of robust audits for practical models is focused on binary classifiers and binary sensitive attributes. While this calls for future work on other tasks and modalities, this first exploration covers a large class of decision systems based on ML algorithms [16]. Our hope is to demonstrate that regulators should be given more than black-box access to AI models as part of the audit procedure.

Contributions: In this work, we investigate whether the platform can engineer models that simultaneously achieve a high utility and evade the audit. To that end, we compare the manipulation-proofness (MP) guarantees of a simple uniform random audit algorithm (Algorithm 1) against the best guarantees a regulator could hope for. Our contributions are threefold.

1) We first consider those hypothesis classes that can perfectly reproduce any labeling of the dataset. This covers two practical cases: either the platform has a model with a very high capacity, or the auditor's prior on the platform's model is uninformative. We prove in Theorem 1 (Subsection III-A) that no audit method—whether active or passive—can deliver a better performance than random sampling. We also prove in Corollary 1 that this impossibility holds even if the hypothesis class can only imperfectly reproduce any labeling of the dataset with a bounded error rate.
2) To uncover what properties of the hypothesis class influence its auditability, in Subsection III-B we analyze the simple class of dictionary models, whose manipulation guarantees can be analytically derived. We identify regimes in which the hypothesis class cannot be audited more efficiently than by random sampling.
3) To build a practical understanding of our theoretical results, we formally define the notion of *audit difficulty* and *capacity* in subsection IV-A. We then evaluate the audit difficulty of classical ML models for tabular data. We empirically confirm the strong connection between the classical *Rademacher complexity* and the difficulty of manipulation-proof auditing. Since modern ML hypothesis classes tend to exhibit larger and larger capacities, we argue that our work brings up the limits of the current formulation of manipulation-proof auditing.

II. Auditing and manipulation-proof estimation

During a typical audit, the auditor defines a measure of interest μ with an associated threshold τ_μ. Classical measures used by auditors are statistical parity indicators [17] focusing on independence (e.g. demographic parity, group fairness), separation (e.g. balance for positive/negative class, equalized odds) and sufficiency (e.g. calibration, predictive parity). Given that demographic parity does not require any ground truth labels and since it is often used as the archetypal example in the literature, we use it as the measure μ throughout this paper. While the results we present refer specifically to demographic parity, it is straightforward to extend them to any parity measure of the form

$$\begin{aligned}\mu(h,S) = &\mathbb{P}\left(h(X)=1|X\in S,E\right) \\ &-\mathbb{P}\left(h(X)=1\middle|X\in S,\overline{E}\right)\end{aligned} \tag{1}$$

with E an event defined with respect to the random variables X, X_A and Y, where X represents the input, Y the ground truth label, and $X_A \in \{0,1\}$ is the sensitive attribute of interest for the auditor. For example, for demographic parity, $E = (X_A = 1)$. We would like to stress that for other less common measures that can nonetheless present an interest for auditors (e.g. level of privacy [18] or the degree of compliance with data minimization [9]), manipulation proof auditing remains an open problem.

A. Machine Learning notations

Except when noted, we will consider a binary classification task as in [19], with finite *input space* $\mathcal{X}$ and output space $\mathcal{Y} = \{0,1\}$.[1] $\mathcal{Y}^{\mathcal{X}}$ denotes the space of functions $\mathcal{X} \to \mathcal{Y}$. For any sample $x \in \mathcal{X}$, we refer to its sensitive attribute (e.g., gender, ethnicity, religion) as $x_A \in \{0,1\}$. The sensitive attribute of the points in $\mathcal{X}$ induces a partition of the input space. We note $\mathcal{X}_A = \{x \in \mathcal{X} : x_A = 1\}$ and remark that $\mathcal{X}_{\overline{A}} = \overline{\mathcal{X}_A}$. For any set V, $\mathcal{P}(V)$ denotes the set of all subsets of V and $\mathcal{U}(V)$ denotes the uniform distribution on V. By training the classification model, the platform effectively chooses a model h in some hypothesis class $\mathcal{H}$. The auditor defines a measure $\mu : \mathcal{H} \times \mathcal{P}(\mathcal{X}) \to \mathbb{R}_+$, which is known by the platform. For any subset $V \subseteq \mathcal{H}$ and $S \subseteq \mathcal{X}$, we define *the diameter of V with respect to the measure μ* as

$$\operatorname{diam}_{\mu(\cdot,S)} V = \max_{h,h'\in V} |\mu(h,S) - \mu(h',S)|,$$

when S is the entire input space $\mathcal{X}$, we abuse the notation and write $\operatorname{diam}_{\mu(\cdot,\mathcal{X})} V = \operatorname{diam}_\mu V$. Finally, define for any subset $V \subset \mathcal{H}$, sample $x \in \mathcal{X}$ and label $y \in \{0,1\}$ the set $V[x,y] = \{h \in V : h(x) = y\}$. The cost $\operatorname{Cost}(V)$ of a subset $V \subset \mathcal{H}$ is defined in Equation 2. Note that when the context is clear, we elide the ϵ for simplicity.

$$\operatorname{Cost}_\epsilon(V) = \begin{cases} 0 \text{ if } \operatorname{diam}_\mu V < \epsilon \\ 1 + \min\limits_{x\in\mathcal{X}} \max\limits_{y\in\{0,1\}} \operatorname{Cost}_\epsilon(V[x,y]) \text{ else}\end{cases} \tag{2}$$

Before we formally define the *capacity* of a hypothesis class in subsection IV-B, we will use the term capacity loosely. Intuitively the capacity of a hypothesis class $\mathcal{H}$ is related to the ability for any labeling of the input space $\mathcal{X}$ to find a hypothesis $h \in \mathcal{H}$ that realizes this labeling. More details on the notion of capacity can be found in section VI.

B. What is an active auditing algorithm?

An audit algorithm $\mathcal{A}$ with label budget s is a sequence of (possibly randomized) $s+1$ functions $(f_0, \ldots, f_s)$. For each iteration i, the function $f_i : (\mathcal{X} \times \{0,1\})^{i+1} \to \mathcal{X}$ chooses the next sample $x_i = f_i\left((x_0, h(x_0)), \ldots, (x_{i-1}, h(x_{i-1}))\right)$ to query and add to the audit set. After the query budget has been spent, the end result of the algorithm is the audit set $S = \mathcal{A}(h)$. Note that most published black-box audits of web platforms are not active [20]. In this case, an audit algorithm reduces to a single (possibly randomized) function f_s which does not depend on the answers provided by the platform.

C. The manipulation-proof auditing framework

Following the framework of Yan and Zhang [1], the platform is assumed to be *self-consistent*, i.e. when the platform returns a given output $y = h(x)$ to an auditor's query x, the platform commits to this value and cannot return a different answer $y' = h(x)$ if x is queried again at a later moment in time. Furthermore, as explained in the threat model Figure 1, it is assumed that the auditor knows the *hypothesis class* $\mathcal{H} \subseteq \{0,1\}^{\mathcal{X}}$ of the model implemented by the platform. The self-consistency of the platform together with the knowledge of the hypothesis class defines a subset of "plausible" models in $\mathcal{H}$ that have the same answers as the platform on the current audit set S. This subset is called the *version space of $\mathcal{H}$ induced by S and h* [14], [21], noted $\mathcal{H}(h,S)$.

$$\mathcal{H}(h,S) = \{h' \in \mathcal{H} \ : \ \forall x \in S, h'(x) = h(x)\} \tag{3}$$

We assume that the platform seeks to maximize its profits, which is not necessarily aligned with the property that the regulator seeks to enforce. During the audit process, the auditor incrementally builds an audit set $S \subseteq \mathcal{X}$ based on their previous queries and the answers of the platform. The goal of the auditor is to produce an estimate $\widehat{\mu}$ as close as possible to the real value while being robust to the potential manipulations implemented by the platform. We now formulate the two requirements of the *manipulation-proof* auditing problem, as introduced in [1].

Create an algorithm $\mathcal{A}$ with smallest budget s such that,

$$\textit{(fidelity)} \quad |\mu(h,\mathcal{A}(h)) - \mu(h,\mathcal{X})| < \epsilon \tag{4}$$

$$\textit{(manipulation-proofness)} \quad \operatorname{diam}_\mu \mathcal{H}\left(h,\mathcal{A}(h)\right) < \epsilon \tag{5}$$

Fidelity is the classical estimation constraints. It requires the estimated value $\widehat{\mu} = \mu(h^*,S)$ to be close to the real value $\mu(h,\mathcal{X})$. In addition, *manipulation-proofness* requires that if the platform changes its implemented instance from h to h' while respecting the self-consistency constraint $h' \in \mathcal{H}(h,S)$, the difference between the previous $\mu(h,\mathcal{X})$ and new $\mu(h',\mathcal{X})$ values of μ must be bounded. Therefore, the μ-diameter is the biggest change in the value of μ the auditor would accept if the platform changed to another (consistent) hypothesis.

D. Comparing manipulation-proof auditing algorithms

There are two ways to compare two audit algorithms $\mathcal{A}$ and $\mathcal{A}'$. Either fix a target manipulation-proofness guarantee ϵ and

[1] Should $\mathcal{X}$ be infinite, Dasgupta, Hsu, Poulis, *et al.* [19] note that it suffices to sample a finite i.i.d. subset $\widetilde{\mathcal{X}}$ and extend all the following bounds by classical generalization bounds.

Table I
THE QUERY COMPLEXITY OF DIFFERENT AUDITING ALGORITHMS IN THE MANIPULATION-PROOF FRAMEWORK, EXTRACTED FROM YAN AND ZHANG [1]

Algorithm	Query complexity
Random sampling (Algorithm 1)	$\mathcal{O}\left(\frac{1}{\epsilon^2}\log\lvert\mathcal{H}\rvert\right)$
Optimal deterministic [1, Algorithm 1]	$\text{Cost}_\epsilon(\mathcal{H})$
Oracle based approximation (AFA) [1, Algorithm 3]	$\mathcal{O}\left(\log\lvert\mathcal{H}\rvert \log\lvert\mathcal{X}\rvert \,\text{Cost}(\mathcal{H})\right)$

evaluate the number of queries needed by $\mathcal{A}$ and $\mathcal{A}'$, or fix the audit budget s and evaluate the μ-diameter of the audit sets built by $\mathcal{A}$ and $\mathcal{A}'$.

Yan and Zhang [1] focused on the former: the study of the *query complexity* of different audit algorithms. For general hypothesis classes, they introduced three auditing algorithms. The first one is the baseline random audit algorithm. This audit algorithm consists in sampling among points with positive and negative sensitive attributes, and computing the empirical frequencies of the events $(h(X) = 1|X_A = 1)$ and $(h(X) = 1|X_A = 0)$ (see Algorithm 1). To capture the minimal query complexity attainable by deterministic audit algorithms, they introduced a second algorithm based on the recursive minimization of $\text{Cost}(\mathcal{H})$. Finally, Yan and Zhang introduced a third, oracle-based, algorithm that we coin AFA. We summarize the query complexities proved by [1] in Table I.

Motivated by the implementation of MP audit algorithms, we choose to focus on the second comparison approach: fixing an audit budget and evaluating the μ-diameter. This approach is better suited to our situation since in practice, auditors have a limited query budget that would be agreed upon with the platform prior to the audit.

E. The computational complexity of manipulation-proof auditing

As exposed in Table I, the best attainable query complexity, as well as the query complexity of the more practical AFA algorithm depend on the value of $\text{Cost}(\mathcal{H})$. In addition, the computational complexity of AFA [1, Algorithm 1] is the time to train a model from the hypothesis class $\mathcal{H}$ multiplied by the query complexity. However, Yan and Zhang prove that $\text{Cost}(\mathcal{H})$ is hard to compute, hard to approximate and hard to optimize [1, Proposition 3.5]. Thus, not only it prevents practical implementations of the optimal deterministic algorithm, it also prevents practical analysis of the query complexity and computational complexity of AFA for large models that are costly to train.

III. THE COMPETITIVE EFFECTIVENESS OF RANDOM AUDITS

Current state-of-the-art models for tabular data (see Figure 5) and image data (see e.g. [22]) are able to fit very large train sets with close to perfect accuracy while retaining good generalization properties. In our setting this would mean that these models can represent any binary classification function $f : \mathcal{X} \to \{0, 1\}$ of the input space. As we saw in subsection II-E, the only tractable algorithm (AFA [1]) that was proposed to solve the manipulation-proof auditing task (Equation 4 and 5) is still too computationally intense to audit large models because it requires to be able to train a lot of copies efficiently. Moreover, while [1] experimented on small datasets with linear models, there exists no implementations or practical experiments on larger models. Thus, the potential gains brought by AFA are hard to predict. Yet, for AFA to be used in practice, it would be necessary to balance the extra cost induced by auditing with AFA with the added guarantees of AFA. Thus, a natural practical question arises. **Is the added manipulation-proofness guarantee worth paying the computational toll?**

To answer this question, instead of analyzing $\text{Cost}(\mathcal{H})$ (which is hard to compute and derive) as [1], we directly express the value of $\text{diam}_\mu\,\mathcal{H}(h, S)$ for specific hypothesis classes. Identifying hypothesis classes $\mathcal{H}$ wherein the value of $\text{diam}_\mu(h, S)$ remains constant across all audit sets S allows us to find scenarios in which enhancing manipulation-proofness guarantees beyond that of a random baseline is impossible.

