An Equivalence Between Private Classification and Online Prediction: Proof Sketch and Extension

Jiaqing Xie
jiaxie@student.ethz.ch

Yucheng Sun

Abstract

This paper aims to build an equivalence between online learning and private learning. It is equivalent to prove whether online learning leads to private learning and vice versa. The second direction has been supported by previous research. The first direction is proven in this report. As an alternative of online learning, this paper introduces finite littlestone classes. This paper then divides the proof into two parts: finite littlestone dimension to stable learnability and stable learnability to differential private learning. It also answers some followed problems.

1 Introduction

Online learning and private learning are two important and hot topics in machine learning theory. There's an open question that whether online learning and private learning are equivalent to each other [1], which is equivalent to state that given a class of predictors $\mathcal{H}$, if $\mathcal{H}$ is differentially private learnable, then it must be online learnable, and vice versa. Importantly, we restrict the learning problems to binary classification problems and leave the regression setting as future works. In order to prove this, two iff. conditions are considered.

DP-learning $\Leftrightarrow$ Littlestone dimension is finite Previous work has proved that a private PAC learner hinted a corresponding finite littlestone dimension Alon et al. [2019b]. The selected paper supplemented the converse direction Bun et al. [2020], therefore combining two results suffices to show the equivalence of private learning and finite littlestone dimension. In this project, we mainly focus on the latter part. Global stability is defined in this paper as an intermediate step for proofs. When proving from finite littlestone dimension to global stable learning, Standard Optimal Algorithm (**SOA**) is involved by operating on a pair of samples. We want to prove that such **SOA**$(\cdot)$ as an algorithm satisfies the definition of global stability that is also generalized well for some finite number of samples. When proving from global stable learning to private learning, stable histograms are involved and we need to construct such a private learner with some privacy/accuracy parameters satisfying stable conditions, as well as satisfying the statement of a generic private learner Kasiviswanathan et al. [2011].

If H is online-learnable $\Leftrightarrow$ its Littlestone dimension is finite Previous works have stated this conclusion with regard to the regret which depends on corresponding littlestone dimension and number of samples Littlestone [1988], Ben-David et al. [2009]. In our selected paper, authors also restated this conclusion when introducing the relationship between online learning and finite Littlestone dimension Bun et al. [2020].

The organization of this paper is structured as follows: Section 2 presents an overview of current research in PAC learning and differential privacy in learning. Section 3 delves into the principles of online learning, global stability, and differential privacy. Section 4 is dedicated to demonstrating that

[1] For simplicity, we may write "approximate private PAC learning" as "differentially private learning" or "DP learning" in this report. However, the hardness of proper pure private learning, improper pure private learning, approximate private learning, and are all strongly separated in the PAC model.

Preprint. Under review.

a finite Littlestone dimension implies stable learnability. Section 5 establishes that stable learnability leads to differential privacy in learning. Finally, Section 6 addresses subsequent questions related to this topic.

2 Extensive Literature Review

This paper is related to characterizing the sample complexity of approximate differential private learning and building a connection between it and online learning. While online learning has been extensively studied and well understood, differentially private learning is comparatively new and less understood. Before moving on, we recall that by definition approximate differential private learning is not harder than pure differential private learning, and improper learning is not harder than proper learning.

A line of work has aimed at understanding the sample complexity of differentially private PAC learning. Kasiviswanathan et al. [2011] first introduced the concept of differentially private PAC learning and proposed a generic learner based on Occam's razor algorithm, which requires $O\left(\frac{\log H}{\varepsilon}\right)$ samples to learn a hypothesis class H. This result is not satisfying because it excludes infinite hypothesis classes. Beimel et al. [2014] studied a particular hypothesis class whose VC dimension is 1 and showed that the sample complexity of *proper* pure private PAC learning on this hypothesis class can not be upper bounded by its VC dimension. They also showed that the sample complexity of *improper* pure private PAC learning on this hypothesis class only requires $O(1)$ samples.

Beimel et al. [2019] introduced a notion called probabilistic representation and proposed a new measure $RepDim(H)$ of hypothesis classes H, which is the size of the smallest probabilistic representation of the concept class. They showed that the sample complexity of improper pure private PAC learning is $\Theta(RepDim(H))$.

Recall that a measure of hypothesis classes is called the little stone dimension, and it is used to characterize the sample complexity of online learning. Feldman and Xiao [2015] showed that the little stone dimension is also a lower bound of sample complexity of pure private PAC learning. This work showed a connection between online learning and private learning: private learning is not easier than online learning.

All the work mentioned previously is about pure differential private PAC learning. Since approximate private learning is easier than pure private learning, one may hope it can learn a "larger" hypothesis class. Bun et al. [2015] studied proper approximate private PAC learning on threshold functions, and established a lower bound of sample complexity. This lower bound implies that the VC dimension is also not an upper bound of sample complexity of approximate private PAC learning. This showed that approximate private PAC learning is also strictly harder than non-private PAC learning. Together with Alon et al. [2019a], we know the sample complexity of proper and improper approximate private PAC learning is lower bounded by $\Omega(log^*(Ldim(H)))$.

Finally, this paper proved that the sample complexity of approximate private learning can also be upper bounded by the little stone dimension ($2^{O(Ldim(H))}$). It achieved this by building a differentially private learning algorithm using an online learning algorithm. Thus, an equivalence between online learning and private learning was established.

3 Preliminaries

As stated, it is difficult to directly prove from online learning to private learning, therefore an introduction of the Littlestone dimension as an intermediate step is necessary. In this section, we first complete the proof that an online learner implies a finite Littlestone dimension. Following that, we define global stability, generalization, differential privacy, and differential private PAC learning in order to provide intuitions for proving that every concept class with finite Littlestone Dimension can be learned by a differential private learner.

