CS839: Differential Privacy and Learning April, 2026

Private and Online Learnability are Equivalent

Instructor: Gavin Brown Hongyi Liu, Zhuxiao Tang

1 Introduction

We will present the following main theorem [1].

Theorem 1.1 (Private learnability is equivalent to online learnability)**.** *Let $\mathcal{H} \subseteq \{0,1\}^X$ be a binary concept class. Then the following are equivalent:*

(1) $\mathcal{H}$ is learnable in the PAC model under approximate differential privacy;

(2) $\mathcal{H}$ is online learnable;

(3) $\mathcal{H}$ has finite Littlestone dimension.

It is well known that online learnability is characterized by the Littlestone dimension, establishing the equivalence between (2) and (3). For simplicity, we mainly focus on the implication (1) $\Rightarrow$ (3).

Theorem 1.2 (Private learning implies finite Littlestone dimension)**.** *Let H be an hypothesis class with Littlestone dimension $d \in \mathbb{N} \cup \{\infty\}$ and let A be a $\left(\frac{1}{16}, \frac{1}{16}\right)$-accurate learning algorithm for H with sample complexity m which satisfies (ε, δ)-differential privacy with $\varepsilon = 0.1$ and $\delta = O\left(\frac{1}{m^2 \log m}\right)$. Then,*

$$m \geq \Omega(\log^* d).$$

In particular any class that is privately learnable has a finite Littlestone dimension.

A central question in learning theory is to identify the structural conditions under which a hypothesis class is learnable under additional constraints such as privacy or sequential prediction. In the classical PAC framework, learnability is characterized by finiteness of the VC dimension. This main theorem provides a sharp answer: for binary classification, approximate differentially private PAC learnability coincides exactly with online learnability, and both are characterized by finiteness of the Littlestone dimension.

2 Related Work

2.1 PAC learnability without privacy

PAC learnability without privacy concern has been studied fundamentally, and it has been established that PAC learnability is characterized by VC dimension. The uniform convergence results of Vapnik and Chervonenkis provided the statistical foundation for controlling generalization through finite VC dimension [16]. Building on this perspective, Blumer et al. proved the classical equivalence that a concept class is PAC learnable in the realizable setting if and only if it has finite VC dimension [4]. Their characterization was complemented by lower bounds on the sample complexity of learning [6], and later refined by near-optimal and optimal sample complexity results [15, 9]. Together, these works establish VC dimension as the central combinatorial notion underlying PAC learnability.

2.2 Online learnability and Littlestone dimension

Introduced by [12], Littlestone dimension captures the mistake complexity of learning in an adversarial sequential setting, where examples arrive one by one and the learner must predict before observing the true label. In this framework, a concept class is online learnable precisely when it has finite Littlestone dimension, making this notion the online analogue of VC dimension in the PAC setting.

2.3 Results on pure-DP PAC learnability

Early results showed that private PAC learning is possible for many concept classes, but often with higher sample complexity than in the non-private setting [11]. In the pure-DP regime, communication-complexity lower bounds made this separation explicit, showing that finite VC dimension alone does not characterize pure-DP PAC learnability [7].

A more refined characterization of pure private learning was later given in terms of probabilistic representations, or equivalently representation dimension [3]. This showed that pure-DP PAC learnability is governed not by VC dimension alone, but by a stronger combinatorial quantity specific to privacy.

2.4 Results on approximate DP PAC Learnability

For approximate DP PAC learning, prior work established a large gap between known lower and upper bounds in terms of the Littlestone dimension d. The first lower bound showed that any private learner requires sample complexity at least $\log^* d$ (results of this paper) [2]. On the upper-bound side, an equivalence between private classification and online prediction implied a doubly exponential dependence on d [5]. This was later improved to a polynomial upper bound $\tilde{O}(d^6)$ [8], and more recently to $\tilde{O}(d^5)$ [13]. Thus, although these works strongly support Littlestone dimension as the governing parameter for approximate private learnability, a substantial gap between the best known lower and upper bounds remains.