In this section, we consider three typical but insightful forms of hypothesis space $\mathcal{H}$ to better understand this balance between computational cost and added robustness. We prove in subsection III-A that for hypothesis classes shattering the whole input space, all the audit algorithms have the same performance as random sampling. Next, to understand what happens for classes that are only able to fit a part of the dataset, we consider the illustrative class $\mathcal{D}_m$ of dictionaries of size m. We derive the exact value of their μ-diameter in subsection III-B and show the link between the memory as an intuitive notion of the capacity and the MP guarantees obtainable when auditing dictionary models. Last but not least, building on the results of subsection III-A and subsection III-B, we introduce a formal notion of the capacity of a binary classification hypothesis class as the maximum number of samples a platform can interpolate while still retaining good generalization performance. Under this definition, we prove in subsection III-C that large capacity models cannot be audited more efficiently than by the random baseline.

Algorithm 1 A random sampling audit strategy

```
Require: Proportions $\beta_1, \beta_2$, budget $s$
Ensure: audit dataset $S$ with $|S| = s$
  $s^+ \leftarrow \lfloor \beta_1 |\mathcal{X}_A| \rfloor$, $s^- \leftarrow \lfloor \beta_2 |\overline{\mathcal{X}_A}| \rfloor$
  $S^+ \leftarrow$ sample $s^+$ points in $\mathcal{X}_A$ without replacement
  $S^- \leftarrow$ sample $s^-$ points in $\overline{\mathcal{X}_A}$ without replacement
  $S = S^+ \sqcup S^-$
```

A. Hypothesis classes that can fit the dataset entirely

To build intuition on the following theorems, let us first consider classes able to fit any distribution on $\mathcal{X}$. This corresponds to the case of a platform with a very large, overparametrized hypothesis class $\mathcal{H}$ able to fit any labeling of the whole input space $\mathcal{X}$. [2] This assumption is equivalent to considering the hypothesis class $\mathcal{H} = \{0,1\}^{\mathcal{X}}$. Because all the functions from the input space to the output space are possible, the answer of the platform on a query x does not give any information on the possible answers to the other queries in $\mathcal{X}$. It follows, that no matter how the points are iteratively chosen, only the number of points (and the value of their associated sensitive attribute) will matter in the computation of the μ-diameter. We now formalize this intuition.

Theorem 1 (No need to aim)**.** *Let $\mathcal{H} = \{0,1\}^{\mathcal{X}}$. For any audit set $S \subseteq \mathcal{X}$ and hypothesis $h \in \mathcal{H}$,*

$$\operatorname{diam}_\mu \mathcal{H}(h,S) = 2 - \big(\mathbb{P}\left(X \in S | X_A = 1\right) + \mathbb{P}\left(X \in S | X_A = 0\right)\big)$$

Proof sketch. The first step in proving Theorem 1 relies on the fact that all the instances $h' \in \mathcal{H}\left(h,S\right)$ have the same value of $\mu(h',S)$. After decomposing the μ-diameter on S and $\overline{S}$, we use this fact to separate the μ-diameter into the difference between a maximization and a minimization problem. The optima of these problems rely on the existence of hypotheses $h^{\uparrow}, h^{\downarrow} \in \mathcal{H}\left(h,S\right)$ that exactly fit the sensitive attribute (resp. its negation) on $\overline{S}$. Since $\mathcal{H}$ is the space of all functions, it is always possible to find such $h^{\uparrow}$ and $h^{\downarrow}$. Finally, we find these optima and simplify their expressions to reach that of Theorem 1. A complete proof is provided in Appendix A. □

The values $\mathbb{P}\left(X \in S | X_A = 1\right)$ and $\mathbb{P}\left(X \in S | X_A = 0\right)$ are aggregated quantities that depend only on the relative proportion of sensitive ($x_A = 1$) and non-sensitive ($x_A = 0$) samples in the audit set S. Therefore, for any pair $(\mathbb{P}\left(X \in S | X_A = 1\right), \mathbb{P}\left(X \in S | X_A = 0\right))$, one can design a random sampling scheme that achieves the desired relative proportions. We expose such algorithm in Algorithm 1. Since the auditor by definition knows the sensitive attribute of each sample, the idea is to sample points from $\mathcal{X}_A$ and $\mathcal{X}_{\overline{A}}$ with the right proportions (β_1, β_2) in S_{random}. Setting $(\beta_1, \beta_2) = (\mathbb{P}\left(X \in S | X_A = 1\right), \mathbb{P}\left(X \in S | X_A = 0\right))$ in Algorithm 1 yields $(\mathbb{P}\left(X \in S_{\text{random}} | X_A = 1\right), \mathbb{P}\left(X \in S_{\text{random}} | X_A = 0\right)) = (\mathbb{P}\left(X \in S | X_A = 1\right), \mathbb{P}\left(X \in S | X_A = 0\right))$. Following Theorem 1, any audit set S with the same relative proportions $(\mathbb{P}\left(X \in S | X_A = 1\right), \mathbb{P}\left(X \in S | X_A = 0\right))$ yields the same μ-diameter. Since any couple $(\mathbb{P}\left(X \in S | X_A = 1\right), \mathbb{P}\left(X \in S | X_A = 0\right))$ is also attainable by the random sampling algorithm described in Algorithm 1, **when the hypothesis class can perfectly fit any arbitrary label distribution, all audit algorithms –active or not– have at most the same manipulation-proofness guarantees as random sampling**.

As a side note, removing the assumption that the auditor knows the hypothesis class implemented by the platform is equivalent to assuming $\mathcal{H} = \{0,1\}^{\mathcal{X}}$. In this sense, by proving that random sampling is optimal when the hypothesis class is unknown, Theorem 1 demonstrates that knowing $\mathcal{H}$ is necessary (but not sufficent) to design more efficient manipulation-proof auditing methods.

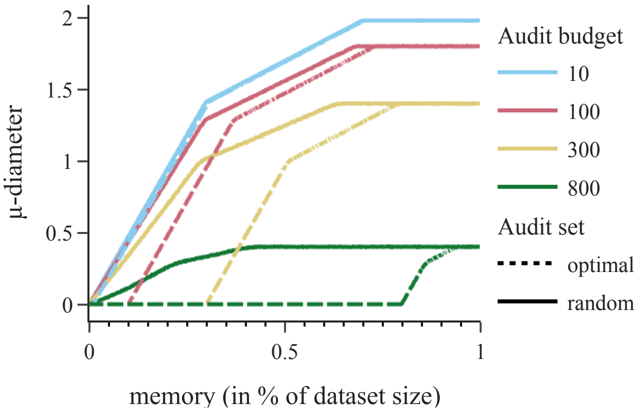

Figure 2. The diameter (vertical axis) resulting from the amount of memory (horizontal axis) of the dictionary model studied in subsection III-B. The various audit budgets are represented by different curve colors, while the optimal audit set appears as dashed curves, and the random baseline audit sets as plain lines.

B. An illustrative example with dictionaries

It is unlikely in practice that any hypothesis class can fit the entire input space $\mathcal{X}$. We now relax this assumption to pursue our analysis of the achievable manipulation-proofness guarantees of models with a large capacity. To that end, we introduce the class $\mathcal{D}_m$ of dictionary models. A dictionary $d \in \mathcal{D}_m$ is built by choosing a set of $m \in [\![n]\!]$ samples in $\mathcal{X}$ and storing the corresponding labels. When the dictionary is asked to label a sample that it did not store, it returns 0 as a default value. Define, for any set of vectors $V \subseteq \mathbb{R}^d$, $\mathfrak{S}(V)$ the set of vectors obtained from V by including all permutations of the coefficients of each $v \in V$. The hypothesis class of dictionaries of memory m is formally introduced in Definition 1.

Definition 1 (Dictionary hypothesis class)**.** *Consider an input space $\mathcal{X}$, $n = |\mathcal{X}|$. The class of dictionaries of memory $m \in [\![n]\!]$ is defined as*

$$\mathcal{D}_m = \mathfrak{S}\left(\{0,1\}^m \times \{0_{\mathbb{R}^{n-m}}\}\right)$$

While such a hypothesis class is not likely to be used in a practical context (as it will typically fail to generalize beyond the encountered examples, exhibiting a blatant overfitting) it is simple enough to support an analysis of the MP guarantees for both randomized and optimal approaches. Moreover, its main parameter (the memory m) directly influences its capacity.

The exact value of the μ-diameter of dictionary hypothesis classes is exposed in Theorem 2. The proof can be found in Equation A.

[2] This does not contradict the No Free Lunch theorem since here, the input space $\mathcal{X}$ is finite.

Theorem 2 (μ-diameter of $\mathcal{D}_m$)**.** *Consider $S \subseteq \mathcal{X}$, $d \in \mathcal{D}_m$. Note $m' = m - |x \in S : d(x) = 1|$. The μ-diameter of $\mathcal{D}_m(d, S)$ is given by*

$$\operatorname{diam}_\mu \mathcal{D}_m(d, S) = \frac{\min(\left|\mathcal{X}_A \cap \overline{S}\right|, m')}{|\mathcal{X}_A|} + \frac{\min(\left|\overline{\mathcal{X}_A} \cap \overline{S}\right|, m')}{\left|\overline{\mathcal{X}_A}\right|}$$

Proof sketch. The proof relies on the same development of the diameter as in the proof of Theorem 1 but instead of finding $h^{\uparrow}$ and $h^{\downarrow}$, we are able to give the values of the optima thanks to the structure of $\mathcal{D}_m$. □

We are interested in the high memory m, low audit budget $|S|$ regime. In this situation, there exist couples (S, m) such that $\left|\mathcal{X}_A \cap \overline{S}\right| \leq m'$ and $\left|\overline{\mathcal{X}_A} \cap \overline{S}\right| \leq m'$. Thus, in this regime, the μ-diameter does not depend on the labels of the particular dictionary d chosen by the platform. Therefore, as for the case $\mathcal{H} = \{0,1\}^{\mathcal{X}}$, in the high memory, low audit budget regime, all audit algorithms – active or not – have at most the same manipulation-proofness guarantees as random sampling.

Simulation of the impact of memory over diameter: The expression of the μ-diameter exposed in Theorem 2 is piecewise linear in the memory m. To gain intuition, we plot the value of $\operatorname{diam}_\mu \mathcal{D}_m(d, S)$ in Figure 2 for a setting where $|\mathcal{X}| = 1000$, $\mathbb{P}(X_A = 1) = 0.3$ and the μ-diameter of the random strategy is averaged over 100 realisations of S. We first observe the drastic impact of dictionary memory on an audit of a fixed budget: for instance, with an audit budget of 300 (representing nearly one-third of the whole input space) an optimal audit set barely achieves a μ-diameter of 1 when auditing dictionaries with memory $m = 500$. Furthermore, given a fixed audit budget, the gap between randomized and optimal audit sets shrinks as the memory grows. This is especially striking in low audit budget regimes, that correspond to a typical audit situation. Moreover, for an audit budget of 100 and memory values larger than 70% the random and optimal audit strategies have the same μ-diameter. This observation hints that Theorem 1's conclusions should hold for a broader set of hypothesis classes.

C. Tying it all together: large capacity and auditability

We derived in subsection III-B the exact expression of the μ-diameter for toy models able to memorize part of the input space. Motivated by the *benign overfitting* phenomenon [22]–[25], we now consider the case of a hypothesis class that is able to perfectly fit any subset $S \subseteq \mathcal{X}$ of reasonable size, but require in addition that the resulting hypothesis h^* maintains good accuracy on the rest of the dataset.

It has been observed that contrary to common knowledge on the bias-variance tradeoff, large ML models can exhibit good generalization properties while perfectly fitting the train data. This benign overfitting phenomenon (also related to *double descent*), is observed in models that are largely over-parametrized compared to the training data available at hand. Nevertheless, we show in Figure 3 that trees and GBDTs can reach the maximum capacity, indicating that they also can interpolate the training data. Drawing intuition from the empirical characterization of benign overfitting in [22]–[25], we derive the formal definition of a large capacity hypothesis class in Definition 2.

Definition 2 (Benign Overfitting Hypothesis class)**.** *Consider an input space $\mathcal{X}$, a hypothesis class $\mathcal{H} \subset \{0,1\}^{\mathcal{X}}$ and a labeling $c \in \{0,1\}^{\mathcal{X}}$. $\mathcal{H}$ is said to exhibit benign overfitting with respect to labeling c if there exists $d_0 \in \mathbb{N}_*$ and $\epsilon \in [0,1)$ such that*

$$\forall d \leq d_0, S \subseteq \mathcal{X}, \sigma \in \{0,1\}^d, \exists h \in \mathcal{H},$$
$$\begin{cases} \forall x_i \in S, h(x_i) = \sigma_i & \textit{(fit any train set)} \\ \mathbb{P}\left(h(X) = c(X) \middle| X \in \overline{S}\right) = 1 - \epsilon & \textit{(with low error on c)} \end{cases}$$

As is stands, Definition 2 is tightly linked to the notion of version space. If $\mathcal{H}$ exhibits overfitting, we are guaranteed that all the version spaces $\mathcal{H}(h^*, S)$ (such that $|S| \leq d_0$) derived from $\mathcal{H}$ contain a hypothesis that generalizes well on the whole dataset. Moreover, Definition 2 is the literal formalization of the notion of benign overfitting considered in [22] and [23]: models that can fit any labeling –even random– of the train set while still having a good test performance when evaluated on the target distribution.

This definition of large capacity models enables the same analysis as in Theorem 1, without the requirement that the hypothesis class $\mathcal{H}$ spans the entire set of functions $\{0,1\}^{\mathcal{X}}$.