3.1 Online Learning and Littlestone Dimension

Online Learning Online learning is to make real-time predictions on a sequence of data. Suppose we have a hypothesis class $\mathcal{H} = \{h : X \to \{\pm 1\}\}$, and a sequence of data $(x_1, y_1), ..., (x_n, y_n) \in$

$X \times \{\pm 1\}$. From the samples, one instance is observed. The predicted label $\hat{y_t}$ for this instance is given by a $h \in \mathcal{H}$. As soon as true label y_t is observed, loss is calculated and the hypothesis class is updated according to some metrics. The loop is iterated until it has been executed for n times. The main goal is to minimize the regret, which refers to the number of mistakes compared to the best hypothesis in $\mathcal{H}$:

$$R(n) = \sum_{t=1}^{n} 1[y_t \neq \hat{y_t}] - \min_{h^* \in H} \sum_{t=1}^{n} 1[y_t \neq h^*]$$

Littlestone Dimension Firstly we introduce mistake bound in online learning and state how this mistake bound is related to the required Littlestone dimension.

Definition 1 (Mistake bound)**.** *Let $\mathcal{H}$ be a hypothesis class. $\mathcal{M}$ is an online learning algorithm. Given any sequence S = $(x_1, y_1), ..., (x_n, y_n) \in X \times \{\pm 1\}$. We denote $MA(S)$ as the number of mistakes that $\mathcal{M}$ makes on samples S. We denote MA($\mathcal{H}$) as the supremum of $MA(S)$ over all possible S, which is referred to as the* ***mistake bound****.*

If we apply consistent or halving online learning algorithm Shalev-Shwartz and Ben-David [2014], we could ensure that the mistake bound is upper bounded. On the contrary, Littlestone dimension (denoted by$Ldim(\cdot)$) is the lower bound of mistake bound, proposed by Littlestone which characterizes learnability:

Theorem 1 (MA($\mathcal{H}$) $\geq Ldim(\mathcal{H})$ Littlestone [1988])**.** *Any online learning algorithm might make at least Ldim($\mathcal{H}$) mistakes. And there exists an online learning algorithm that makes exactly $Ldim(\mathcal{H})$ mistakes. This holds only for realizable settings.*

The online learning algorithm that could match MA($\mathcal{H}$) with $Ldim(\mathcal{H})$ is called Standard Optimal Algorithm (SOA). This algorithm is also essential for proving a global stable learner which will be mentioned later. A mistake tree is constructed to indicate Littlestone dimension, which is equivalent to the depth of the largest complete tree which is shattered by $\mathcal{H}$, where each sample is mapped to each node in this complete tree. It makes sure that the littlestone dimension is reduced by 1 by each round, so overall it makes up to $Ldim(\mathcal{H})$ mistakes. Taking MA($\mathcal{H}$) $\geq Ldim(\mathcal{H})$ and MA($\mathcal{H}$) $\leq Ldim(\mathcal{H}) \Rightarrow$ MA($\mathcal{H}$) $= Ldim(\mathcal{H})$. Here we ignore the detailed description of the SOA algorithm.

In theorem 1 we state that SOA only holds for realizable settings. However, it could also be extended to non-realizable settings by case analysis on new coming sample (x_{t+1}, y_{t+1}) since we already know that $(x_1, y_1), ..., (x_t, y_t) \in X \times \{\pm 1\}$ are realizable. If the new coming sample still makes the hypothesis realizable, we still apply the original update rule in SOA. If new coming sample makes it unrealizable, we set $h_{t+1}(x_t) = h_t(x_t)$ and $h_{t+1}(x_{t+1}) = y_{t+1}$ to make it realizable.

Online learning ⇔ Finite Littlestone dimension

Theorem 2 (Online learning ⇔ Bounded Regret w.r.t. $Ldim(\cdot)$ Ben-David et al. [2009])**.** *If $LDim(\cdot)$ is finite, then there exist an online learning algorithm $\mathcal{M}$ whose regret is bounded by $O(\sqrt{Ldim(H) \cdot T})$ and vice versa.*

Therefore it is equivalent to say that $\mathcal{H}$ is online learnable iff. its $Ldim(\cdot)$ is finite. Then the remaining parts are proving from the finite Littlestone dimension to a private learner.

3.2 Global Stability

There are many types of stability, including uniform hypothesis stability, PAC-Bayes stability and many others. In this research, we mainly focus on global stability. It's different from the previously mentioned stability since such stability requires resampling the entire input instead of changing samples in a local way. Here we give the definition of global stability.

Definition 2 (Global Stability)**.** *A randomized algorithm $\mathcal{M}$: $\mathcal{X}^n \rightarrow \mathcal{R}^k$ is called (n, η) globally stable if for sample size $n \in \mathbb{N}, \eta > 0$, and a hypothesis $h \in \mathcal{H}$ where $\mathcal{H}$ is a finite hypothesis class such that the following probability holds:*

$$\Pr_{S \sim \mathcal{D}^n}[\mathcal{M}(S) = h] \geq \eta$$

where D is a realizable distribution.

It holds when applying learning algorithms to those finite hypothesis classes. Global stability indicates the following property regarding generalization.

Lemma 1 (Global stability towards generalization)**.** *Hypothesis class is based on the setting of binary classification, where we suppose the class $\mathcal{H} \in \{\pm 1\}^X$. We assume that our $\mathcal{M}$ is realizable which means that the generalization error is zero for any sample S: $loss_S(\mathcal{M}(S)) = 0$. If $\mathcal{M}$ is (n, η) globally stable, then considering $h \in \mathcal{H}$ in global stability, the generalization error of h on realizable $\mathcal{D}$ is bounded by:*

$$loss_{\mathcal{D}}(h) \leq \frac{\ln(\frac{1}{\eta})}{n}$$

Proof. Let $\alpha = \text{loss}_{\mathcal{D}}(h)$. Event 1 E_1: h is consistent with input S $\Rightarrow \Pr[E_1] = (1-\alpha)^n$. Event 2 E_2: $\mathcal{M}(S) = h \Rightarrow \Pr[E_2] \geq \eta$ (global stability). We know that if h is realizable, then it must be consistent with input, therefore we have: $\eta \leq \Pr[E_2] \leq \Pr[E_1] = (1-\alpha)^n \leq e^{-\alpha n}$. Solve for α we have $\text{loss}_{\mathcal{D}}(h) = \alpha \leq \frac{\ln(\frac{1}{\eta})}{n}$. This means that if a online learner is a global stable learner, then it should generalize well.