2.5 Future work

An important direction for future work is to close the large gap between the current lower and upper bounds for approximate DP PAC learnability. At present, the best known lower bound is only $\log^*(\mathrm{Ldim}(\mathcal{H}))$, while the best upper bounds are polynomial in $\mathrm{Ldim}(\mathcal{H})$. This leaves open the possibility that the true sample complexity may be much smaller than the currently known upper bounds. A natural question is whether every class $\mathcal{H}$ can be privately learned with sample complexity

$$\mathrm{poly}\big(\mathrm{VC}(\mathcal{H}), \log^*(\mathrm{Ldim}(\mathcal{H}))\big).$$

Establishing such a bound, or ruling it out with stronger lower bounds, would significantly sharpen our understanding of approximate private learnability. More broadly, this question asks whether the dependence on Littlestone dimension can be reduced to an iterated logarithm, with the remaining complexity controlled by the classical VC dimension.

3 Preliminaries

3.1 PAC Learning

Let X be the domain set and $Y = \{0, 1\}$ be the label set. A *hypothesis* is a function $h : X \to Y$. An *example* is a pair $(x, y) \in X \times Y$. A *sample* S is a finite sequence of examples.

Definition 3.1 (Population & Empirical loss)**.** Let $\mathcal{D}$ be a distribution over $X \times \{\pm 1\}$. The population loss of a hypothesis $h : X \to \{\pm 1\}$ is defined by

$$\text{loss}_{\mathcal{D}}(h) = \Pr_{(x,y)\sim\mathcal{D}}[h(x) \neq y].$$

Let $S = ((x_i, y_i))_{i=1}^n$ be a sample. The empirical loss of h with respect to S is defined by

$$\text{loss}_S(h) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}[h(x_i) \neq y_i].$$

We can understand the definitions for population and empirical loss as:

- Population loss = true error on the underlying distribution (what we care about)
- Empirical loss = error on the observed sample (what we can compute)

The whole PAC framework is about making sure: $\text{loss}_{\mathcal{D}}(h) \approx \text{loss}_S(h)$.

Definition 3.2 (PAC learner)**.** Let $\mathcal{H} \subseteq Y^X$ be a hypothesis class. A sample S is said to be *realizable* by $\mathcal{H}$ if there is $h \in \mathcal{H}$ such that $\text{loss}_S(h) = 0$. A distribution $\mathcal{D}$ is said to be *realizable* by $\mathcal{H}$ if there is $h \in \mathcal{H}$ such that $\text{loss}_{\mathcal{D}}(h) = 0$. A learning algorithm A is a (possibly randomized) mapping taking input samples to output hypotheses. We denote by $A(S)$ the distribution over hypotheses induced by the algorithm when the input sample is S. We say that A *learns* a class $\mathcal{H}$ with α-error, $(1-\beta)$-confidence, and sample-complexity m if for every realizable distribution $\mathcal{D}$:

$$\Pr_{S\sim\mathcal{D}^m,\, h\sim A(S)}\left[\text{loss}_{\mathcal{D}}(h) > \alpha\right] \leq \beta.$$

For brevity, if A is a learning algorithm with α-error and $(1-\beta)$-confidence we will say that A is an (α, β)-accurate learner.

The two parameters α and β capture this tradeoff: α controls how close the learned hypothesis must be to optimal accuracy (approximation), while β controls the probability of failure (confidence). Thus, PAC learning formalizes the goal of learning from limited data in a way that guarantees both reliability and near-optimal performance.

3.2 Differential Privacy

Definition 3.3 ((ε, δ)-indistinguishability)**.** For $a, b, \varepsilon, \delta \in [0, 1]$, we write $a \approx_{\varepsilon,\delta} b$ if

$$a \leq e^{\varepsilon} b + \delta \quad \text{and} \quad b \leq e^{\varepsilon} a + \delta.$$

We say that two probability distributions p, q are (ε, δ)-indistinguishable if for every event E,

$$p(E) \approx_{\varepsilon,\delta} q(E).$$

Definition 3.4 (Differential Privacy)**.** A randomized algorithm

$$A : (X \times \{\pm 1\})^m \to \{\pm 1\}^X$$

is (ε, δ)-differentially private if for every two samples $S, S' \in (X \times \{\pm 1\})^m$ that disagree on a single example, the output distributions $A(S)$ and $A(S')$ are (ε, δ)-indistinguishable.