Corollary 1 (Benign overfitting and μ-diameter)**.** *Let $\mathcal{X}$ and $\mathcal{H} \subseteq \{0,1\}^{\mathcal{X}}$ be any input space and hypothesis class. Assume that $\mathcal{H}$ exhibits benign overfitting with respect to the sensitive attribute X_A and its opposite $1 - X_A$* [3]*, then $\forall d \leq d_0, S \in \mathcal{X}^d$,*

$$\begin{aligned} \operatorname{diam}_\mu \mathcal{H}(h^*, S) \geq\, & \mathbb{P}(X \in S | X_A = 1) + \mathbb{P}(X \in S | X_A = 0) \\ & - 2\,\mathbb{P}(X \in S) - 2\epsilon\,(1 - \mathbb{P}(X \in S)) \end{aligned}$$

Observe that lower bound on the μ-diameter given by Corollary 1 only depends on the aggregated quantities $\mathbb{P}\left(X \in \overline{S}\right)$, $\mathbb{P}\left(X \in \overline{S} \middle| X_A = 1\right)$ and $\mathbb{P}\left(X \in \overline{S} \middle| X_A = 1\right)$. As for Theorem 1, this implies that no audit method, active or not can perform better than a simple random sampling baseline (Algorithm 1) with the right proportions β_1 and β_2. The term $\mathbb{P}(X \in S | X_A = 1) + \mathbb{P}(X \in S | X_A = 0) - 2\,\mathbb{P}(X \in S)$ indicates the importance of the relative proportion of these two audited groups in the audit set as in Theorem 1. The term $2\epsilon\,(1 - \mathbb{P}(X \in S))$ indicates that as expected, the larger the error rate ϵ gets, the smaller the μ-diameter will be. Thus, **when the hypothesis class exhibits benign overfitting, all audit algorithms –active or not– have at most the same manipulation-proofness guarantees as random sampling.** This shows that large models currently used in production are not auditable more efficiently than by random sampling.

IV. Audit difficulty and model capacity

As shown in section III, the random audit baseline is optimal when the model has a large capacity, but has no guarantee of optimality when the hypothesis class is constrained to lower

[3] That is, Definition 2 holds for $c = x_A$ and $c = 1 - x_A$

capacities. To compare ML algorithms in practice, we now introduce a measure of *audit difficulty* and a measure of *model capacity*. We will use these methods to empirically evaluate the difficulty of auditing several models of increasing capacities in section V.

A. Measuring the audit difficulty of practical models

The difficulty of auditing a hypothesis class $\mathcal{H}$, is defined (Equation 6) as the μ-diameter obtained and averaged over audit datasets S sampled by the random audit baseline Algorithm 1 with budget s.

$$\text{Difficulty}(\mathcal{H}, s) = \mathbb{E}_{S,h^*}\left[\text{diam}_\mu \mathcal{H}(S, h^*)\right] \tag{6}$$

a) The audit difficulty is a lower bound of the auditor "power": In a perfect situation, for any budget $s = |S|$, the auditor would be able to select the audit set S^* that attains the minimum μ-diameter, whatever the hypothesis class $\mathcal{H}$ and chosen hypothesis $h^* \in \mathcal{H}$ are. As explained in subsection II-E, this is not possible in practice for computational reasons and thus cannot be simulated. Thus we evaluate the audit difficulty with the baseline random audit strategy (Algorithm 1). Taking the expectation of $\text{diam}_\mu \mathcal{H}(h^*, S)$ over random audits allows to upper bound the value of the minimum attainable μ-diameter $\min_\mathcal{A} \text{diam}_\mu \mathcal{H}(h^*, \mathcal{A}(h^*))$.

b) The audit difficulty is a lower bound of the platform "power": In a fully adversarial setting, whatever the hypothesis class $\mathcal{H}$, the platform would choose the hypothesis h^* that maximizes $\text{diam}_\mu \mathcal{H}(h^*, S)$ for most of the audit sets S the auditor could come up with. While this would effectively be the worst case for the auditor, it is however unlikely to happen in practice since the platform would have to balance the maximization of the accuracy with the maximization of the μ-diameter. Therefore, we consider the more practical situation in which the platform can freely choose the hypothesis class $\mathcal{H}$ but the implemented instance h^* minimizes a classical loss L adapted to the model being trained (e.g. cross-entropy or ℓ_2 norm). This can be seen as a lower bound of the adversarial "power" of the platform.

B. Measuring the capacity of practical models

There are multiple operationalizations of the notion of capacity, from theoretically-rooted metrics such as the VC dimension [26] or Rademacher complexity [27], to more empirical definition such as the number of iterations until overfitting [22]. The interplay between VC-dimension and audit difficulty is already pointed out in [1], where it is observed that models of VC-dimension higher than $1,600$ have a high audit difficulty.

Unfortunately, the VC dimension of a class is difficult to estimate in practical settings. Instead, the empirical Rademacher complexity (Equation 7) is leveraged to quantify the capacity of the studied hypothesis classes. Informally in our setting, a hypothesis class has a high Rademacher complexity if whatever the labels and size of an audit set S, there exists an instance $h_S \in \mathcal{H}$ that fits those labels on S with high accuracy. To avoid threshold effects in our experiments, we average the complexity over different sizes of D considered in the Rademacher metric (Equation 8). Formally:

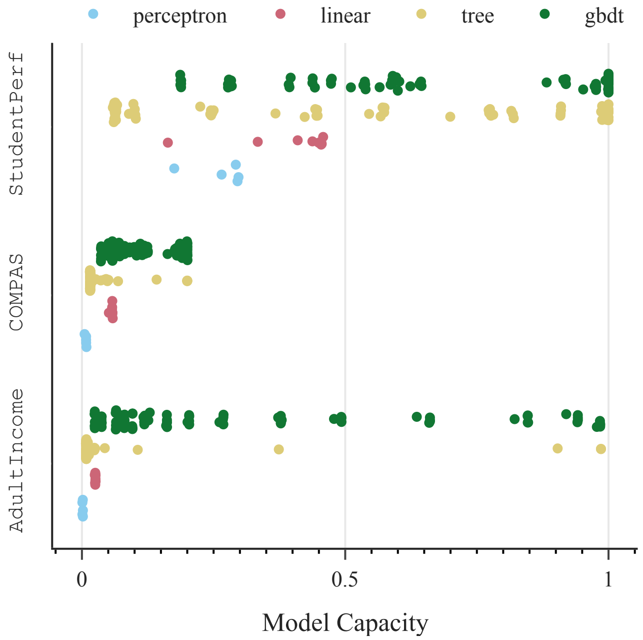

Figure 3. Distribution of the capacity (horizontal axis) for different hyperparameters choices on the three datasets (vertical axis). Each model is trained with different hyperparameter values with each couple (model, hyperparameter) representing a different hypothesis class $\mathcal{H}$. For each (model, hyperparameter) couple, the empirical Rademacher values $R_m(\mathcal{H} \circ D)$ are averaged over 15 realizations of D and σ_i before computing the model capacity.

$$R_m(\mathcal{H} \circ D) = \frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}\sim\{\pm1\}^m}\left[\sup_{h\in\mathcal{H}} \sum_{x_i \in D} \sigma_i h(x_i)\right] \tag{7}$$

$$\text{Capacity}(\mathcal{H}) = \mathbb{E}_{\substack{D\sim\mathcal{X}^m \\ m\sim[\![|\mathcal{X}|]\!]}}\left[R_m(\mathcal{H} \circ D)\right] \tag{8}$$

V. Experiments

In this section, we explore the relation of the audit difficulty (Equation 6) with the capacity of hypothesis classes (Equation 8). The following experiments were run on three tabular datasets: `StudentPerf` [15], `COMPAS` [3] and `AdultIncome` [28]. Dataset statistics and considered tasks are presented in Table II. Neural methods on tabular data are still outperformed by tree methods [29]. We thus choose to focus our study on the four following models: linear models, perceptrons, decision trees and gradient-boosted trees. Similar to [29], we selected a range of hyperparameters for each model and sampled a total of 500 hyperparameters over the 4 models. In previous sections, we stated results with respect to a given hypothesis class $\mathcal{H}$. In the following experiments, a hypothesis class $\mathcal{H}$ represents a couple (model, hyperparameters). Thus, a model represents a family of hypothesis classes $\mathcal{F} = (\mathcal{H}_1, \ldots, \mathcal{H}_f)$, each hypothesis class $\mathcal{H}_i$ being associated with a hyperparameters tuple.

The hyperparameters and their value range are presented in Table III. For each model, we created a grid with all

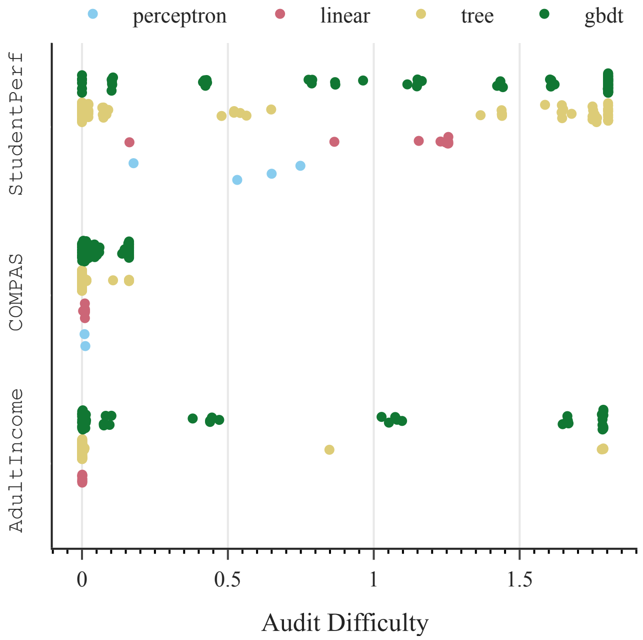

Figure 4. Distribution of the audit Difficulty values (horizontal axis) of different models $\mathcal{F}$ on a selection of datasets (vertical axis). Each bar represents a different model $\mathcal{F}$ (trees, linear models, ...). Each model is trained with different hyperparameter values with each couple (model, hyperparameter) representing a different hypothesis class $\mathcal{H}$. For each dataset, the size of the audit set is set to 10% of the dataset size: $|S| = 0.1\,|\mathcal{X}|$. For each (model, hyperparameter) couple, the μ-diameter are averaged over 15 audit datasets before computing the audit difficulty.

Table II
DATASETS STATS

dataset	Size n	Features d	Task
StudentPerf	395	43	Predict if students pass the exam
COMPAS	6172	21	Predict subject recidivism
AdultIncome	$22,268$	10	Predict if income is $\geq$ \$50,000

the possible combinations of hyperparameter values and ran our experiments on all the resulting (model, hyperparameter) couples. The code needed to run the experiment, the hyperparameters, the data we obtained and the code to reproduce the figures will be made available upon publication.

A. Simulating hypothesis spaces with a broad range of audit difficulty and capacity

In Figure 4, we plot the audit difficulty of different hypothesis classes. These classes are constructed by using multiple hyperparameters for each family $\mathcal{F}$ listed in Table III; each dot then represents a specific (family, hyperparameter set) couple. On one hand, for large datasets (such as AdultIncomeand COMPAS), we observe that simpler models (linear, perceptron) have a very low audit difficulty, no matter the hyperparameter set used. On the other hand, for smaller datasets (such as StudentPerf), smaller models (such as linear models or perceptrons) can also fit the data hence also becoming harder to audit.

Table III
VALUE RANGE FOR THE HYPERPARAMETERS OF THE MODELS USED IN THE EXPERIMENTS.

Model & hyperparameters	Value range
LINEAR	
penalty	(None, l2)
C	(0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000)
PERCEPTRON	
penalty	(l2,)
alpha	(1e-06, 1e-05, 0.0001, 0.001, 0.01)
TREE	
max_depth	(2, 4, 8, 16, 32, 64, 128)
ccp_alpha	(0.001, 0.003, 0.005, 0.007, 0.01, 0.05, 0.1, 0.2, 0.5, 0.0)
GBDT	
max_depth	(1, 2, 4, 8)
n_estimators	(100, 200, 500)
reg_lambda	(0.0, 1e-6, 1e-3, 0.1, 1.0, 1e6, 1e7)
max_leaves	(0,)
learning_rate	(0.3,)
gamma	(0.0,)
min_child_weight	(0.0,)
max_delta_step	(0.0,)
subsample	(1.0,)
reg_alpha	(0.0,)
early_stopping_rounds	(None,)

Similarly, in Figure 3, we plot the capacity of the simulated hypothesis classes on AdultIncome, COMPAS and StudentPerf. As discussed before, it can be observed that for AdultIncome and StudentPerf datasets, tree-based models reach the maximum capacity value of 1. However, on the COMPAS dataset all hypothesis classes exhibit capacity values that do not exceed 0.2 points. This has been observed before [30] and does not affect our main argument on the link between model capacity and audit difficulty.

B. Model capacity conditions the audit difficulty

In subsection IV-A we compared different models and how difficult they were to audit, depending on the chosen hyperparameters. We now take a closer look at the impact of a model's capacity on its audit difficulty, in an attempt to confirm the link between both concepts. We plot in Figure 5 the relation between the capacity of a hypothesis class and its audit difficulty. Points also represent (model, hyperparameter) couples, while the vertical error bars represent the standard deviation of the μ-diameter values for different random audit sets S.