This gives us the intuition when proving from finite Littlestone dimension to a global stable learner. Suppose finite Littlestone dimension $Ldim(\cdot)$ is equal to d. Then we are going to find 1) a specific finite η which is related to d, a specific largest number of n samples that are generated from distribution D, which is related to d and η and is also finite. If these two parameters are found under the finite Littlestone dimension setting, then the direction finite $Ldim(\cdot) \Rightarrow$ global stable learner holds.

3.3 Differential Privacy

Definition 3 (Differential privacy)**.** *A randomized algorithm $\mathcal{M} \colon \mathcal{X}^n \to \mathcal{R}^k$ is said to be (ε, δ)-*differentially private *if for all measurable $S \subseteq \mathcal{R}^k$ and all neighboring datasets $x, y \in \mathcal{X}$:*

$$\Pr[\mathcal{M}(x) \in S] \leq e^{\varepsilon} \Pr[\mathcal{M}(y) \in S] + \delta\,,$$

where two datasets x, y are said to be neighboring *if* $\text{dist}(x, y) \leq 1$ *(i.e., for* $\text{dist}(,)$ *being the Hamming distance, if they only differ in (at most) one entry).*

There is a nice property called post-processing property.

Theorem 3 (Post-processing)**.** *Let $\mathcal{M} \colon \mathcal{X}^n \to \mathcal{R}^k$ be an (ε, δ)-*differentially private *algorithm. Let $f \colon \mathcal{R}^k \to \mathcal{R}^k$ be an arbitrary mapping. Then $f \circ \mathcal{M}$ is also (ε, δ)-*differentially private.

We also recall the composition theorem of differential privacy.

Theorem 4 (Composition theorem)**.** *Let $\mathcal{M}_1 \colon \mathcal{X}^n \to \mathcal{R}^k$ be an $(\varepsilon_1, \delta_1)$-*differentially private *algorithm and $\mathcal{M}_2 \colon \mathcal{X}^n \to \mathcal{R}^k$ be an $(\varepsilon_2, \delta_2)$-*differentially private-*algorithm, then their combination $M_{1,2}(x) = (\mathcal{M}_1(x), \mathcal{M}_2(x))$ is $(\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)$-*differentially private.

A randomized algorithm is said to satisfy pure and approximate differential privacy if $\delta = 0$ and only satisfy approximate differential privacy otherwise. A differentially private PAC learning algorithm is just any PAC learning algorithm which satisfies differentially privacy constraints.

As mentioned in section 2, Kasiviswanathan et al. [2011] proposed a generic learner which requires $O\left(\frac{\log H}{\varepsilon}\right)$ samples to learn a hypothesis class H. This is stated formally in the following theorem. This learner is used in the proof in this paper.

Theorem 5 (Generic learner)**.** *Let $H \subseteq \{\pm 1\}^X$ be a collection of hypothesis. For*

$$n = O\left(\frac{\log(|H|) + \log(1/\beta)}{\alpha\varepsilon}\right)$$

there exists an $(\varepsilon, \delta)-$differentially private algorithm GenericLearner : $(X \times \{\pm 1\})^n \to H$ such that the following holds.

Let D be a distribution over $(X \times \{\pm 1\})$ such that there exists $h^* \in H$ with $loss_D(h^*) \leq \alpha$. Then on input $S \sim D^n$, algorithm GenericLearner outputs, with probability at least $1 - \beta$, a hypothesis $\hat{h} \in H$ such that $loss_D(\hat{h}) \leq 2\alpha$.

Another technique in differential privacy that is used is called stable histogram.

Theorem 6 (Stable Histogram (Korolova et al. [2009]))**.** *Let X be any data domain. For*

$$n \geq O\bigg(\frac{\log(1/\eta\beta\delta)}{\eta\varepsilon}\bigg)$$

there exists an (ε, δ)-differentially private *algorithm Hist which, with probability at least* $1-\beta$*, on input* $S = (x_1, ..., x_n)$ *outputs a list* $L \subset X$ *and a sequence of estimates* $\alpha \in [0,1]^{|L|}$ *such that*

- *Every x with $freqS(x) \geq \eta$ appears in L and*
- *For every $x \in L$, the estimate a_x satisfies $|a_x - freqS(x)| \leq \eta$*

4 Finite Littlestone dimension implies global stable learning

Theorem 7 (Bounded $Ldim(\cdot) \Rightarrow$ Global Stable Learner)**.** *Let $\mathcal{H}$ be a hypothesis class woth a finite Littlestone dimension d ≥ 1. Assume that* $\alpha > 0$*. Set*

$$m = 2^{2^{d+2}+1}4^{d+1} \cdot \lceil \frac{2^{d+2}}{\alpha} \rceil$$

There exists a randomized alogrithm $\mathcal{M} : X \times \{\pm 1\}^m \to \{\pm 1\}^X$ and a hypothesis f such that:

$$\underset{S \sim \mathcal{D}^m}{Pr}[\mathcal{M}(S) = f] \geq \frac{1}{(d+1)2^{2^d+1}} \text{ and } loss_{\mathcal{D}}(f) \leq \alpha$$

where $\mathcal{D}$ is a realizable distribution.