In the learning setting, $X \times \{\pm 1\}$ corresponds to labeled examples, and the algorithm outputs form a distribution over hypothesis space, i.e. $\{\pm 1\}^X$ in binary classification setting.

3.3 Littlestone Dimension and Thresholds

The Littlestone dimension measures the sequential complexity of a hypothesis class:

Definition 3.5 (Littlestone dimension)**.** Let $\mathcal{H} \subseteq \{\pm 1\}^X$ be a hypothesis class. A complete binary tree is said to be *shattered* by $\mathcal{H}$ if each internal node is labeled by a point $x \in X$, and for every root-to-leaf path with corresponding sequence of labeled examples $(x_1, y_1), \ldots, (x_t, y_t)$, there exists a hypothesis $h \in \mathcal{H}$ such that $h(x_i) = y_i$ for all $i = 1, \ldots, t$. The *Littlestone dimension* of $\mathcal{H}$, denoted $\mathrm{Ldim}(\mathcal{H})$, is the maximum depth t of a complete binary tree that is shattered by $\mathcal{H}$. If no such finite maximum exists, we say that $\mathrm{Ldim}(\mathcal{H}) = \infty$.

We say that $\mathcal{H}$ is a Littlestone class if it has finite Littlestone dimension. It captures how long an adversary can force a learner to make mistakes in an online prediction setting where examples are presented adaptively. With large Littlestone dimension, it can sustain long sequences of adaptive labelings, making learning intrinsically harder.

Definition 3.6 (Thresholds)**.** Let X be a totally ordered domain. The class of *threshold functions* over X is defined as

$$\mathrm{THRESH}_X = \{h_t : t \in X\},$$

where for each $t \in X$, the function $h_t : X \to \{\pm 1\}$ is given by

$$h_t(x) = \begin{cases} 1 & \text{if } x \geq t, \\ -1 & \text{if } x < t. \end{cases}$$

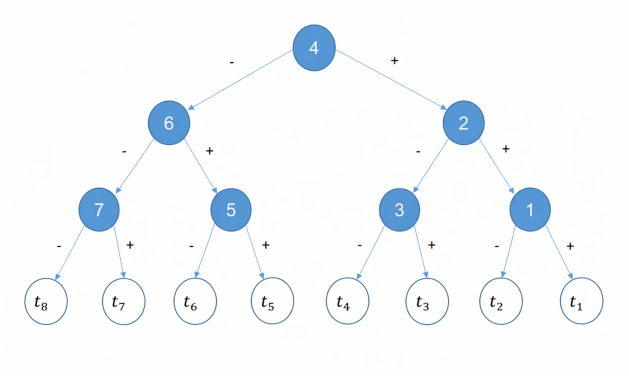

Figure 1: A tree shattered by the class $\mathcal{H} \in \{\pm 1\}$ that contains the threshold functions t_i, where $t_i(j) = +1$ if and only if $i \leq j$. In this example, $\mathrm{Ldim}(\mathcal{H}) = 3$. [1]

The following theorem, due to Shelah, establishes a fundamental quantitative relationship between Littlestone dimension and thresholds.

Theorem 3.7 (Littlestone dimension and thresholds[14, 10])**.** *Let $\mathcal{H}$ be a hypothesis class. Then:*

1. *If* $\mathrm{Ldim}(\mathcal{H}) \geq d$*, then* $\mathcal{H}$ *contains* $\lfloor \log d \rfloor$ *thresholds.*
2. *If* $\mathcal{H}$ *contains* d *thresholds, then* $\mathrm{Ldim}(\mathcal{H}) \geq \lfloor \log d \rfloor$*.*

This theorem shows that Littlestone dimension and threshold structure are tightly linked up to logarithmic factors. Intuitively, having large Littlestone dimension means that the hypothesis class can sustain long sequences of adaptively chosen, consistent labelings, which in turn implies the presence of an ordered structure within the domain. This ordered structure manifests as thresholds: hypotheses that separate the domain at different cut points. Conversely, if a class contains many thresholds, then it must be sufficiently rich to encode long sequential patterns, yielding large Littlestone dimension. Thus, thresholds serve as a canonical "hard" substructure that captures the sequential complexity measured by Littlestone dimension.