Consistent with the intuition and results developed until now, we observe that for all the datasets, the audit difficulty increases with the capacity of the hypothesis class. While on both AdultIncome and StudentPerf, the μ-diameter reaches the maximum capacity value at almost 2, for COMPAS, the effect is not as dramatic. To highlight the connection between the results exposed in section III and the empirical

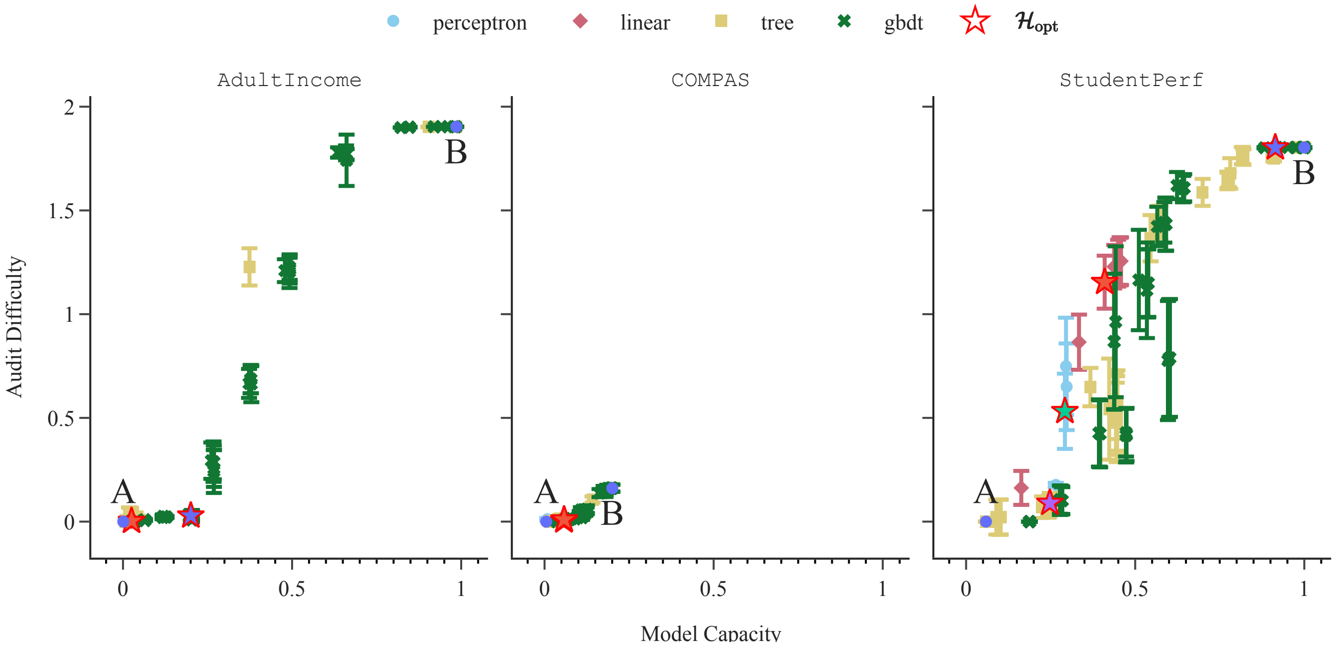

Figure 5. Distribution of the audit difficulty values (vertical axis) of different models versus their capacity (horizontal axis) on a selection of datasets. Each point represents a couple (model, hyperparameter). For each dataset, the size of the audit set is set to 10% of the dataset size: $|S| = 0.1\,|\mathcal{X}|$. For each (model, hyperparameter) couple, the Difficulty is averaged over 15 audit datasets, and the capacity is computed over 30 randomizations of the dataset labels. The error bars represent the standard deviation.

relation found between model capacity and audit difficulty, we focus next on two specific points, marked with the letters A and B in Figure 5.

First, consider the point $A = (\text{Capacity} \simeq 0, \text{Difficulty} \simeq 0)$. For a hypothesis class to have a null capacity, it has to have null Rademacher complexity on any subset of the sample space. This is verified by models that perform no better than random labels generation. Since the value of $\mu(h, \mathcal{X})$ of any instance of such hypothesis class is only determined by the ratio of samples with a positive sensitive attribute, the μ-diameter of such hypothesis class is null. This is why in Figure 5, models with near-zero capacity have a very low (if not null) audit difficulty.

The second notable point is $B = (\text{Capacity} = \text{Capacity}_{\text{max}}, \text{Difficulty} = \text{Difficulty}_{\text{max}})$. Any hypothesis with a unitary capacity has a unitary Rademacher complexity for any dataset size s and thus shatters any subset of $\mathcal{X}$. Therefore, at point B, Theorem 1's hypothesis $\mathcal{H} = \{0, 1\}^{\mathcal{X}}$ holds. This means that hypothesis classes that are characterized by this point cannot be audited more efficiently than by a random audit strategy. It follows that (at least on StudentPerf and AdultIncome) the platform can always choose a hypothesis class that cannot be audited efficiently by any strategy, forcing the auditor to prompt most of the input space to obtain robustness guarantees.

Generalization versus diameter: We saw that by choosing the right hypothesis class (that is, the right set of hyperparameters), the platform can easily evade the audit. However, in practice the choice of hypothesis class is also guided by a classical train-dev-test separation, choosing the hyperparameter set that generalizes best. What is the typical μ-diameter of hypotheses classes that generalize well? To answer this question, we simulate a 5-fold hyperparameter optimization procedure. For each family of models, we denote $\mathcal{H}_{\text{opt}}$ the hypothesis class with the set of hyperparameters that minimize the 5-fold average test loss in its model family $\mathcal{F}$. For each model family, $\mathcal{H}_{\text{opt}}$ is differentiated in Figure 5 by a star marker with red edges. Interestingly, for COMPAS and AdultIncome datasets and for all model families, the generalization-optimal hypothesis classes $\mathcal{H}_{\text{opt}}$ have a relatively low capacity compared to the maximum achievable capacity, especially for tree-based models. For the StudentPerf dataset, the results are more nuanced, most likely because the dataset has a limited size, which implies that it is simpler to reach high capacity values.

As a glimmer of hope, from point A to B, there is a range of hypothesis classes for which the random strategy could be improved as seen by the size of the y-axis error bars. Overall, the hypothesis classes that are most likely to be implemented by faithful platforms (the hypothesis classes that generalize well) are already straightforward to audit (they have a Difficulty ≈ 0). Yet, unfaithful platforms wanting to game the audit can always choose a hypothesis class that forces the auditor to issue a lot of queries to reach higher manipulation-proofness guarantees.

C. The cost of exhausting the auditor

We observed in section III and subsection IV-A that the hypothesis classes that are the hardest to audit are those with the largest capacity. Yet, we also observed that the

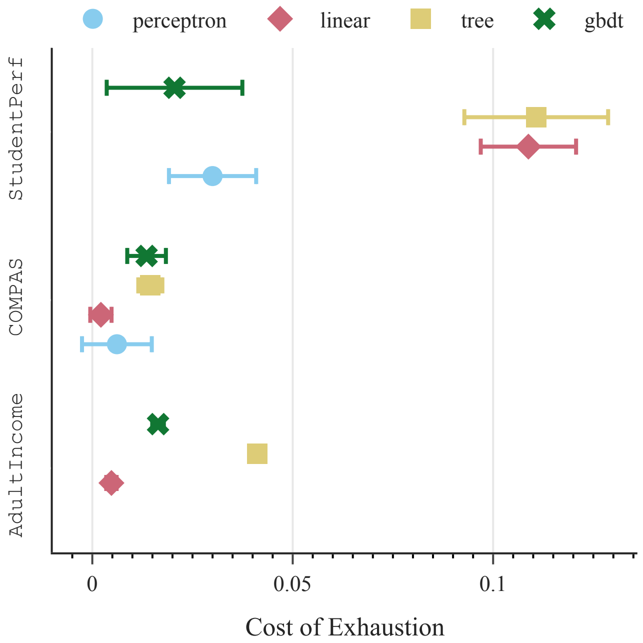

Figure 6. Distribution of the *cost of exhaustion* for the four model families (perceptron, linear, tree and GBDT) on the three considered datasets. The error bars show the 95% confidence interval on the values of the difference of Accuracy$_{\text{test}}$ for the best hypotheses in $\mathcal{H}^{\text{acc}}$ and $\mathcal{H}^{\mu}$. For all models, on all datasets (except for trees and linear models on `StudentPerf`), the cost of exhaustion is below 1%. Trees are the models with the highest cost of exhaustion, while for all the other models, the cost of exhaustion remains relatively low (in particular for the large capacities GDBTs), indicating a negligible accuracy cost for audit evasion.

hypothesis classes most likely to be implemented (i.e. the ones with the highest generalization) have a low μ-diameter and are not those with high capacity. In the manipulation-proof framework of [1] we operate in, the platform chooses the hypothesis class without constraints before disclosing it to the auditor. Therefore, when choosing a specific model family $\mathcal{F}$, a malicious platform would have the possibility to trade performance (i.e. generalization capability) with the ability to attempt audit evasion. To understand the trade-offs involved in balancing these two objectives, we introduce the notion of CostOfExhaustion$(\mathcal{F})$ of a model family $\mathcal{F}$.

Consider a model family $\mathcal{F}$ (e.g. a linear model or tree). We define the hypothesis class in $\mathcal{F}$ with the best trained test accuracy:

$$\mathcal{H}^{\text{acc}} = \arg\max_{\mathcal{H}\in\mathcal{F}} \max_{h\in\mathcal{H}} \text{Accuracy}_{\text{test}}(h, \mathcal{X}) \tag{9}$$

Assuming that an honest platform chooses its hypothesis class based on generalization capabilities, $\mathcal{H}^{\text{acc}}$ is the hypothesis class an honest platform would actually choose. Then, define the hypothesis class in $\mathcal{F}$ with the largest difficulty of audit:

$$\mathcal{H}^{\mu} = \arg\max_{\mathcal{H}\in\mathcal{F}} \mathbb{E}_{S\sim\mathcal{X}^s}\left[\text{diam}_{\mu}(\arg\max_{h\in\mathcal{H}} \text{Accuracy}_{\text{test}}(h), S)\right] \tag{10}$$

Should a platform try to escape audits at a low cost, they would try to find a hypothesis class whose optimal hypothesis h^* leads to a high μ-diameter. Thus, the cost of exhaustion is the accuracy cost of using the hypothesis class $\mathcal{H}^{\mu}$ compared to using $\mathcal{H}^{\text{acc}}$:

$$\text{CostOfExhaustion}(\mathcal{F}) = \max_{h\in\mathcal{H}^{\text{acc}}} \text{Accuracy}_{\text{test}}(h, \mathcal{X}) - \max_{h\in\mathcal{H}^{\mu}} \text{Accuracy}_{\text{test}}(h, \mathcal{X}) \tag{11}$$

The cost of exhaustion is plotted in Figure 6, for the four model families already considered, on the three datasets. The error bars show the 95% confidence interval on the values of the difference of Accuracy$_{\text{test}}$ for the best hypotheses in $\mathcal{H}^{\text{acc}}$ and $\mathcal{H}^{\mu}$. For all models, on all considered datasets (except for trees and linear models on the dataset `StudentPerf`), the cost of exhaustion is below 1%. Trees are the models with the highest cost of exhaustion. In fact, as we observed in Figure 5, given enough capacity, trees can reach the maximum audit difficulty. Yet, it is known that without regularization, complex trees can easily overfit the training data, thus lowering the max test accuracy of the $\mathcal{H}^{\mu}$ class compared to the max test accuracy of $\mathcal{H}^{\text{acc}}$. On the other hand, the models with the lowest cost of exhaustion (except on `StudentPerf`) are linear models. As observed on Figure 3, for all datasets, linear models span a small portion of the capacity range (around .1 points for `StudentPerf` and less than .01 points for `COMPAS` and `AdultIncome`), compared to larger models (e.g. GBDTs) which cover almost the entire capacity range on `StudentPerf` and `AdultIncome`. This result is challenging for the existence of efficient audits in the manipulation-proof framework. In fact, the witnessed low cost of exhaustion for larger capacity models indicates that platforms may evade audits at the cost of a minor loss in accuracy.

D. Effects of the audit set size

In this section, we experiment with different sizes of audit dataset and show that our conclusions do not change with the change in dataset size (we had $|S| = .1\,|\mathcal{X}|$ in previous experiments). To do so, we select three different hypotheses classes for each model family. We choose the hypothesis class that generalizes best $\mathcal{H}_{\text{opt}}$, the hypothesis class with the lowest capacity $\mathcal{H}_{-}$ and with the highest capacity $\mathcal{H}_{+}$. In Figure 7 we show the audit difficulty of each hypothesis class against the size of the audit dataset $|S|$. The results indicate that there is no significant inversion of the audit difficulty between the various hypotheses in the range of interest. Results in Figure 7 are shown only for the `AdultIncome` dataset. The results for the other datasets are showed in the Appendix, in Figures 8 and 9, which to the same conclusion.