Algorithm 1 Choosing $\mathcal{D}_t$

```
 1: $\mathcal{D}_t \leftarrow \mathcal{D}_t(t, n)$ are defined by induction on $t$:
 2: $\mathcal{D}_0$: outputs the empty sample $\emptyset$ with probability $1$
 3: if $Pr[SOA(S \circ T) = f] \geq 2^{-2^{d+2}}$ then
 4:     $\mathcal{D}_k$ is un-defined if $\mathcal{D}_{k-1}$ is un-defined
 5: end if
 6: for each time step $t$ do:
 7:     Draw $S_0, S_1 \sim \mathcal{D}_{t-1}$ and $T_0, T_1 \sim D^n$
 8:     $f_0 =$ SOA$(S_0 \circ T_0), f_1 =$ SOA$(S_1 \circ T_1)$, $\circ$ means append
 9:     if $f_0 = f_1$ then
10:         goto 4
11:     end if
12:     Pick $x \in \{x : f_0(x) \neq f_1(x)\}$ and sample $y \sim \{+1, -1\}$ uniformly
13:     if $f_0(x) \neq y$ then
14:         output $S_0 \circ T_0 \circ (x, y)$, else $S_1 \circ T_1 \circ (x, y)$
15:     end if
16: end for
```

Distribution $\mathcal{D}_t$ From algorithm 2, we could observe that $t \cdot (n+1)$ samples are generated, where $t \cdot n$ samples are generated from $\mathcal{D}$ and t samples are generated from line 12. They are called **tournament samples**.

Proposition 1. *Suppose that $\mathcal{D}_t$ is well defined. And we apply SOA algorithm on $S \circ T$, where $S \sim \mathcal{D}_t$ and $T \sim \mathcal{D}^n$, then*

1. *Each tournament example forces a mistake with $SOA(\cdot)$*
2. *$SOA(S \circ T)$ is consistent with T*

The first item is correct when we look from line 12 to line 13 in algorithm 2. When it doesn't make a mistake on $SOA(S_0, T_0)(x)$, which means that $SOA(S_0, T_0)(x) = y$, then we must have $SOA(S_1, T_1)(x) \neq y$. Otherwise it violates the inequality of $SOA(S_0, T_0)(\cdot)$ and $SOA(S_1, T_1)(\cdot)$. The second item is also correct if $S \circ T$ is realizable by $\mathcal{H}$. We've already mentioned that if h is realizable, then it must be consistent with input. If $S \circ T$ is not realizable by $\mathcal{H}$, then we simply apply the extension rules in 3.1 to make it realizable.

Existence of Frequent Hypotheses

Lemma 2 (global stability)**.** *There exists t $\leq$ d (Ldim) and an hypothesis f s.t.*

$$\underset{S\sim\mathcal{D}_t,T\sim\mathcal{D}^n}{Pr}[SOA(S\circ T)=f]\geq 2^{-2^{d+2}}$$

Proof. Proof is given by contradiction. Suppose that $\mathcal{D}_d$ is well defined, and $\underset{S\sim\mathcal{D}_d,T\sim\mathcal{D}^n}{Pr}$[SOA($S\circ T)=f]<2^{-2^{d+2}}$. Let ρ_t be the probability that t tournament samples are consistent with f. We know that at iteration t, all previous tournament samples in S_0 and S_1 are consistent with f respectively. Therefore the probability of this prior is given by ρ_{t-1}^2. From line 3 of the algorithm to obtain $\mathcal{D}_t$, we know that the probability that $f_0=f_1$ is no larger than $2^{-2^{d+2}}\leq 8\cdot 2^{-2^{d+2}}$. So the the probability that $f_0\neq f_1$ is larger than $1-8\cdot 2^{-2^{d+2}}\geq -8\cdot 2^{-2^{d+2}}$. The probability of random sampling of y equal to the target hypothesis is equal to 1/2 since $y\in\{\pm 1\}$. Put all things altogether we have:

$$\rho_k\geq\frac{1}{2}[\rho_{t-1}^2-8\cdot 2^{-2^{d+2}}]$$

The rest of the proof is to use induction to prove $\rho_t\geq 4\cdot 2^{-2^{t+1}}$. Firstly it holds for $t=1$ since $\rho_0=1$, then we suppose that it holds for $t-1$. Then for t, we have

$$\rho_t\geq\frac{1}{2}[\rho_{t-1}^2-8\cdot 2^{-2^{d+2}}]\geq\frac{1}{2}[(4\cdot 2^{-2^t})^2-8\cdot 2^{-2^{d+2}}] \tag{1}$$

$$=8\cdot 2^{-2^{t+1}}-4\cdot 2^{-2^{d+2}}\geq 4\cdot 2^{-2^{t+1}} \tag{2}$$

The last inequality holds since $4\cdot 2^{-2^{t+1}}\geq 4\cdot 2^{-2^{d+2}}$ for all t < d, where $t,d\in\mathbb{N}^+$. Therefore $\rho_t\geq 4{\cdot}2^{-2^{t+1}}\geq 4{\cdot}2^{-2^{d+2}}\geq 2^{-2^{d+2}}$. Let's consider the case when t = d, we know from proposition 1 that it enforces a mistake for each round. We also know that for SOA, the largest number of mistakes is equal to Ldim = d, therefore, if all tournament samples in $S\sim\mathcal{D}_d$ is consistent with f, then $SOA(S)=SOA(S\circ T)=f$, therefore, $\underset{S\sim\mathcal{D}_d,T\sim\mathcal{D}^n}{Pr}(SOA(S\circ T)=f)\geq\rho_d\geq 2^{-2^{d+2}}$ which contradicts the initial proposition above and the proof ends. □