Theorem 3.8 (Thresholds are not privately learnable)**.** *Let $X \subseteq \mathbb{R}$ and let A be a $\left(\frac{1}{16}, \frac{1}{16}\right)$-accurate learning algorithm for the class of thresholds over X with sample complexity m which satisfies (ε, δ)-differential privacy with $\varepsilon = 0.1$ and $\delta = O\left(\frac{1}{m^2 \log m}\right)$. Then,*

$$m \geq \Omega(\log^* |X|).$$

In particular, the class of thresholds over an infinite X can not be learned privately.

To prove Theorem 1.2, it suffices to prove Theorem 3.8. Because, if the hypothesis class $\mathcal{H}$ has Littlestone dimension $\mathrm{Ldim}(\mathcal{H}) = d$, then by Theorem 3.7, the class $\mathcal{H}$ contains $\lfloor \log d \rfloor$ thresholds. Any private learner for $\mathcal{H}$ therefore induces a private learner for this threshold subclass. Applying Theorem 3.8 yields

$$m \geq \Omega\big(\log^*(\lfloor \log d \rfloor)\big) = \Omega(\log^* d),$$

In particular, if $\mathrm{Ldim}(\mathcal{H}) = \infty$, then $\mathcal{H}$ contains arbitrarily large threshold classes, contradicting private learnability. Therefore, any privately learnable class has a finite Littlestone dimension.

3.4 Additional Notations

A sample $S = ((x_i, y_i))_{i=1}^m$ of an even length is called *balanced* if half of its labels are $+1$'s and half are -1's, and that S is said to be *increasing* if $x_1 < x_2 < \ldots < x_m$. For $x \in X$ define $\mathrm{ord}_S(x)$ by $|\{i \mid x_i \leq x\}|$. Define $S_X = \{x_i \mid 1 \leq i \leq m\}$.

Let A be a randomized learning algorithm. It will be convenient to associate with A and S the function $A_S : X \to [0, 1]$ defined by

$$A_S(x) = \Pr_{h \sim A(S)}[h(x) = 1].$$

Intuitively, this function represents the average hypothesis outputted by A when the input sample is S.

Definition 3.9 (Iterated Logarithm)**.** The iterated logarithm $\log^{(k)}(x)$ is defined recursively by

$$\log^{(i)}(x) = \begin{cases} \log x & i = 0, \\ 1 + \log^{(i-1)}(\log x) & i > 0. \end{cases}$$

Definition 3.10 (Log-star Function)**.** The function $\log^* x$ is defined as the number of times the logarithm must be applied before the result is at most 1. Equivalently, it satisfies the recursion

$$\log^* x = \begin{cases} 0 & x \leq 1, \\ 1 + \log^*(\log x) & x > 1. \end{cases}$$

4 Private Learning Implies Finite Littlestone Dimension

In this section we prove that every class $\mathcal{H}$ which can be PAC-learned by a DP algorithm has a finite Littlestone dimension (Theorem 1.2). As shown in Section 3.3, it suffices to prove Theorem 3.8. In Section 4.1 we provide an overview of the proof. Then, in Section 4.2 we give formal statements of all corresponding lemmas and selected proofs.

4.1 Proof Overview

An common approach of proving lower bound (for example, the "No-Free-Lunch Theorem") is constructing a fixed hard distribution over inputs, and establishing that the distribution is hard to learn for any algorithm. However, in [2] the authors argue that there is no single distribution which is "hard" for all DP algorithms that learn thresholds. Instead, they construct a hard distribution in terms of every learning algorithm A. The construction utilizes the assumption that the domain is "homogeneous" w.r.t. A.

The proof consists of two parts:

1. The first part shows that for any algorithm, there is a large subset of the domain that is *homogeneous with respect to the algorithm*. This notion of homogeneity places useful restrictions on the algorithm on input samples from the homogeneous set.
2. The second part of the argument utilizes the homogeneity of $X' \subseteq X$ to derive a lower bound on the sample complexity of the algorithm in terms of $|X'|$.