VI. Related work

The problem of manipulation-proof auditing and more generally black-box, remote, and robust property verification of ML platforms arises from the need to enforce regulations. As an example, consider the European Union. Classical fairness regulation of online ML models mainly comes from the *Racial Equality Directive* [31], the *Framework Equality Directive* [32] and the *Gender Equality Directives* [33], [34]. Recently, the

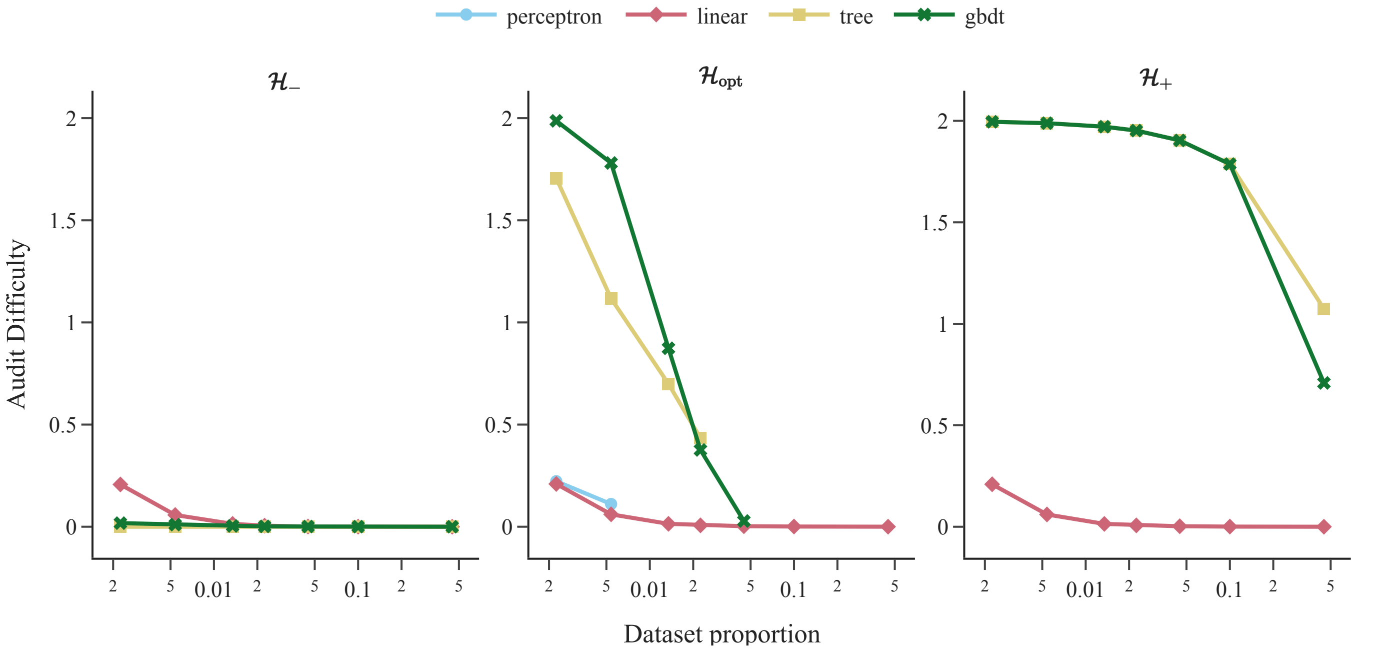

Figure 7. Evolution of the μ-diameter with the size of the audit set S represented as a proportion of the total dataset size for the `AdultIncome` dataset. Each line represents an audited model, whose hyperparameters are either tuned for the best generalization, either tuned for the highest capacity or tuned for the lowest capacity. For each (model, hyperparameter) couple, the μ-diameter is averaged over 15 audit datasets.

EU set out to create regulations specific to online platforms. These are the *AI Act* [35], the *Digital Services Act* [36] and the *Digital Markets Act* [37]. These directives provide a legal framework that prescribes what online platforms may and may not do, but offer little to verify that these rules are respected in practice. The manipulation-proof framework is a first attempt to provide operational solutions that can detect when platforms do not follow the law.

In addition, our results are mostly related to the following lines of work.

a) Algorithm auditing: The field of algorithm auditing is interested in understanding the impact of algorithms on the lives or the people impacted by those algorithms' decisions. In practice, auditing algorithms *in vivo* (that is as they are deployed in online services) is challenging because they constantly evolve, mostly without records [38]. For a survey on examples of published academic audits of decision systems, refer to [20]. Moreover, because it is impossible for researchers or regulators to audit each automated decision system, it has been observed that most of the recent discoveries of problematic algorithm behavior have surfaced thanks to users of those systems [39], [40]. Again, after a problematic algorithm behavior has been detected and after a court decision has been made, we still need to be able to monitor that this decision is respected.

b) Audit metrics and audit design: With the advent of broadly publicized algorithm audits such as COMPAS [3] or Reuters' study on Amazon's recruiting tool [4], there has been an effort to devise metrics and their interpretations to better understand the impact of algorithms on their users. Most of the effort has been directed towards the operationalization of fairness values into the ML framework [17]. Classical fairness measures include Demographic Parity [41], Equalized Odds [42], Equal Opportunity [42] or Predictive Parity [43]. All of these measures encompass different visions of fairness and choosing one versus the other has political implications on the considered notion of fairness [44], [45]. While still marginal, some works are interested in other aspects of the audit of AI algorithms. For example, [46] is interested in the verification that online platforms comply with the Data Minimization Principle. Another interesting work [47] considers the problem of automatically auditing the privacy guarantees offered by AI algorithms. However, most of the presented works do not yet consider the possibility of the platform gaming their audit.

c) Robust verification: The literature on robust auditing is still in its infancy. The manipulation-proof [1] framework has only recently been introduced. However, with its goal of efficiently choosing the next audit query based on previous queries and the associated outputs of the API, the manipulation-proof framework exhibits clear links with the active learning literature [48], [49]. With the aim of finding methods to ease the audit, [2] showed that the explanation provided by the platform can greatly improve the robustness of audits. For example, they show that for linear classifiers, a single result along its counterfactual explanation allows to totally characterize the model. Our work does not assume that the auditor has access to explanations. It is likely that faithful explanation could lead to audit algorithms with increased MP guarantees. On another line of works, [50] and [51] suggest instantiating an audit protocol in which both the platform and the auditor would be active, drawing inspiration from zero-knowledge proofs and interactive verification protocols.

d) Benign overfitting and model capacity: As we proved in this work, audit difficulty has deep connections with model capacity and their ability to perfectly fit arbitrary datasets. Classical metrics that capture the notion of model capacity include the VC-dimension [52] or the Rademacher Complexity (which we used for its usability in practice) [27]. Moreover, our experiments on the link between auditability and model capacity have been motivated by the recent finding that larger models can fit the training dataset perfectly while still showing good generalization properties [22]. This effect has been observed for linear models [53], Support Vector Machines [54] and Decision Trees [55]. In the manipulation-proof audit setting, we show that this type of behavior is very problematic. In fact, if a model is able to fit any audit set and yet keep its generalization performance, platforms do not even have to lie to the auditor. They just have to train their model to give the answers the auditor expects on their audit set. Then, the platform can define any objective for the rest of the input space, even if it does not align with the auditor's metric.

Interestingly, the connection between model capacity and audit query complexity is not limited to manipulation-proof estimation of parity measures. In their work on certified feature sensitivity auditing [2], Yadav, Moshkovitz, and Chaudhuri provide an algorithm to audit feature sensitivity for decision trees whose query complexity grows linearly with the capacity (number of nodes) of the tree.

VII. Conclusion and discussions

The introduction of the *manipulation-proofness* framework [1] has certainly been an important step for auditors to start understanding that algorithmic audits can suffer from platform manipulations and what cost that brings along.

In this work, we conducted a thorough exploration of the concept of manipulation-proofness. We derived theoretical conditions on the hypothesis class implemented by the platform for the impossibility of efficient manipulation-proof audits. We carried out a thorough experimental validation on the *audit difficulty* of state-of-the-art models for tabular data. Our results draw a connection between the capacity of the audited model and the difficulty of the audit task.

We now discuss some countermeasures to improve the audit robustness. A promising line of work is to require platforms to provide certificates. Since the goal of certificates is to provide a cheap verification procedure (at the cost of a potentially high certificate generation cost), this would shift the computational burden to the platform. One example of a fairness certificate was provided in [7]. Such extended assumptions (over mere black box audits) are certainly an interesting research line for future works.

In the end, when implementing large-capacity models, a platform can always game the audit without sacrificing too much accuracy. We believe that this demonstrates the limitations of black-box auditing for regulation, even when the hypothesis class used by the platform is known to the regulator. We claim that regulators should be given more than black-box access to AI models as part of the audit procedure or that they should explore certification-based audits such as [7]. Therefore, we urge the community to participate in the search for audit frameworks that are both exploitable in practice and also supported by theoretical guarantees.

References

[1] T. Yan and C. Zhang, “Active fairness auditing,” in *Proceedings of the 39th International Conference on Machine Learning*, PMLR, Jun. 28, 2022, pp. 24 929–24 962. [Online]. Available: https://proceedings.mlr.press/v162/yan22c.html (visited on 12/01/2022).

[2] C. Yadav, M. Moshkovitz, and K. Chaudhuri. “XAudit : A Theoretical Look at Auditing with Explanations.” version 3. arXiv: 2206.04740 [cs]. (Jun. 5, 2023), [Online]. Available: http://arxiv.org/abs/2206.04740 (visited on 09/18/2023), preprint.

[3] J. Larson, S. Mattu, L. Kirchner, and J. Angwin, “How We Analyzed the COMPAS Recidivism Algorithm,” *ProPublica*, May 23, 2016. [Online]. Available: https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm (visited on 03/06/2023).

[4] J. Dastin, “Amazon scraps secret AI recruiting tool that showed bias against women,” *ReutersRetail*, Oct. 10, 2018. [Online]. Available: https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G (visited on 03/06/2023).

[5] M. T. Ribeiro, S. Singh, and C. Guestrin, “”Why Should I Trust You?”: Explaining the Predictions of Any Classifier,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD ’16, New York, NY, USA: Association for Computing Machinery, Aug. 13, 2016, pp. 1135–1144, ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2939778. [Online]. Available: https://doi.org/10.1145/2939672.2939778 (visited on 01/20/2023).

[6] S. M. Lundberg and S.-I. Lee, “A Unified Approach to Interpreting Model Predictions,” in *Advances in Neural Information Processing Systems*, vol. 30, Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html (visited on 03/08/2023).

[7] A. S. Shamsabadi, S. C. Wyllie, N. Franzese, *et al.*, “Confidential-PROFITT: Confidential PROof of FaIr Training of Trees,” presented at the The Eleventh International Conference on Learning Representations, Feb. 1, 2023. [Online]. Available: https://openreview.net/forum?id=iIfDQVyuFD (visited on 03/08/2023).

[8] J. N. Matias, A. Hounsel, and N. Feamster, “Software-Supported Audits of Decision-Making Systems: Testing Google and Facebook’s Political Advertising Policies,” *Proceedings of the ACM on Human-Computer Interaction*, vol. 6, 118:1–118:19, CSCW1 Apr. 7, 2022. DOI: 10.1145/3512965. [Online]. Available: https://doi.org/10.1145/3512965 (visited on 03/08/2023).

[9] B. Rastegarpanah, K. Gummadi, and M. Crovella, “Auditing Black-Box Prediction Models for Data Minimization Compliance,” in *Advances in Neural Information Processing Systems*, vol. 34, Curran Associates, Inc., 2021, pp. 20 621–20 632. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/ac6b3cce8c74b2e23688c3e45532e2a7-Abstract.html (visited on 11/02/2022).

[10] O. Goldreich, *Foundations of Cryptography: Volume 2, Basic Applications*. Cambridge university press, 2009.

[11] D. Metaxa, J. S. Park, R. E. Robertson, *et al.*, “Auditing Algorithms: Understanding Algorithmic Systems from the Outside In,” *Foundations and Trends in HumanComputer Interaction*, vol. 14, no. 4, pp. 272–344, 2021, ISSN: 1551-3955, 1551-3963. DOI: 10.1561/1100000083. [Online]. Available: http://www.nowpublishers.com/article/Details/HCI-083 (visited on 11/23/2022).

[12] C. Sandvig, K. Hamilton, K. Karahalios, and C. Langbort, “Auditing algorithms: Research methods for detecting discrimination on internet platforms,” *Data and discrimination: converting critical concerns into productive inquiry*, vol. 22, no. 2014, pp. 4349–4357, 2014. [Online]. Available: https://www.kevinhamilton.org/share/papers/Auditing%20Algorithms%20–%20Sandvig%20–%20ICA%202014%20Data%20and%20Discrimination%20Preconference.pdf (visited on 10/11/2023).

[13] B. Chugg, S. Cortes-Gomez, B. Wilder, and A. Ramdas, “Auditing Fairness by Betting,” presented at the Thirty-Seventh Conference on Neural Information Processing Systems, Nov. 2, 2023. [Online]. Available: https://openreview.net/forum?id=EEVpt3dJQj (visited on 01/17/2024).

[14] S. Hanneke, “Theory of Disagreement-Based Active Learning,” *Foundations and Trends in Machine Learning*, vol. 7, no. 2-3, pp. 131–309, Jun. 11, 2014, ISSN: 1935-8237, 1935-8245. DOI: 10.1561/2200000037. [Online]. Available: https://www.nowpublishers.com/article/Details/MAL-037 (visited on 01/12/2023).

[15] P. Cortez and A. Silva, “Using data mining to predict secondary school student performance,” *EUROSIS*, Jan. 1, 2008.

[16] R. Richardson, “Defining and Demystifying Automated Decision Systems,” *Maryland Law Review*, vol. 81, p. 785, 2021–2022. [Online]. Available: https://heinonline.org/HOL/Page?handle=hein.journals/mllr81&id=805&div=&collection=.

[17] S. Barocas, M. Hardt, and A. Narayanan, *Fairness and Machine Learning: Limitations and Opportunities*. MIT Press, 2023. [Online]. Available: https://books.google.com/books?hl=en&lr=&id=ouawEAAAQBAJ&oi=fnd&pg=PA83&dq=Fairness+and+Machine+Learning&ots=2kDVfTXV8P&sig=uYNciZV1R1c7X_LakbSmwN_YJYk (visited on 01/04/2024).

[18] F. Lu, J. Munoz, M. Fuchs, *et al.*, “A General Framework for Auditing Differentially Private Machine Learning,” presented at the Advances in Neural Information Processing Systems, vol. 35, Dec. 6, 2022, pp. 4165–4176. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2022/hash/1add3bbdbc20c403a383482a665eb5a4-Abstract-Conference.html (visited on 08/16/2023).

[19] S. Dasgupta, D. Hsu, S. Poulis, and X. Zhu, “Teaching a black-box learner,” in *Proceedings of the 36th International Conference on Machine Learning*, PMLR, May 24, 2019, pp. 1547–1555. [Online]. Available: https://proceedings.mlr.press/v97/dasgupta19a.html (visited on 12/12/2022).

[20] J. Bandy, “Problematic Machine Behavior: A Systematic Literature Review of Algorithm Audits,” *Proceedings of the ACM on Human-Computer Interaction*, vol. 5, 74:1–74:34, CSCW1 Apr. 22, 2021. DOI: 10.1145/3449148. [Online]. Available: https://dl.acm.org/doi/10.1145/3449148 (visited on 10/12/2023).