Generalization Bound

Lemma 3 (generalization)**.** *Suppose that the distribution $\mathcal{D}_t$ is realizable, if a hypothesis f satisfies:*

$$\underset{S\sim\mathcal{D}_t,T\sim\mathcal{D}^n}{Pr}(SOA(S\circ T)=f)\geq 2^{-2^{d+2}}$$

then it generalizes well: loss$_{\mathcal{D}}(f)\leq\frac{2^{d+2}}{n}$

Proof. We use the similar proof step as we introduced in the generalization bound in lemma 1. First we let $\alpha=\text{loss}_{\mathcal{D}}(h)$. Event 1 E_1: f is consistent with T $\Rightarrow$ Pr$[E_1]=(1-\alpha)^n$. Event 2 E_2: $SOA(S\circ T)=f\Rightarrow$ Pr$[E_2]\geq 2^{-2^{d+2}}$. We know if h is realizable, then it must be consistent with input T $\Rightarrow$ Pr$[E_1]\geq$ Pr$[E_2]\Rightarrow 2^{-\alpha n}\geq e^{-\alpha n}\geq(1-\alpha)^n\geq 2^{-2^{d+2}}$. Therefore, $\alpha n\leq 2^{d+2}\Rightarrow\text{loss}_{\mathcal{D}}(h)=\alpha\leq\frac{2^{d+2}}{n}$. Littlestone dimension is finite, as well as the input sample size n, therefore the upper bound is bounded, then the generalization loss is bounded on realizable distribution $\mathcal{D}$. Proof ends. □

Monte-Carlo Variant of $\mathcal{D}_t$ Notice in algorithm 2 that there's a high probability that $SOA(S_0\circ T_0)=SOA(S_1\circ T_1)$. We may generate unbounded number of samples $\Rightarrow\mathcal{D}_t$ will become undefined. To avoid this, we add the largest number of examples to be generated before line 7 in algorithm 2: N. This variant of the algorithm is called Monte-Carlo variant since in Monte-Carlo we sample finite samples to approximate solution. In order to determine such N, we could first analyze the expectation of generated samples from $\mathcal{D}_t$: $\mathbb{E}(M_t)$, and use it after to bound the probability over N.

Lemma 4 (Expected sample complexity from $\mathcal{D}_t$)**.** *Suppose $\mathcal{D}_t$ is well defined. M_t is the number of samples that are generated at round t, then*

$$\mathbb{E}[M_t]\leq 4^{t+1}\cdot n$$

Proof. There are three steps in this proof. 1) $\mathbb{E}[M_0] = 0$ since for $\mathcal{D}_0$ it generates nothing. 2) We instead prove that

$$\mathbb{E}[M_{i+1}] \leq 4\mathbb{E}[M_i] + 4n$$

where $i \in (0, t)$. 3) If 2) is true, using induction we have:

$$\mathbb{E}[M_{t+1}] \leq \left(\sum_{i=1}^{t+1} 4^i\right) n \leq \frac{4n}{3}(4^{t+1} - 1) \leq 4^{t+2} \cdot n$$

Let's denote $\mathcal{R}$ as the number of times line 7 in algorithm 2 is being executed. $\mathcal{R}$ is distributed geometrically with success prob. θ where

$$\theta = 1 - \underset{S_0,T_0,S_1,T_1}{Pr}[SOA(S_0 \circ T_0) = SOA(S_1 \circ T_1)] \tag{3}$$

$$= 1 - \sum_f \underset{S,T}{Pr}[SOA(S_0 \circ T_0) = f]^2 \tag{4}$$

$$\geq 1 - 2^{-2^{d+2}} \tag{5}$$

Last inequality holds since $\mathcal{D}_i$ is well defined so

$$\underset{S \sim \mathcal{D}_i, T \sim \mathcal{D}^n}{Pr}[\text{SOA}(S \circ T) = f] < 2^{-2^{d+2}}$$

as mentioned in proof of lemma 2. We could re-define M_{i+1} as the sum of M_{i+1}^j over j, which means 1) M_{i+1}^j are samples generated in the j-th execution if $\mathcal{R}$ > j. 2) 0 if $\mathcal{R}$ < j $\Rightarrow M_{i+1} = \sum_{j=1}^{\infty} M_{i+1}^j \Rightarrow \mathbb{E}[M_{i+1}] = \mathbb{E}[\sum_{j=1}^{\infty} M_{i+1}^j]$. For the probability of $\mathcal{R} > j$ is $(1-\theta)^{j-1}$ and in the j-th round two samples from $\mathcal{D}_i$ and $\mathcal{D}^n$ are generated respectively, so

$$\mathbb{E}[M_{i+1}^j] = (1-\theta)^{j-1}[2\mathbb{E}[M_i] + 2n]$$

We have proved that $\theta \geq 1 - 2^{-2^{d+2}} \Rightarrow 1 - 1/2 = 1/2$ Therefore,

$$\mathbb{E}[M_{i+1}] = \mathbb{E}[\sum_{j=1}^{\infty} M_{i+1}^j] = \sum_{j=1}^{\infty} \mathbb{E}[M_{i+1}^j] \tag{6}$$

$$= \sum_{j=1}^{\infty} (1-\theta)^{j-1}[2\mathbb{E}[M_i] + 2n] \tag{7}$$

$$\leq [2\mathbb{E}[M_i] + 2n] \sum_{j=1}^{\infty} \frac{1}{2}^{j-1} \tag{8}$$

$$= [2\mathbb{E}[M_i] + 2n] \cdot 2 = 4\mathbb{E}[M_i] + 4n \tag{9}$$

2) is proved. Prood ends. □

Online Learning Algorithm G We set input sample size n = $\lceil \frac{2^{d+2}}{\alpha} \rceil$. We set the sample complexity upper bound N = $2^{2^{d+2}+1} 4^{d+1} \cdot n$. We draw t = $\{0, 1, ..., d\}$ uniformly at random. Then we output $h = SOA(S \circ T)$, where $S \sim \tilde{\mathcal{D}}_t, T \sim \mathcal{D}^n$. We need to show that:

$$\Pr[G(S) = f] \geq \frac{2^{-2^{d+2}}}{d+1} \text{ and } \text{loss}_{\mathcal{D}}(f) \leq \alpha$$

where we use lemmas 2 to 4.