Reduction to Homogeneous Sets. A subset $X' \subseteq X$ is called *homogeneous with respect to the algorithm* A if there is a list of numbers $p_0, p_1, \ldots, p_m \in [0, 1]$ such that for every increasing balanced sample S of points from X' and for every x' from $X' \setminus S_X$ with $\mathrm{ord}_S(x') = i$:

$$|A_S(x') - p_i| \leq \gamma,$$

where γ is sufficiently small. Intuitively, a set X' is homogeneous w.r.t. A if the behavior of A only depends on the position of x' relative to the sample S, not on the actual values. See Figure 2 for an illustration.

In [2], the authors prove that every algorithm has large homogeneous sets (Lemma 4.2). This is a highly technical proof using Ramsey theory, and it is independent of differential privacy. For simplicity, we will not include this proof in this report but refer readers to [2].

Lower Bound for Homogeneous Algorithms. We next assume that $X' = \{1, \ldots, k\}$ is a large homogeneous set with respect to A (with $\gamma = 0$). We will obtain a lower bound on the sample complexity of A, denoted by m, by constructing a family $\mathcal{P}$ of distributions (over a large subset of X') such that:

1. $|\mathcal{P}| \leq 2^{\tilde{O}(m^2)}$;

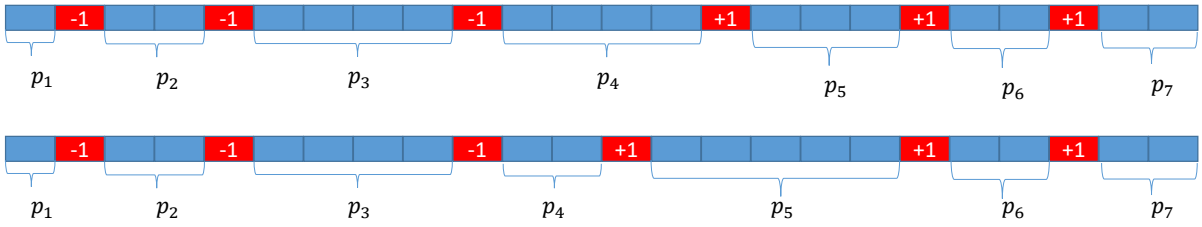

Figure 2: Depiction of two possible outputs of an algorithm over an homogeneous set, given two input samples from the set (marked in red). The number p_i denote, for a given point x, the probability that $h(x) = 1$, where $h \sim A(S)$ is the hypothesis h outputted by the algorithm on input sample S. These probabilities depends (up to a small additive error) only on the interval that x belongs to. In the figure above we changed in the input the fourth example – this only affects the interval and not the values of the p_i's (again, up to a small additive error).

2. $|\mathcal{P}| \geq \Omega(k)$.

Combining these inequalities yields a lower bound on m and concludes the proof. Detailed algebra can be found after Lemma 4.3. In the following we illustrate how to construct $\mathcal{P}$.

Let S be an increasing balanced sample of points from X'. Using the fact that A learns thresholds it is shown that for some $i_1 < i_2$ we have that $p_{i_1} \leq 1/3$ and $p_{i_2} \geq 2/3$. Thus, by a simple averaging argument there is some $i_1 \leq i \leq i_2$ such that $p_i - p_{i-1} \geq \Omega(1/m)$.

The construction of $\mathcal{P}$ proceeds as follows and is depicted in Figure 3: pick an increasing sample S such that the interval (x_{i-1}, x_{i+1}) has size $n = \Omega(k)$. For $x \in (x_{i-1}, x_{i+1})$, let S_x denote the sample obtained by replacing x_i with x in S. Each output distribution $A(S_x)$ can be seen as a distribution over the cube $\{\pm 1\}^n$ (by restricting the output hypothesis to the interval (x_{i-1}, x_{i+1}), which is of size n). This is the family of distributions $\mathcal{P} = \{P_x : x \leq n\}$. Since A is private, and by choice of the interval (x_i, x_{i+1}) we obtain that $\mathcal{P}$ has the following two properties:

1. $P_{x'}, P_{x''}$ are (ε, δ)-indistinguishable for all x', x'', and
2. Put $r = \frac{p_{i-1}+p_i}{2}$, then for all $P_x \in \mathcal{P}$

$$(\forall x' \leq n) : \Pr_{h \sim P_x}\left[h(x') = 1\right] = \begin{cases} r - \Omega(1/m) & x' < x, \\ r + \Omega(1/m) & x' > x. \end{cases}$$

Item 1 holds because of differential privacy. Item 2 holds because if $x' < x$, then x' lies in the $(i-1)$-th interval. Consequently, $\Pr[h(x') = 1] = p_{i-1}$, which implies that $r - \Pr[h(x') = 1] = \frac{p_i - p_{i-1}}{2} = \Omega(1/m)$. The case that $x' > x$ is similar.