[21] T. M. Mitchell, “Generalization as search,” *Artificial Intelligence*, vol. 18, no. 2, pp. 203–226, Mar. 1, 1982, ISSN: 0004-3702. DOI: 10.1016/0004-3702(82)90040-6. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0004370282900406 (visited on 06/06/2023).

[22] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, “Understanding deep learning (still) requires rethinking generalization,” *Communications of the ACM*, vol. 64, no. 3, pp. 107–115, Feb. 22, 2021, ISSN: 0001-0782. DOI: 10.1145/3446776. [Online]. Available: https://dl.acm.org/doi/10.1145/3446776 (visited on 04/11/2023).

[23] M. Belkin, D. Hsu, S. Ma, and S. Mandal, “Reconciling modern machine-learning practice and the classical biasvariance trade-off,” *Proceedings of the National Academy of Sciences*, vol. 116, no. 32, pp. 15 849–15 854, Aug. 6, 2019. DOI: 10.1073/pnas.1903070116. [Online]. Available: https://www.pnas.org/doi/10.1073/pnas.1903070116 (visited on 10/03/2023).

[24] L. Arnould, C. Boyer, and E. Scornet, “Is interpolation benign for random forest regression?” In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, PMLR, Apr. 11, 2023, pp. 5493–5548. [Online]. Available: https://proceedings.mlr.press/v206/arnould23a.html (visited on 10/03/2023).

[25] S. Buschjger and K. Morik. “There is no Double-Descent in Random Forests.” arXiv: 2111.04409 [cs,

stat]. (Nov. 8, 2021), [Online]. Available: http://arxiv.org/abs/2111.04409 (visited on 10/03/2023), preprint.

[26] V. N. Vapnik and A. Ya. Chervonenkis, “On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities,” *Theory of Probability & Its Applications*, vol. 16, no. 2, pp. 264–280, Jan. 1971, ISSN: 0040-585X. DOI: 10.1137/1116025. [Online]. Available: https://epubs.siam.org/doi/10.1137/1116025 (visited on 07/26/2023).

[27] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*, 1st ed. Cambridge University Press, May 19, 2014, ISBN: 978-1-107-05713-5 978-1-107-29801-9. DOI: 10.1017/CBO9781107298019. [Online]. Available: https://www.cambridge.org/core/product/identifier/9781107298019/type/book (visited on 03/31/2023).

[28] F. Ding, M. Hardt, J. Miller, and L. Schmidt, “Retiring Adult: New Datasets for Fair Machine Learning,” in *Advances in Neural Information Processing Systems*, vol. 34, Curran Associates, Inc., 2021, pp. 6478–6490. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/32e54441e6382a7fbacbbbaf3c450059-Abstract.html (visited on 03/10/2023).

[29] L. Grinsztajn, E. Oyallon, and G. Varoquaux, “Why do tree-based models still outperform deep learning on typical tabular data?” In *Advances in Neural Information Processing Systems*, vol. 35, Dec. 6, 2022, pp. 507–520. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2022/hash/0378c7692da36807bdec87ab043cdadc-Abstract-Datasets_and_Benchmarks.html (visited on 05/10/2023).

[30] J. Dressel and H. Farid, “The accuracy, fairness, and limits of predicting recidivism,” *Science Advances*, vol. 4, no. 1, eaao5580, Jan. 17, 2018, ISSN: 2375-2548. DOI: 10.1126/sciadv.aao5580. pmid: 29376122. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5777393/ (visited on 10/11/2023).

[31] *Council Directive 2000/43/EC of 29 June 2000 implementing the principle of equal treatment between persons irrespective of racial or ethnic origin*, Jun. 29, 2000. [Online]. Available: http://data.europa.eu/eli/dir/2000/43/oj/eng (visited on 10/12/2023).

[32] *Council Directive 2000/78/EC of 27 November 2000 establishing a general framework for equal treatment in employment and occupation*, Nov. 27, 2000. [Online]. Available: http://data.europa.eu/eli/dir/2000/78/oj/eng (visited on 10/12/2023).

[33] *Council Directive 2004/113/EC of 13 December 2004 implementing the principle of equal treatment between men and women in the access to and supply of goods and services*, Dec. 13, 2004. [Online]. Available: http://data.europa.eu/eli/dir/2004/113/oj/eng (visited on 10/12/2023).

[34] *Directive 2006/54/EC of the European Parliament and of the Council of 5 July 2006 on the implementation of the principle of equal opportunities and equal treatment of men and women in matters of employment and occupation (recast)*, Jul. 5, 2006. [Online]. Available: http://data.europa.eu/eli/dir/2006/54/oj/eng (visited on 10/12/2023).

[35] *Proposal for a REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL LAYING DOWN HARMONISED RULES ON ARTIFICIAL INTELLIGENCE (ARTIFICIAL INTELLIGENCE ACT) AND AMENDING CERTAIN UNION LEGISLATIVE ACTS*, 2021. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52021PC0206 (visited on 10/12/2023).

[36] *Regulation (EU) 2022/2065 of the European Parliament and of the Council of 19 October 2022 on a Single Market For Digital Services and amending Directive 2000/31/EC (Digital Services Act) (Text with EEA relevance)*, Oct. 19, 2022. [Online]. Available: http://data.europa.eu/eli/reg/2022/2065/oj/eng (visited on 10/12/2023).

[37] *Regulation (EU) 2022/1925 of the European Parliament and of the Council of 14 September 2022 on contestable and fair markets in the digital sector and amending Directives (EU) 2019/1937 and (EU) 2020/1828 (Digital Markets Act) (Text with EEA relevance)*, Sep. 14, 2022. [Online]. Available: http://data.europa.eu/eli/reg/2022/1925/oj/eng (visited on 10/12/2023).

[38] D. Metaxa, J. S. Park, R. E. Robertson, *et al.*, “Auditing Algorithms: Understanding Algorithmic Systems from the Outside In,” *Foundations and Trends in HumanComputer Interaction*, vol. 14, no. 4, pp. 272–344, Nov. 24, 2021, ISSN: 1551-3955, 1551-3963. DOI: 10.1561/1100000083. [Online]. Available: https://www.nowpublishers.com/article/Details/HCI-083 (visited on 10/12/2023).

[39] A. DeVos, A. Dhabalia, H. Shen, K. Holstein, and M. Eslami, “Toward User-Driven Algorithm Auditing: Investigating users strategies for uncovering harmful algorithmic behavior,” in *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, ser. CHI ’22, New York, NY, USA: Association for Computing Machinery, Apr. 29, 2022, pp. 1–19, ISBN: 978-1-4503-9157-3. DOI: 10.1145/3491102.3517441. [Online]. Available: https://dl.acm.org/doi/10.1145/3491102.3517441 (visited on 10/12/2023).

[40] W. H. Deng, B. Guo, A. Devrio, H. Shen, M. Eslami, and K. Holstein, “Understanding Practices, Challenges, and Opportunities for User-Engaged Algorithm Auditing in Industry Practice,” in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, ser. CHI ’23, New York, NY, USA: Association for Computing Machinery, Apr. 19, 2023, pp. 1–18, ISBN: 978-1-4503-9421-5. DOI: 10.1145/3544548.3581026. [Online]. Available: https://dl.acm.org/doi/10.1145/3544548.3581026 (visited on 10/12/2023).

[41] T. Calders, F. Kamiran, and M. Pechenizkiy, "Building Classifiers with Independency Constraints," in *2009 IEEE International Conference on Data Mining Workshops*, Dec. 2009, pp. 13–18. DOI: 10.1109/ICDMW.2009.83. [Online]. Available: https://ieeexplore.ieee.org/document/5360534 (visited on 10/12/2023).

[42] M. Hardt, E. Price, and N. Srebro, "Equality of opportunity in supervised learning," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16, Red Hook, NY, USA: Curran Associates Inc., Dec. 5, 2016, pp. 3323–3331, ISBN: 978-1-5108-3881-9.

[43] S. Corbett-Davies, E. Pierson, A. Feller, S. Goel, and A. Huq, "Algorithmic Decision Making and the Cost of Fairness," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '17, New York, NY, USA: Association for Computing Machinery, Aug. 4, 2017, pp. 797–806, ISBN: 978-1-4503-4887-4. DOI: 10.1145/3097983.3098095. [Online]. Available: https://dl.acm.org/doi/10.1145/3097983.3098095 (visited on 10/12/2023).

[44] H. Heidari, M. Loi, K. P. Gummadi, and A. Krause, "A Moral Framework for Understanding Fair ML through Economic Models of Equality of Opportunity," in *Proceedings of the Conference on Fairness, Accountability, and Transparency*, ser. FAT* '19, New York, NY, USA: Association for Computing Machinery, Jan. 29, 2019, pp. 181–190, ISBN: 978-1-4503-6125-5. DOI: 10.1145/3287560.3287584. [Online]. Available: https://dl.acm.org/doi/10.1145/3287560.3287584 (visited on 10/12/2023).

[45] Arvind Narayanan, director, *Tutorial: 21 fairness definitions and their politics*, Mar. 1, 2018. [Online]. Available: https://www.youtube.com/watch?v=jIXIuYdnyyk (visited on 10/12/2023).

[46] B. Rastegarpanah, K. Gummadi, and M. Crovella, "Auditing Black-Box Prediction Models for Data Minimization Compliance," in *Advances in Neural Information Processing Systems*, vol. 34, Curran Associates, Inc., 2021, pp. 20 621–20 632. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/ac6b3cce8c74b2e23688c3e45532e2a7-Abstract.html (visited on 10/12/2023).

[47] F. Lu, J. Munoz, M. Fuchs, *et al.*, "A General Framework for Auditing Differentially Private Machine Learning," in *Advances in Neural Information Processing Systems*, vol. 35, Dec. 6, 2022, pp. 4165–4176. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2022/hash/1add3bbdbc20c403a383482a665eb5a4-Abstract-Conference.html (visited on 10/12/2023).

[48] S. Hanneke, "Teaching Dimension and the Complexity of Active Learning," in *Learning Theory*, N. H. Bshouty and C. Gentile, Eds., ser. Lecture Notes in Computer Science, Berlin, Heidelberg: Springer, 2007, pp. 66–81, ISBN: 978-3-540-72927-3. DOI: 10.1007/978-3-540-72927-3_7.

[49] S. Dasgupta, D. Hsu, S. Poulis, and X. Zhu, "Teaching a black-box learner," in *Proceedings of the 36th International Conference on Machine Learning*, PMLR, May 24, 2019, pp. 1547–1555. [Online]. Available: https://proceedings.mlr.press/v97/dasgupta19a.html (visited on 10/12/2023).

[50] A. S. Shamsabadi, S. C. Wyllie, N. Franzese, *et al.*, "Confidential-PROFITT: Confidential PROof of FaIr Training of Trees," presented at the The Eleventh International Conference on Learning Representations, Sep. 29, 2022. [Online]. Available: https://openreview.net/forum?id=iIfDQVyuFD (visited on 10/12/2023).

[51] S. Goldwasser, G. N. Rothblum, J. Shafer, and A. Yehudayoff, "Interactive Proofs for Verifying Machine Learning," in *DROPS-IDN/Opus:13580*, Schloss-Dagstuhl - Leibniz Zentrum fr Informatik, 2021. DOI: 10.4230/LIPIcs.ITCS.2021.41. [Online]. Available: https://drops-beta.dagstuhl.de/entities/document/10.4230/LIPIcs.ITCS.2021.41 (visited on 10/12/2023).

[52] V. N. Vapnik and A. Ya. Chervonenkis, "On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities," *Theory of Probability & Its Applications*, vol. 16, no. 2, pp. 264–280, Jan. 1971, ISSN: 0040-585X. DOI: 10.1137/1116025. [Online]. Available: https://epubs.siam.org/doi/10.1137/1116025 (visited on 10/12/2023).

[53] P. L. Bartlett, P. M. Long, G. Lugosi, and A. Tsigler, "Benign overfitting in linear regression," *Proceedings of the National Academy of Sciences*, vol. 117, no. 48, pp. 30 063–30 070, Dec. 2020. DOI: 10.1073/pnas.1907378117. [Online]. Available: https://www.pnas.org/doi/10.1073/pnas.1907378117 (visited on 10/12/2023).

[54] K. Wang, V. Muthukumar, and C. Thrampoulidis, "Benign Overfitting in Multiclass Classification: All Roads Lead to Interpolation," in *Advances in Neural Information Processing Systems*, vol. 34, Curran Associates, Inc., 2021, pp. 24 164–24 179. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/caaa29eab72b231b0af62fbdff89bfce-Abstract.html (visited on 10/12/2023).

[55] L. Arnould, C. Boyer, and E. Scornet. "Is interpolation benign for random forest regression?" arXiv: 2202.03688 `[math, stat]`. (Feb. 9, 2023), [Online]. Available: http://arxiv.org/abs/2202.03688 (visited on 10/12/2023), preprint.

[56] A. Agarwal, A. Beygelzimer, M. Dudik, J. Langford, and H. Wallach, "A Reductions Approach to Fair Classification," in *Proceedings of the 35th International Conference on Machine Learning*, PMLR, Jul. 3, 2018, pp. 60–69. [Online]. Available: https://proceedings.mlr.press/v80/agarwal18a.html (visited on 01/17/2023).