1. Using lemma 2, there exists $t^* \leq d$ and f^* such that

$$\underset{S \sim \mathcal{D}_{t^*}, T \sim \mathcal{D}^n}{Pr}[\text{SOA}(S \circ T) = f^*] \geq 2^{-2^{d+2}}$$

2. Using lemma 3, assume that t^* is minimal, then

$$\text{loss}_{\mathcal{D}}(f^*) \leq \frac{2^{d+2}}{n} \leq \alpha$$

3. Using lemma 4 and Markov's inequality, the probability that number of generated samples is larger than upper bound:

$$\begin{aligned}\text{Pr}[M_{t^*} > 2^{2^{d+2}+1} \cdot 4^{d+1} \cdot n] &\leq \frac{\mathbb{E}[M_{t^*}]}{2^{2^{d+2}+1} \cdot 4^{d+1} \cdot n} \qquad (10)\\ &\leq \frac{4^{t^*+1} \cdot n}{2^{2^{d+2}+1} \cdot 4^{d+1} \cdot n} \qquad (11)\\ &\leq \frac{1}{2^{2^{d+2}+1}} \qquad (12)\end{aligned}$$

4. Therefore,

$$\underset{S\sim\tilde{\mathcal{D}}_{t^*},T\sim\mathcal{D}^n}{Pr}[\text{SOA}(S \circ T) = f^*] = \underset{S\sim\mathcal{D}_{t^*},T\sim\mathcal{D}^n}{Pr}[\text{SOA}(S \circ T) = f^* \text{ and } M_{t^*} \leq 2^{2^{d+2}+1}4^{d+1} \cdot n] \tag{13}$$

$$\geq 2^{-2^{d+2}} - \frac{1}{2^{2^{d+2}+1}} \text{ omit 1 here} \tag{14}$$

$$\geq 2^{-2^d-1} \tag{15}$$

The probability of selecting optimal t^* is equal to $\frac{1}{d+1}$, then it leads to $\frac{2^{-2^d-1}}{d+1}$. Proof ends.

5 Global stable learning implies differential private learner

This part is derived from a "standard" technique. A generic private learner is used after reducing the number of hypothesis to a comparatively small number.

Theorem 8. *Let H be a concept class over data domain X. Let $G : (X \times \{\pm 1\})^m \to \{\pm 1\}^X$ be a randomized algorithm such that, for D a realizable distribution and $S \sim D^m$, there exists a hypothesis h such that $Pr[G(S) = h] \geq \eta$ and $loss_D(h) \leq \alpha/2$. Then for some*

$$n = O\bigg(\frac{m \log(1/\eta\beta\delta)}{n\varepsilon} + \frac{\log(1/\eta\beta)}{\alpha\varepsilon}\bigg)$$

there exists an (ε, δ)-differentially private algorithm $M : (X \times \{\pm 1\})^m \to \{\pm 1\}^X$ which, given n i.i.d. samples from D, produces a hypothesis $\hat{h}$ such that $loss_D(\hat{h}) \leq \alpha$ with probability at least $1 - \beta$.

Proof. This paper proves that algorithm 1 can achieve this task.

Algorithm 1. *Require: Stable learner G, Stable Histogram algorithm $Hist$, generic learner $GenericLearner$*

Step 1. Let $S_1, ..., S_k$ each consist of m i.i.d. samples from D. Run G on each batch of samples producing $h_1 = G(S_1), ..., h_k = G(S_k)$.

Step 2. Run the Stable Histogram algorithm $Hist$ on input $H = (h_1, ..., h_k)$ using privacy parameters $(\epsilon/2, \delta)$ and accuracy parameters $(\eta/8, \beta/3)$, producing a list L of frequent hypotheses.

Step 3. Let S' consist of n' i.i.d. samples from D. Run $GenericLearner(S')$ using the collection of hypotheses L with privacy parameter $(\epsilon/2, 0)$ and accuracy parameters $(\alpha/2, \beta/3)$ to output a hypothesis $\hat{h}$.

To prove a differential private algorithm is correct, we need to prove both its privacy and utility guarantees. The privacy guarantee is very easy to prove. This algorithm combined two differential private algorithm, and their privacy guarantees have been proved in previous work. Thus, by using theorems 3 and 4, we directly know that this algorithm is $(\varepsilon, \delta)-$differential private.

The only thing left is to prove that this algorithm produces a hypothesis $\hat{h}$ such that $loss_D(\hat{h}) \leq \alpha$ with probability at least $1 - \beta$. Using standard generalization arguments, we know that

$$|\text{freq}_H(h) - \underset{S\sim D^m}{\text{Pr}}[G(s) = h]| \leq \frac{\eta}{8}.$$

Then, by using the property of theorem 6, we know that the stable histogram algorithm produced a list L, which satisfies the following properties with probability larger than $1 - \beta/2$:

1. the best hypothesis h^* is in the list;
2. Estimate of every hypothesis in the list is within the true value plus an additive error not larger than $\eta/8$.

If these properties are satisfied, then the estimate of h^* is not smaller than $\frac{3}{4}\eta$. Thus, we can remove every hypothesis in the list the estimate of which is smaller than $\frac{3}{4}\eta$, and the number of remaining hypothesis is not larger than $\frac{2}{\eta}$. Thus, by using 5, we know the learner can learn a good hypothesis $\hat{h}$ such that $loss_D(\hat{h}) \leq \alpha$ with probability at least $1 - \beta/3$.

By applying the union bound, we know the probability of success is not smaller than $1 - \beta$. By simple calculation, one can show the claimed sample complexity satisfies the needs.