It remains to show that $\Omega(k) \leq |\mathcal{P}| \leq 2^{\tilde{O}(m^2)}$. The lower bound follows directly from the definition of $\mathcal{P}$. The upper bound will be proved using Items 1 and 2 in Lemma 4.5.

4.2 A Lower Bound for Privately Learning Thresholds

In this section, we give full statements of all corresponding lemmas and selected proofs that are omitted in Section 4.1. Most of proofs ideas are explained in Section 4.1, except for Lemma 4.5. So

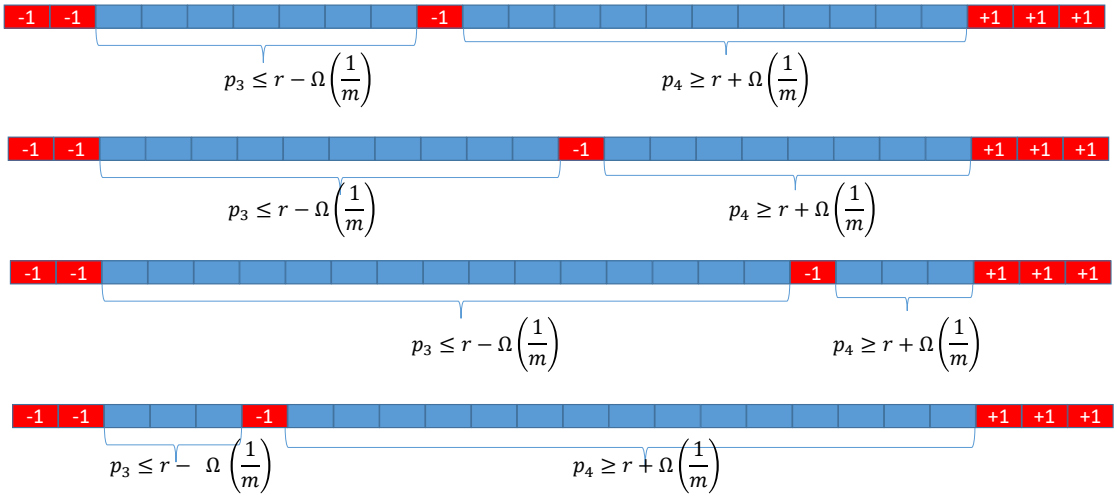

Figure 3: An illustration of the definition of the family $\mathcal{P}$. Given an homogeneous set and two consecutive intervals where there is a gap of at least $\Omega(1/m)$ between p_i and p_{i-1} (here $i = 4$). The distributions in P correspond to the different positions of the i'th example, which separates between the $(i-1)$'th and the i'th intervals.

we only include the full proof for Lemma 4.5 in this section.

Definition 4.1 (m-homogeneous set)**.** A set $X' \subseteq X$ is *m-homogeneous* with respect to a learning algorithm A if there are numbers $p_i \in [0,1]$, for $0 \le i \le m$ such that for every increasing balanced realizable sample $S \in \left(X' \times \{\pm 1\}\right)^m$ and for every $x \in X' \setminus S_X$:

$$\left|A_S(x) - p_i\right| \le \frac{1}{10^2 m},$$

where $i = \mathrm{ord}_S(x)$. The list $(p_i)_{i=0}^m$ is called the probabilities-list of X' with respect to A.