Appendix

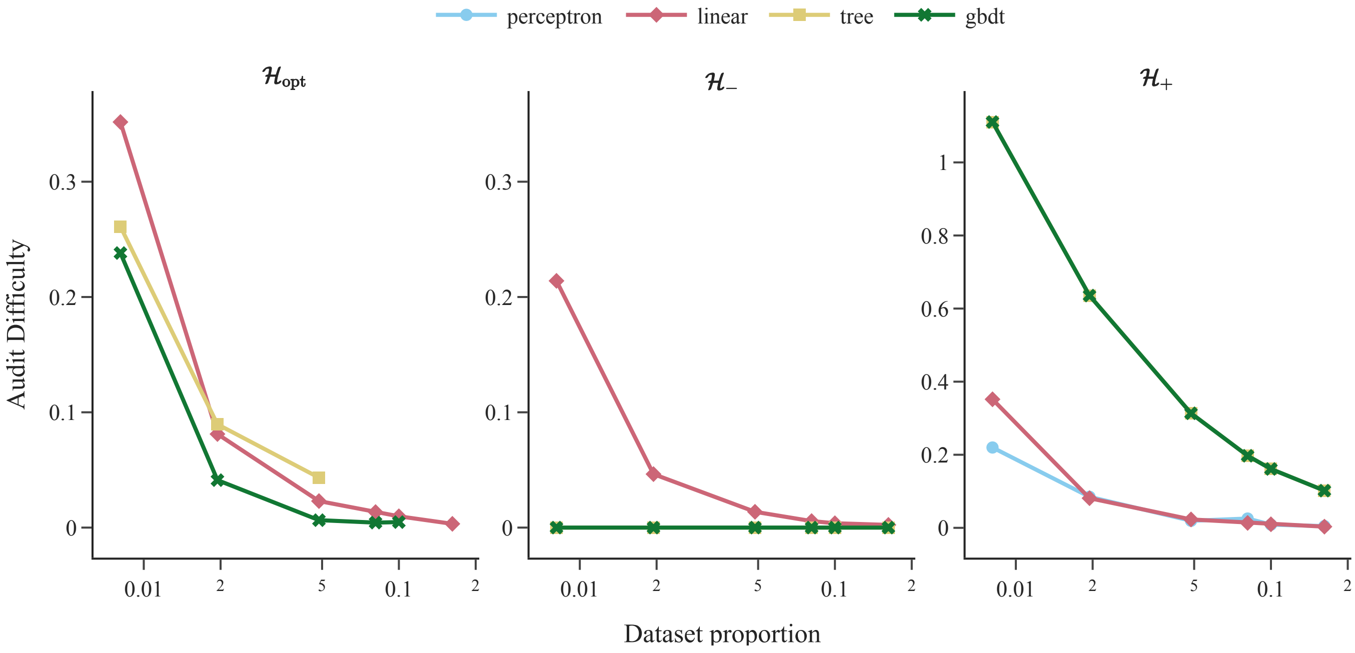

Figure 8. Evolution of the μ-diameter with the size of the audit set S represented as a proportion of the total dataset size for the `AdultIncome` dataset. Each line represents an audited model, whose hyperparameters are either tuned for the best generalization, either tuned for the highest capacity or tuned for the lowest capacity. For each (model, hyperparameter) couple, the μ-diameter is averaged over 15 audit datasets.

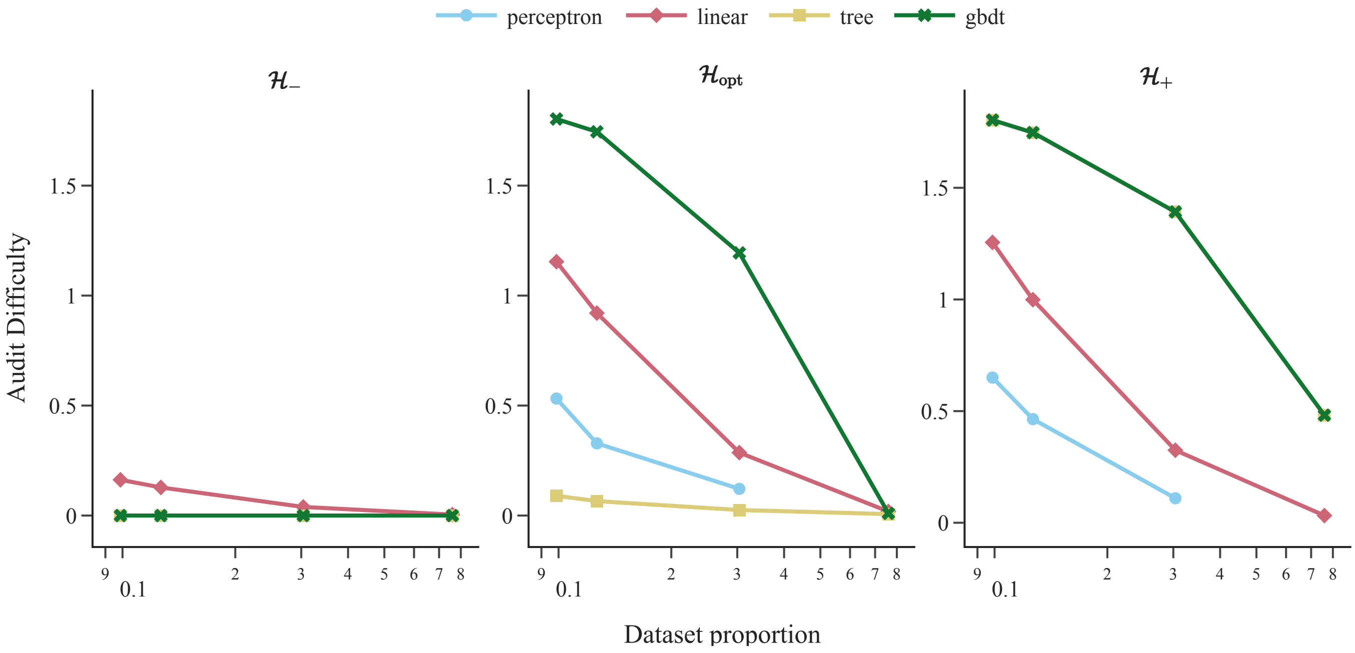

Figure 9. Evolution of the μ-diameter with the size of the audit set S represented as a proportion of the total dataset size for the `AdultIncome` dataset. Each line represents an audited model, whose hyperparameters are either tuned for the best generalization, either tuned for the highest capacity or tuned for the lowest capacity. For each (model, hyperparameter) couple, the μ-diameter is averaged over 15 audit datasets.

Appendix

Proofs for Section III (The competitive effectiveness of random audits)

Theorem 1 (No need to aim)**.** *Let $\mathcal{H}=\{0,1\}^{\mathcal{X}}$. For any audit set $S\subseteq\mathcal{X}$ and hypothesis $h\in\mathcal{H}$,*

$$\begin{aligned}\operatorname{diam}_\mu \mathcal{H}(h,S) = 2 - \big(&\mathbb{P}\left(X\in S|X_A=1\right)\\ &+\mathbb{P}\left(X\in S|X_A=0\right)\big)\end{aligned}$$

Proof. The proof is executed in 4 steps: decomposition of the value of $\mu(h,\mathcal{X})$ on S and $\overline{S}$, decomposition of the μ-diameter on S and $\overline{S}$, solving the optimization on the decomposed problems and conclusion.

Step 1: Decompose μ

For any $h\in\mathcal{H}$, $S\subseteq\mathcal{X}$

$$\begin{aligned}
\mu(h,\mathcal{X}) &= \mathbb{P}\left(h(X)=1|X_A=1\right)-\mathbb{P}\left(h(X)=1|X_A=0\right)\\
&= \mathbb{P}\left(h(X)=1|X_A=1,X\in S\right)\underbrace{\mathbb{P}\left(X\in S|X_A=1\right)}_{\alpha}\\
&\qquad+\mathbb{P}\left(h(X)=1\middle|X_A=1,X\in\overline{S}\right)\underbrace{\mathbb{P}\left(X\in\overline{S}\middle|X_A=1\right)}_{1-\alpha}\\
&\qquad-\mathbb{P}\left(h(X)=1|X_A=0,X\in S\right)\underbrace{\mathbb{P}\left(X\in S|X_A=0\right)}_{\alpha-\delta}\\
&\qquad-\mathbb{P}\left(h(X)=1\middle|X_A=0,X\in\overline{S}\right)\underbrace{\mathbb{P}\left(X\in\overline{S}\middle|X_A=0\right)}_{1-\alpha+\delta}\\
&=\alpha\underbrace{\big(\mathbb{P}\left(h(X)=1|X_A=1,X\in S\right)-\mathbb{P}\left(h(X)=1|X_A=0,X\in S\right)\big)}_{\mu(h,S)}\\
&\qquad+(1-\alpha)\underbrace{\left(\mathbb{P}\left(h(X)=1\middle|X_A=1,X\in\overline{S}\right)-\mathbb{P}\left(h(X)=1\middle|X_A=0,X\in\overline{S}\right)\right)}_{\mu(h,S)}\\
&\qquad+\delta\left(\mathbb{P}\left(h(X)=1|X_A=0,X\in S\right)-\mathbb{P}\left(h(X)=1\middle|X_A=0,X\in\overline{S}\right)\right)\\
&=\alpha\mu(h,S)+(1-\alpha)\mu(h,\overline{S})\\
&\qquad+\delta\left(\mathbb{P}\left(h(X)=1|X_A=0,X\in S\right)-\mathbb{P}\left(h(X)=1\middle|X_A=0,X\in\overline{S}\right)\right)
\end{aligned}\tag{12}$$

Step 2: Decompose the μ-diameter

For any $h,h'\in\mathcal{H}(h^*,S)$,

$$\begin{aligned}
\mu(h,\mathcal{X})-\mu(h',\mathcal{X}) = \alpha\ &\underbrace{\left(\mu(h,S)-\mu(h',S)\right)}_{=0\text{ since }h(S)=h'(S)=h^*(S)}\ +(1-\alpha)\left(\mu(h,\overline{S})-\mu(h',\overline{S})\right)\\
&+\delta\underbrace{\big(\mathbb{P}\left(h(X)=1|X_A=0,X\in S\right)-\mathbb{P}\left(h'(X)=1|X_A=0,X\in S\right)\big)}_{=0\text{ since }h(S)=h'(S)=h^*(S)}\\
&+\delta\left(\mathbb{P}\left(h'(X)=1\middle|X_A=0,X\in\overline{S}\right)-\mathbb{P}\left(h(X)=1\middle|X_A=0,X\in\overline{S}\right)\right)
\end{aligned}$$

Using the definition and separability of the μ-diameter, we have

$$\operatorname{diam}_\mu\left(h^*,S\right)=\max_{h\in\mathcal{H}(h^*,S)}\mu(h,S)-\min_{h'\in\mathcal{H}(h^*,S)}\mu(h,S)\tag{13}$$

Therefore, by grouping the terms that depend on h and h' in the previous development:

$$\begin{aligned}
\operatorname{diam}_\mu\left(h^*,S\right)=\ &\max\nolimits_{h\in\mathcal{H}(h^*,S)}\left[(1-\alpha)\mu(h,\overline{S})-\delta\,\mathbb{P}\left(h(X)=1\middle|X_A=0,X\in\overline{S}\right)\right]\\
&-\ \min\nolimits_{h'\in\mathcal{H}(h^*,S)}\left[(1-\alpha)\mu(h',\overline{S})-\delta\,\mathbb{P}\left(h'(X)=1\middle|X_A=0,X\in\overline{S}\right)\right]
\end{aligned}\tag{14}$$

Step 3: Solve each optimization problem

To solve the two optimization problems, we come back to the definition of μ.

$$\text{🟧} = \max_{h\in\mathcal{H}(h^*,S)} \left\{(1-\alpha)\,\mathbb{P}\left(h(X)=1\middle|X_A=1,X\in\overline{S}\right)-(1-\alpha+\delta)\,\mathbb{P}\left(h(X)=1\middle|X_A=0,X\in\overline{S}\right)\right\} \tag{15}$$

$$= -(1-\alpha+\delta) + \max_{h\in\mathcal{H}(h^*,S)} \left\{(1-\alpha)\,\mathbb{P}\left(h(X)=1\middle|X_A=1,X\in\overline{S}\right)+(1-\alpha+\delta)\,\mathbb{P}\left(h(X)=0\middle|X_A=0,X\in\overline{S}\right)\right\} \tag{16}$$

Similarly,

$$\text{🟦} = \min_{h\in\mathcal{H}(h^*,S)} \left\{(1-\alpha)\,\mathbb{P}\left(h(X)=1\middle|X_A=1,X\in\overline{S}\right)-(1-\alpha+\delta)\,\mathbb{P}\left(h(X)=1\middle|X_A=0,X\in\overline{S}\right)\right\} \tag{17}$$

$$= -(1-\alpha+\delta) + \min_{h\in\mathcal{H}(h^*,S)} \left\{(1-\alpha)\,\mathbb{P}\left(h(X)=1\middle|X_A=1,X\in\overline{S}\right)+(1-\alpha+\delta)\,\mathbb{P}\left(h(X)=0\middle|X_A=0,X\in\overline{S}\right)\right\} \tag{18}$$

We write $h^{\uparrow}$ (resp. $h^{\downarrow}$) the minimizer of 🟧 (resp. 🟦).

$$h^{\uparrow}(x) = \begin{cases} 1 \text{ if } x_A = 1 \text{ and } x\in\overline{S} \\ 0 \text{ if } x_A = 0 \text{ and } x\in\overline{S} \\ 0 \text{ else} \end{cases} \tag{19}$$

$$h^{\downarrow}(x) = \begin{cases} 1 \text{ if } x_A = 0 \text{ and } x\in\overline{S} \\ 0 \text{ if } x_A = 1 \text{ and } x\in\overline{S} \\ 0 \text{ else} \end{cases} \tag{20}$$