□

6 Extension

General loss functions As mentioned in the introduction part, the main setting for this paper is binary classification. However, a recent study Jung et al. [2020] has extended the results to multi-class classification and regression. Firstly, we could regard a regression problem as a multi-class classification problem if we map them onto a real line and perform binning the intervals. For the multi-class classification settings, the main idea is to set a tolerance parameter τ s.t. $\text{loss}_\tau[y, \hat{y}] = \mathbb{I}[|y - \hat{y}| > \tau]$. Also, the Ldim is said to be a function of τ. They have proposed an algorithm called `Color and Choose` in order to prove the therem given below:

Algorithm 2 Color and Choose

```
Input: multi-class hypothesis class $\mathcal{H} \subseteq [K]^{\mathcal{X}}$, shattered binary tree $T$, tolerance $\tau$
Choose an arbitrary hypothesis $h_0 \in \mathcal{H}$
Color each vertex $x$ of $T$ by $h_0(x) \in [K]$
Find a color $k$ such that the sub-tree $T' \subseteq T$ of color $k$ has the largest height
Let $x_0$ be the root node of $T'$
Let $x_1$ be a child of $x_0$ such that the edge $(x_0, x_1)$ is labeled as $k'$ with $|k - k'| \geq \frac{\tau}{2}$
Let $T''$ be a sub-tree of $T'$ rooted at $x_1$
Let $H' = \{h \in \mathcal{H} | h(x_0) = k'\}$
Output: $k, k', h_0, x_0, H', T''$
```

Theorem 9 (Existence of a large set of thresholds)**.** *Let $\mathcal{H} \subseteq [K]^X$ and $\mathcal{F} \subseteq [-1, 1]^X$ be multi-class and regression hypothesis classes, respectively.*

1. *If Ldim$_{2\tau}(\mathcal{H}) \geq d$, then $\mathcal{H}$ contains $\left\lfloor \frac{\log_K d}{K^2} \right\rfloor$ thresholds with a gap τ.*
2. *If fat$_\gamma(\mathcal{F}) \geq d$, then $\mathcal{F}$ contains $\left\lceil \frac{\gamma^2}{10^4} \log_{\frac{100}{\gamma}} d \right\rceil$ thresholds with a margin $\frac{\gamma}{5}$.*

where the fat-shattering dimension with scale γ, denoted as $\text{fat}_\gamma(\mathcal{F})$, is the largest d such that $\mathcal{F}$ γ-shatters a mistake tree of depth

Theorem 10 (Lower bound of the sample complexity to privately learn thresholds)**.** *Let $\mathcal{F} = \{f_i\}_{1:n} \subseteq [-1, 1]^{\mathcal{X}}$ be a set of threshold functions with a margin γ on a domain $\{x_i\}_{1:n} \subseteq \mathcal{X}$ along with bounds $u, u' \in [-1, 1]$. Suppose A is a $(\frac{\gamma}{200}, \frac{\gamma}{200})$-accurate learning algorithm for $\mathcal{F}$ with sample complexity m. If A is (ε, δ)-DP with $\varepsilon = 0.1$ and $\delta = O\left(\frac{1}{m^2 \log m}\right)$, then it can be shown that $m \geq \Omega(\log^* n)$.*

Then combining theorem 9 and theorem 10 leads to the conclusion that private learnability implies online learnability.

On the other side, the proof is similar to the proof given in this paper, where it introduces global stability as an intermediate proof step, where the probability for global stability is $O(K^{-d})$ and the number of samples for differential private learning is $O(\frac{\log(1/\eta\beta\delta)}{n\epsilon})$ to guarantee a (ϵ, δ) differential private learner with probability at least $1-\beta$. It is true and easy to extend to multi-class classification settings. However, for regression setting, authors state that for a relaxed condition, it is not possible to directly use the lemma proving from global stability to differential private learning. Authors have come up with some conditions to make it realizable.

1. Either $\mathcal{F}$ or $\mathcal{X}$ is finite.
2. The range of $\mathcal{F}$ over $\mathcal{X}$ is finite (i.e., $\{f(x) \mid f \in \mathcal{F}, x \in \mathcal{X}\} < \infty$).
3. $\mathcal{F}$ has a finite cover with respect to the sup-norm at every scale.
4. $\mathcal{F}$ has a finite sequential Pollard Pseudo-dimension.

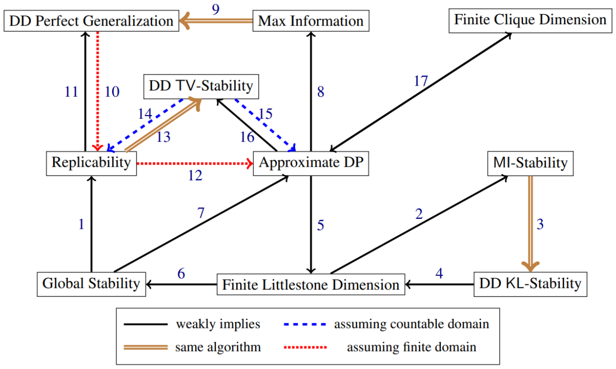
A recent study has extended these situations to a more generalized case, which solves the direction from global stability to differential private learning for the regression settings. Golowich [2021]:

Theorem 11 (Private nonparametric regression; informal version of Theorem E.1)**.** *Let $\mathcal{H}$ be a class of hypotheses $h : \mathcal{X} \to [-1, 1]$. For any $\varepsilon, \delta, \eta \in (0, 1)$, for some $n = 2^{\mathcal{O}(\mathit{sfat}_\eta(\mathcal{H}))}/\varepsilon\eta^4$, there is an (ε, δ)-differentially private algorithm which, given n i.i.d. samples from any distribution $\mathcal{Q}$ on $\mathcal{X} \times [-1, 1]$, with high probability outputs a hypothesis $\hat{h} : \mathcal{X} \to [-1, 1]$ so that*

$$\mathit{err}_{\mathcal{Q}}(\hat{h}) \leq \inf_{h \in \mathcal{H}} \mathit{err}_{\mathcal{Q}}(h) + \mathcal{O}\left(\eta \cdot \mathit{sfat}_\eta(\mathcal{H})\right).$$

Then it completes the proof from extending the binary classification problem to multi-class classification and regression problems.