Lemma 4.2 (Every algorithm has large homogeneous sets)**.** *Let A be a (possibly randomized) algorithm that is defined over input samples of size m over a domain $X \subseteq R$ with $|X| = n$. Then, there is a set $X' \subseteq X$ that is m-homogeneous with respect to A of size*

$$|X'| \ge \frac{\log^{(m)}(n)}{2^{O(m \log m)}}.$$

Lemma 4.3 (Large homogeneous sets imply lower bounds for private learning)**.** *Let A be an $(0.1, \delta)$-differentially private algorithm with sample complexity m and $\delta \le \frac{1}{10^3 m^2 \log m}$. Let $X' = \{1, \ldots, k\}$ be m-homogeneous with respect to A. Then, if A empirically learns the class of thresholds over X' with $(1/16, 1/16)$-accuracy, then*

$$k \le 2^{O(m^2 \log^2 m)}$$

(i.e. $m \ge \Omega\left(\frac{\sqrt{\log k}}{\log \log k}\right)$).

Lemmas 4.2 and 4.3 combined with basic algebra implies that $m \ge \Omega(\log^* n)$, where n is the parameter in Lemma 4.2. Indeed, Lemma 4.2 implies the existence of an homogeneous set X' with respect to A of size $k \ge \log^{(m)}(n)/2^{O(m \log m)}$. We then restrict A to input samples from the set X',

and by relabeling the elements of X' assume that $X' = \{1, \ldots, k\}$. Lemma 4.3 then implies that $k = 2^{O(m^2 \log^2 m)}$. Together we obtain that

$$\log^{(m)}(n) \leq 2^{c \cdot m^2 \log m}$$

for some constant $c > 0$. Applying the iterated logarithm $t = \log^*(2^{c \cdot m^2 \log m}) = \log^*(m) + O(1)$ times on the inequality yields that

$$\log^{(m+t)}(n) = \log^{(m+\log^*(m)+O(1))}(n) \leq 1,$$

and therefore $\log^*(n) \leq \log^*(m) + m + O(1)$, which implies that $m \geq \Omega(\log^* n)$ as required.

To prove Lemma 4.3, we need to prove the following two lemmas:

Lemma 4.4. *Let A, X', m, k as in Lemma 4.3, and set $n = k - m$. Then there exists a family $\mathcal{P} = \{P_i : i \leq n\}$ of distributions over $\{\pm 1\}^n$ with the following properties:*

1. *Every $P_i, P_j \in \mathcal{P}$ are $(0.1, \delta)$-indistinguishable.*
2. *There exists $r \in [0, 1]$ such that for all $i, j \leq n$:*

$$\Pr_{v \sim P_i}\big[v(j) = 1\big] = \begin{cases} \leq r - \frac{1}{10m} & j < i, \\ \geq r + \frac{1}{10m} & j > i. \end{cases}$$

Lemma 4.5. *Let $\mathcal{P}, n, m, r$ as in Lemma 4.4. Then $n \leq 2^{10^3 m^2 \log^2 m}$.*

Proof. Set $T = 10^3 m^2 \log^2 m - 1$, and $D = 10^2 m^2 \log T$. We want to show that $n \leq 2^{T+1}$. Assume towards contradiction that $n > 2^{T+1}$. Consider the family of distributions $Q_i = P_i^D$ for $i = 1, \ldots, n$ (P_i^D means composite P_i for D times). By basic composition lemma, each Q_i, Q_j are $(0.1D, \delta D)$-indistinguishable.

We next define a set of mutually disjoint events E_i for $i \leq 2^T$ that are measurable with respect to each of the Q_i's. For a sequence of vectors $\mathbf{v} = (v_1, \ldots, v_D)$ in $\{\pm 1\}^n$ we let $\bar{\mathbf{v}} \in \{\pm 1\}^n$ be the threshold vector defined by

$$\bar{\mathbf{v}}(j) = \begin{cases} -1 & \frac{1}{D}\sum_{i=1}^{D} v_i(j) \leq r, \\ +1 & \frac{1}{D}\sum_{i=1}^{D} v_i(j) \geq r. \end{cases}$$

Given a point in the support of any of the Q_i's, namely a sequence $\mathbf{v} = (v_1, \ldots, v_D)$ of D vectors in $\{\pm 1\}^n$ define a mapping B according to the outcome of T steps of binary search on $\bar{\mathbf{v}}$ as follows: probe the $\frac{n}{2}$'th entry of $\bar{\mathbf{v}}$; if it is $+1$ then continue recursively with the first half of $\bar{\mathbf{v}}$. Else, continue recursively with the second half of $\bar{\mathbf{v}}$. Define the mapping $B = B(\mathbf{v})$ to be the entry that was probed at the T'th step. The events E_j correspond to the 2^T different outcomes of B. These events are mutually disjoint by the assumption that $n > 2^{T+1}$.