The optimizers $h^{\uparrow}$ and $h^{\downarrow}$ yield the optima

$$\text{🟧} = -(1-\alpha+\delta) + (1-\alpha)\underbrace{\mathbb{P}\left(h^{\uparrow}(X)=1\middle|X_A=1,X\in\overline{S}\right)}_{=1} + (1-\alpha+\delta)\underbrace{\mathbb{P}\left(h^{\uparrow}(X)=0\middle|X_A=0,X\in\overline{S}\right)}_{=1} \tag{21}$$

$$= 1-\alpha \tag{22}$$

$$\text{🟦} = -(1-\alpha+\delta) + (1-\alpha)\underbrace{\mathbb{P}\left(h^{\downarrow}(X)=1\middle|X_A=1,X\in\overline{S}\right)}_{=0} + (1-\alpha+\delta)\underbrace{\mathbb{P}\left(h^{\downarrow}(X)=0\middle|X_A=0,X\in\overline{S}\right)}_{=0} \tag{23}$$

$$= -(1-\alpha+\delta) \tag{24}$$

Step 4: Conclusion

$$\operatorname{diam}_\mu \mathcal{H}(h^*,S) = \text{🟧} - \text{🟦} \tag{25}$$

$$= (1-\alpha) + (1-\alpha+\delta) \tag{26}$$

$$= 2 - \big(\mathbb{P}(X\in S|X_A=1) + \mathbb{P}(X\in S|X_A=0)\big) \tag{27}$$

□

Theorem 2 (μ-diameter of $\mathcal{D}_m$). *Consider $S\subseteq\mathcal{X}$, $d\in\mathcal{D}_m$. Note $m' = m - |x\in S : d(x)=1|$. The μ-diameter of $\mathcal{D}_m(d,S)$ is given by*

$$\operatorname{diam}_\mu \mathcal{D}_m(d,S) = \frac{\min(\left|\mathcal{X}_A\cap\overline{S}\right|, m')}{|\mathcal{X}_A|} + \frac{\min(\left|\overline{\mathcal{X}_A}\cap\overline{S}\right|, m')}{\left|\overline{\mathcal{X}_A}\right|}$$

Proof. In the proof of Theorem 1, we established the following identity (for any hypothesis class thus for $\mathcal{D}_m$, and for any S and d^*):

$$\operatorname{diam}_\mu \mathcal{D}(d^*,S) \tag{28}$$

$$= \max_{d\in\mathcal{D}(d^*,S)} \left\{\mathbb{P}\left(X\in\overline{S}\middle|X_A=1\right)\mathbb{P}\left(d(X)=1\middle|X_A=1,X\in\overline{S}\right) + \mathbb{P}\left(X\in\overline{S}\middle|X_A=0\right)\mathbb{P}\left(d(X)=0\middle|X_A=0,X\in\overline{S}\right)\right\}$$

$$- \min_{d\in\mathcal{D}(d^*,S)} \left\{\mathbb{P}\left(X\in\overline{S}\middle|X_A=1\right)\mathbb{P}\left(d(X)=1\middle|X_A=1,X\in\overline{S}\right) + \mathbb{P}\left(X\in\overline{S}\middle|X_A=0\right)\mathbb{P}\left(d(X)=0\middle|X_A=0,X\in\overline{S}\right)\right\} \tag{29}$$

First, observe that in the two optimization problems, the value of the objective function does not depend on the values of d on S. Moreover, the choices of the labels $d(x)$ for $x\in\overline{S}$ can be made freely as long as d does not have more than $m' = m - |x\in S : d^*(x)=1|$ "1"s (because it has to use $|x\in S : d^*(x)=1|$ slots of memory to store the answers of d^* on S).

Therefore, the dictionary that optimizes 🟧 is built by storing as many "1"s in d on the entries of $x \in \mathcal{X}_A \cap \overline{S}$ within the limits of the m' slots left. This leads to

$$\text{🟧} = \mathbb{P}\left(X \in \overline{S} \middle| X_A = 1\right) \frac{\min(\left|\mathcal{X}_A \cap \overline{S}\right|, m')}{\left|\mathcal{X}_A \cap \overline{S}\right|} + \mathbb{P}\left(X \in \overline{S} \middle| X_A = 0\right) * 1 \tag{30}$$

Next, rewriting as a maximization problem, we get

$$\begin{aligned}&\text{🟦} = \mathbb{P}\left(X \in \overline{S} \middle| X_A = 1\right) + \mathbb{P}\left(X \in \overline{S} \middle| X_A = 0\right)\\ &- \min_{d \in \mathcal{D}(d^*,S)} \left\{\mathbb{P}\left(X \in \overline{S} \middle| X_A = 1\right) \mathbb{P}\left(d(X) = 0 \middle| X_A = 1, X \in \overline{S}\right) + \mathbb{P}\left(X \in \overline{S} \middle| X_A = 0\right) \mathbb{P}\left(d(X) = 1 \middle| X_A = 0, X \in \overline{S}\right)\right\}\end{aligned} \tag{31}$$

Similar to the case of 🟧, the dictionary that optimizes 🟦 is built by storing as many "1"s in d on the entries of $x \in \overline{\mathcal{X}_A} \cap \overline{S}$ withing the limits of the m' slots left. This leads to

$$\text{🟦} = \mathbb{P}\left(X \in \overline{S} \middle| X_A = 1\right) + \mathbb{P}\left(X \in \overline{S} \middle| X_A = 0\right) \tag{32}$$

$$- \mathbb{P}\left(X \in \overline{S} \middle| X_A = 1\right) * 1 - \mathbb{P}\left(X \in \overline{S} \middle| X_A = 0\right) \frac{\min(\left|\overline{\mathcal{X}_A} \cap \overline{S}\right|, m')}{\left|\overline{\mathcal{X}_A} \cap \overline{S}\right|} \tag{33}$$

Composing the expressions of 🟧 and 🟦, we get

$$\operatorname{diam}_\mu \mathcal{D}(d^*, S) = \mathbb{P}\left(X \in \overline{S} \middle| X_A = 1\right) \frac{\min(\left|\mathcal{X}_A \cap \overline{S}\right|, m')}{\left|\mathcal{X}_A \cap \overline{S}\right|} + \mathbb{P}\left(X \in \overline{S} \middle| X_A = 0\right) \frac{\min(\left|\overline{\mathcal{X}_A} \cap \overline{S}\right|, m')}{\left|\overline{\mathcal{X}_A} \cap \overline{S}\right|} \tag{34}$$

Here, it is important to understand that in the notations $\mathbb{P}(X \in S)$ or $\mathbb{P}(d(X) = 1)$, X is a random variable taking values in $\mathcal{X}$ with a uniform probability. Therefore, $\mathbb{P}\left(X \in \overline{S} \middle| X_A = 1\right) = \frac{\left|\mathcal{X}_A \cap \overline{S}\right|}{|\mathcal{X}_A|}$ and $\mathbb{P}\left(X \in \overline{S} \middle| X_A = 0\right) = \frac{\left|\overline{\mathcal{X}_A} \cap \overline{S}\right|}{\left|\overline{\mathcal{X}_A}\right|}$, which simplifies the previous equation

$$\operatorname{diam}_\mu \mathcal{D}(d^*, S) = \frac{\min(\left|\mathcal{X}_A \cap \overline{S}\right|, m')}{|\mathcal{X}_A|} + \frac{\min(\left|\overline{\mathcal{X}_A} \cap \overline{S}\right|, m')}{\left|\overline{\mathcal{X}_A}\right|} \tag{35}$$

□

Corollary 1 (Benign overfitting and μ-diameter). *Let $\mathcal{X}$ and $\mathcal{H} \subseteq \{0,1\}^{\mathcal{X}}$ be any input space and hypothesis class. Assume that $\mathcal{H}$ exhibits benign overfitting with respect to the sensitive attribute X_A and its opposite $1 - X_A$* [4]*, then $\forall d \leq d_0, S \in \mathcal{X}^d$,*

$$\begin{aligned}\operatorname{diam}_\mu \mathcal{H}(h^*, S) \geq& \, \mathbb{P}(X \in S | X_A = 1) + \mathbb{P}(X \in S | X_A = 0)\\ &- 2\,\mathbb{P}(X \in S) - 2\epsilon(1 - \mathbb{P}(X \in S))\end{aligned}$$

Proof. Note $\alpha_1 = \mathbb{P}\left(X \in \overline{S} \middle| X_A = 1\right)$ and $\alpha_0 = \mathbb{P}\left(X \in \overline{S} \middle| X_A = 0\right)$. In the proof of Theorem 1, we established the following equality:

$$\operatorname{diam}_\mu \mathcal{H}(h^*, S) \tag{36}$$

$$\begin{aligned}&= \max_{h \in \mathcal{H}(h^*,S)} \left\{\alpha_1 \mathbb{P}\left(h(X) = 1 \middle| X_A = 1, X \in \overline{S}\right) + \alpha_0 \mathbb{P}\left(h(X) = 0 \middle| X_A = 0, X \in \overline{S}\right)\right\}\\ &- \min_{h \in \mathcal{H}(h^*,S)} \left\{\alpha_1 \mathbb{P}\left(h(X) = 1 \middle| X_A = 1, X \in \overline{S}\right) + \alpha_0 \mathbb{P}\left(h(X) = 0 \middle| X_A = 0, X \in \overline{S}\right)\right\}\end{aligned} \tag{37}$$

And

$$\text{🟦} = \alpha_1 + \alpha_0 - \max_{h \in \mathcal{H}(h^*,S)} \left\{\alpha_1 \mathbb{P}\left(h(X) = 0 \middle| X_A = 1, X \in \overline{S}\right) + \alpha_0 \mathbb{P}\left(h(X) = 1 \middle| X_A = 0, X \in \overline{S}\right)\right\} \tag{38}$$

Since $\mathcal{H}$ exhibits benign overfitting with respect to the sensitive attribute and $|S| <= d_0$, there exists $h \in \mathcal{H}(h^*, S)$ such that $\mathbb{P}\left(h(X) = X_A \middle| X \in \overline{S}\right) = 1 - \epsilon$. Moreover,

$$\begin{aligned}\mathbb{P}\left(h(X) = X_A \middle| X \in \overline{S}\right) =& \, \mathbb{P}\left(h(X) = 1 \middle| X \in \overline{S}, X_A = 1\right) \mathbb{P}\left(X_A = 1 \middle| X \in \overline{S}\right)\\ &+ \mathbb{P}\left(h(X) = 0 \middle| X \in \overline{S}, X_A = 0\right) \mathbb{P}\left(X_A = 0 \middle| X \in \overline{S}\right)\end{aligned} \tag{39}$$

$$\begin{aligned}=& \alpha_1 \frac{\mathbb{P}(X_A = 1)}{\mathbb{P}\left(X \in \overline{S}\right)} \mathbb{P}\left(h(X) = 1 \middle| X \in \overline{S}, X_A = 1\right)\\ &+ \alpha_0 \frac{\mathbb{P}(X_A = 0)}{\mathbb{P}\left(X \in \overline{S}\right)} \mathbb{P}\left(h(X) = 1 \middle| X \in \overline{S}, X_A = 0\right)\end{aligned} \tag{40}$$

[4] That is, Definition 2 holds for $c = x_A$ and $c = 1 - x_A$

Since $\mathbb{P}\left(X_A = 0\right) + \mathbb{P}\left(X_A = 1\right) = 1$, $\mathbb{P}\left(X_A = 0\right) \geq 0$ and $\mathbb{P}\left(X_A = 1\right) \geq 0$, we have

$$\begin{aligned}
&\alpha_1 \, \mathbb{P}\left(h(X) = 1 \middle| X \in \overline{S}, X_A = 1\right) + \alpha_0 \, \mathbb{P}\left(h(X) = 1 \middle| X \in \overline{S}, X_A = 0\right) \\
&\geq \mathbb{P}\left(X_A = 1\right) \alpha_1 \, \mathbb{P}\left(h(X) = 1 \middle| X \in \overline{S}, X_A = 1\right) + \mathbb{P}\left(X_A = 0\right) \alpha_0 \, \mathbb{P}\left(h(X) = 1 \middle| X \in \overline{S}, X_A = 0\right) && (41) \\
&= (1 - \epsilon) \, \mathbb{P}\left(X \in \overline{S}\right) && (42)
\end{aligned}$$

Therefore,

$$\text{🟧} \geq (1 - \epsilon) \, \mathbb{P}\left(X \in \overline{S}\right) \tag{43}$$

With the same arguments, we prove

$$\text{🟦} \leq \alpha_0 + \alpha_1 - (1 - \epsilon) \, \mathbb{P}\left(X \in \overline{S}\right) \tag{44}$$

To conclude,

$$\operatorname{diam}_\mu \mathcal{H}(h^*, S) \geq 2(1 - \epsilon) \, \mathbb{P}\left(X \in \overline{S}\right) - (\alpha_0 + \alpha_1) \tag{45}$$

□

How is the μ-diameter measured in practice

As originally defined in [1] and following the definition of the μ-diameter, the evaluation of $\operatorname{diam}_\mu(S, h^*)$ requires to solve the following optimization problem:

$$\begin{aligned}
\max_{h, h'} \quad & |\mu(h, S) - \mu(h')| && (46) \\
\text{subject to} \quad & h(x) = h'(x) = h^*(x) \quad \forall x \in S && (47)
\end{aligned}$$

This problem be separated in two optimization problems: the maximization/minimization over $h \in \mathcal{H}$ of $\mu(h, S)$ under the constraint that $\forall x \in S, h(x) = h^*(x)$.

$$\begin{aligned}
\max_h / \min_h \quad & \mu(h, S) && (48) \\
\text{subject to} \quad & h(x) = h^*(x) \quad \forall x \in S && (49)
\end{aligned}$$

As proposed by [1], we use the method introduced by [56] to reframe this constrained optimization problem as a sequence of weighted classification tasks. Then, we use off-the-self estimators from scikit-learn and XGBoost to perform the optimization with the appropriate weights.