Other stability metrics An open question that is left in this paper is how the global stability metric is related to other stability metrics in the learning theory field. Through a thorough investigation, additional stability includes approximate Differential Privacy Dwork et al. [2006], KL-Stability McAllester [1998], TV-Stability Kalavasis et al. [2023], f-Divergence Stability Esposito et al. [2020], and Mutual Information Stability Xu and Raginsky [2017] etc. Connections between each pair of stability mentioned are built Moran et al. [2023]: Although originally the authors proposed the stability including PAC-Bayes stability and statistical stability, given that PAC-Bayes stability is equivalent to approximate DP and statistical stability is equivalent to TV stability to statistical problems, it answers the question that the original paper proposed.



Edge key:

1. Lemma C.6
2. Lemma C.13 ([PNG22])
3. Lemma C.11
4. Lemma C.8 ([LM20])
5. Corollary 2, [ALMM19]
6. Theorem 10, [BLM20]
7. Theorem 17, [BLM20]
8. Corollary 3.13, [BGH$^+$23]
9. Lemma 3.14, [BGH$^+$23]
10. Theorem 3.17, [BGH$^+$23]
11. Theorem 3.19, [BGH$^+$23]
12. Theorem 3.1, [BGH$^+$23]
13. Theorem 1, [KKMV23]
14. Theorem 2, [KKMV23]
15. Theorem 4, [KKMV23]
16. Theorem 3, [KKMV23]
17. Theorems 2.2, 2.7 [AMSY23]

Figure 1: Equivalence between each pair of stability

References

Noga Alon, Roi Livni, Maryanthe Malliaris, and Shay Moran. Private PAC learning implies finite littlestone dimension. In Moses Charikar and Edith Cohen, editors, *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019*, pages 852–860. ACM, 2019a. doi: 10.1145/3313276.3316312. URL `https://doi.org/10.1145/3313276.3316312`.

Noga Alon, Roi Livni, Maryanthe Malliaris, and Shay Moran. Private pac learning implies finite littlestone dimension. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 852–860, 2019b.

Amos Beimel, Hai Brenner, Shiva Prasad Kasiviswanathan, and Kobbi Nissim. Bounds on the sample complexity for private learning and private data release. *Mach. Learn.*, 94(3):401–437, 2014. doi: 10.1007/S10994-013-5404-1. URL `https://doi.org/10.1007/s10994-013-5404-1`.

Amos Beimel, Kobbi Nissim, and Uri Stemmer. Characterizing the sample complexity of pure private learners. *J. Mach. Learn. Res.*, 20:146:1–146:33, 2019. URL `http://jmlr.org/papers/v20/18-269.html`.

Shai Ben-David, Dávid Pál, and Shai Shalev-Shwartz. Agnostic online learning. In *COLT*, volume 3, page 1, 2009.

Mark Bun, Kobbi Nissim, Uri Stemmer, and Salil P. Vadhan. Differentially private release and learning of threshold functions. In Venkatesan Guruswami, editor, *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 634–649. IEEE Computer Society, 2015. doi: 10.1109/FOCS.2015.45. URL `https://doi.org/10.1109/FOCS.2015.45`.

Mark Bun, Roi Livni, and Shay Moran. An equivalence between private classification and online prediction. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 389–402. IEEE, 2020.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*, pages 265–284. Springer, 2006.

Amedeo Roberto Esposito, Michael Gastpar, and Ibrahim Issa. Robust generalization via f- mutual information. In *2020 IEEE International Symposium on Information Theory (ISIT)*, pages 2723–2728. IEEE, 2020.

Vitaly Feldman and David Xiao. Sample complexity bounds on differentially private learning via communication complexity. *SIAM J. Comput.*, 44(6):1740–1764, 2015. doi: 10.1137/140991844. URL `https://doi.org/10.1137/140991844`.

Noah Golowich. Differentially private nonparametric regression under a growth condition. In *Conference on Learning Theory*, pages 2149–2192. PMLR, 2021.

Young Jung, Baekjin Kim, and Ambuj Tewari. On the equivalence between online and private learnability beyond binary classification. *Advances in neural information processing systems*, 33: 16701–16710, 2020.

Alkis Kalavasis, Amin Karbasi, Shay Moran, and Grigoris Velegkas. Statistical indistinguishability of learning algorithms. *arXiv preprint arXiv:2305.14311*, 2023.

Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SIAM Journal on Computing*, 40(3):793–826, 2011.

Aleksandra Korolova, Krishnaram Kenthapadi, Nina Mishra, and Alexandros Ntoulas. Releasing search queries and clicks privately. In Juan Quemada, Gonzalo León, Yoëlle S. Maarek, and Wolfgang Nejdl, editors, *Proceedings of the 18th International Conference on World Wide Web, WWW 2009, Madrid, Spain, April 20-24, 2009*, pages 171–180. ACM, 2009. doi: 10.1145/1526709.1526733. URL `https://doi.org/10.1145/1526709.1526733`.

Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine learning*, 2:285–318, 1988.

David A McAllester. Some pac-bayesian theorems. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 230–234, 1998.

Shay Moran, Hilla Schefler, and Jonathan Shafer. The bayesian stability zoo. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Aolin Xu and Maxim Raginsky. Information-theoretic analysis of generalization capability of learning algorithms. *Advances in Neural Information Processing Systems*, 30, 2017.