Notice that for any possible i in the image of B, applying the binary search on a sufficiently large i.i.d sample $\mathbf{v}$ from P_i would yield $B(\mathbf{v}) = i$ with high probability. Quantitatively, a standard application of Chernoff inequality and a union bound imply that the event $E_i = \{\mathbf{v} : B(\bar{\mathbf{v}}) = i\}$ for $\mathbf{v} \sim Q_i$, has probability at least

$$1 - T \exp\Big(-2\frac{1}{10^2 m^2} D\Big) = 1 - T \exp(-2 \log T) \geq \frac{2}{3}.$$

We claim that for all $j \leq n$, and i in the image of B:

$$Q_j(E_i) \geq \frac{1}{2}\exp(-0.1D). \tag{1}$$

This will finish the proof since the 2^T events are mutually disjoint, and therefore

$$1 \geq Q_j(\cup_i E_i) = \sum_i Q_j(E_i) \geq 2^T \cdot \frac{1}{2}e^{-0.1D} = 2^{T-1}e^{-0.1D},$$

however, $2^{T-1}e^{-0.1D} > 1$ by the choice of T, D, which is a contradiction.

Thus it remains to prove Equation (1). This follows since Q_i, Q_j are $(0.1D, D\delta)$-indistinguishable:

$$\frac{2}{3} \leq Q_i(E_i) \leq \exp(0.1D)Q_j(E_i) + D\delta,$$

and by the choice of δ, which implies that $\frac{2}{3} - D\delta \geq \frac{1}{2}$. □

References

[1] Noga Alon, Mark Bun, Roi Livni, Maryanthe Malliaris, and Shay Moran. Private and online learnability are equivalent. *J. ACM*, 69(4):28:1–28:34, 2022.

[2] Noga Alon, Roi Livni, Maryanthe Malliaris, and Shay Moran. Private pac learning implies finite littlestone dimension. STOC 2019, New York, NY, USA, 2019. Association for Computing Machinery.

[3] Amos Beimel, Kobbi Nissim, and Uri Stemmer. Characterizing the sample complexity of pure private learners. *Journal of Machine Learning Research*, 20(146):1–33, 2019.

[4] Anselm Blumer, A. Ehrenfeucht, David Haussler, and Manfred K. Warmuth. Learnability and the vapnik-chervonenkis dimension. *J. ACM*, 36(4):929–965, October 1989.

[5] Mark Bun, Roi Livni, and Shay Moran. An equivalence between private classification and online prediction. pages 389–402, 11 2020.

[6] Andrzej Ehrenfeucht, David Haussler, Michael Kearns, and Leslie Valiant. A general lower bound on the number of examples needed for learning. *Information and Computation*, 82(3):247–261, 1989.

[7] Vitaly Feldman and David Xiao. Sample complexity bounds on differentially private learning via communication complexity. *SIAM Journal on Computing*, 44(6):1740–1764, 2015.

[8] Badih Ghazi, Noah Golowich, Ravi Kumar, and Pasin Manurangsi. Sample-efficient proper pac learning with approximate differential privacy. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2021, page 183–196, New York, NY, USA, 2021. Association for Computing Machinery.

[9] Steve Hanneke. The optimal sample complexity of pac learning. *Journal of Machine Learning Research*, 17(38):1–15, 2016.

[10] W. Hodges. *A Shorter Model Theory*. Cambridge University Press, 1922.

[11] 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.

[12] Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2(4):285–318, 1988.

[13] Xin Lyu. Private learning of littlestone classes, revisited, 2025.

[14] S. Shelah. *Classification Theory and the Number of Non-isomorphic Models*. Advances in Psychology. North-Holland, 1990.

[15] Hans U. Simon. An almost optimal pac algorithm. In Peter Grünwald, Elad Hazan, and Satyen Kale, editors, *Proceedings of The 28th Conference on Learning Theory*, volume 40 of *Proceedings of Machine Learning Research*, pages 1552–1563, Paris, France, 03–06 Jul 2015. PMLR.

[16] V. N. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and Its Applications*, 16:264–280, 1971.