

Multiclass versus Binary Differentially Private PAC Learning

Mark Bun*

Marco Gaboardi†

Satchit Sivakumar‡

July 26, 2021

Abstract

We show a generic reduction from multiclass differentially private PAC learning to binary private PAC learning. We apply this transformation to a recently proposed binary private PAC learner to obtain a private multiclass learner with sample complexity that has a polynomial dependence on the multiclass Littlestone dimension and a poly-logarithmic dependence on the number of classes. This yields an exponential improvement in the dependence on both parameters over learners from previous work. Our proof extends the notion of Ψ -dimension defined in work of Ben-David et al. [BCHL95] to the online setting and explores its general properties.

Contents

1	Introduction	3
1.1	Online vs. Private Learnability	3
1.2	Techniques	4
2	Background	5
2.1	Differentially Private PAC Learning	6
2.2	Multiclass Littlestone Dimension	7
3	Main Results	7
3.1	Reduction from multiclass private PAC learning to binary private PAC learning	7
3.2	Connection between Multiclass and Binary Littlestone Dimension	10
4	Ψ-Littlestone Dimension	11
4.1	Definition	11
4.2	Properties of Ψ -Littlestone Dimension	12
5	Proof of Theorem 3.4	13
5.1	Sauer’s Lemma for Multiclass Littlestone Dimension	14
5.1.1	0-Cover Function	14
5.1.2	Statement of theorem	14
5.2	Lower Bound for 0-Cover Function	17
5.3	Putting the Pieces Together	18
6	Tightness of Theorem 1.3	19
7	Reverse Direction	21

*Boston University, mbun@bu.edu

†Boston University, gaboardi@bu.edu

‡Boston University, satchit@bu.edu

8	Pure Differential Privacy	23
A	Boosting Private Learners	28

1 Introduction

Machine learning and data analytics are increasingly deployed on sensitive information about individuals. Differential privacy [DMNS06] gives a mathematically rigorous way to enable such analyses while guaranteeing the privacy of individual information. The model of *differentially private PAC learning* [KLN⁺08] captures binary classification for sensitive data, providing a simple and broadly applicable abstraction for many machine learning procedures. Private PAC learning is now reasonably well-understood, with a host of general algorithmic techniques, lower bounds, and results for specific fundamental concept classes [BNSV15, FX14, BNS13, BNS19, ALMM19, KLM⁺20, BMNS19, KMST20].

Beyond binary classification, many problems in machine learning are better modeled as *multiclass learning* problems. Here, given a training set of examples from domain \mathcal{X} with labels from $[k] = \{0, 1, \dots, k\}$, the goal is to learn a function $h : \mathcal{X} \rightarrow [k]$ that approximately labels the data and generalizes to the underlying population from which it was drawn. Much less is presently known about differentially private multiclass learnability than is known about private binary classification, though it appears that many specific tools and techniques can be adapted one at a time. In this work, we ask: *Can we generically relate multiclass to binary learning so as to automatically transfer results from the binary setting to the multiclass setting?*

To illustrate, there is a simple reduction from a given multiclass learning problem to a sequence of binary classification problems. (This reduction was described by Ben-David et al. [BCHL95] for non-private learning, but works just as well in the private setting.) Intuitively, one can learn a multi-valued label one bit at a time. That is, to learn an unknown function $f : \mathcal{X} \rightarrow [k]$, it suffices to learn the $\lceil \log_2(k+1) \rceil$ binary functions $f_i : \mathcal{X} \rightarrow [k]$, where each f_i is the i^{th} bit of f .

Theorem 1.1 (Informal). *Let H be a concept class consisting of $[k]$ -valued functions. If all of the binary classes $H|_i = \{f_i : f \in H\}$ are privately learnable, then H is privately learnable.*

Beyond its obvious use for enabling the use of tools for binary private PAC learning on the classes $H|_i$, we show that Theorem 1.1 has strong implications for relating the private learnability of H to the combinatorial properties of H itself. Our main application of this reductive perspective is an improved sample complexity upper bound for private multiclass learning in terms of online learnability.

1.1 Online vs. Private Learnability

A recent line of work has revealed an intimate connection between differentially private learnability and learnability in Littlestone’s mistake-bound model of online learning [Lit87]. For binary classes, the latter is tightly captured by a combinatorial parameter called the Littlestone dimension; a class H is online learnable with mistake bound at most d if and only if its Littlestone dimension is at most d . The Littlestone dimension also qualitatively characterizes private learnability. If a class H has Littlestone dimension d , then every private PAC learner for H requires at least $\Omega(\log^* d)$ samples [ALMM19]. Meanwhile, Bun et al. [BLM20] showed that H is privately learnable using $2^{O(2^d)}$ samples, and Ghazi et al. [GGKM20] gave an improved algorithm using $\tilde{O}(d^6)$ samples. (Moreover, while quantitatively far apart, both the upper and lower bound are tight up to polynomial factors as functions of the Littlestone dimension alone [KLM⁺20].)

Jung et al. [JKT20] recently extended this connection from binary to multiclass learnability. They gave upper and lower bounds on the sample complexity of private multiclass learnability in terms of the *multiclass Littlestone dimension* [DSBDSS11]. Specifically, they showed that if a multi-valued class H has multiclass Littlestone dimension d , then it is privately learnable using $2^{O(k^d)}$ samples and that every private learner requires $\Omega(\log^* d)$ samples.

Jung et al.’s upper bound [JKT20] directly extended the definitions and arguments from Bun et al.’s [BLM20] earlier $2^{O(2^d)}$ -sample algorithm for the binary case. While plausible, it is currently unknown and far from obvious whether similar adaptations can be made to the improved binary algorithm of Ghazi et al. [GGKM20]. Instead of attacking this problem directly, we show that Theorem 1.1, together with additional insights relating multiclass and binary Littlestone dimensions, allows us to *generically* translate sample complexity upper bounds for private learning in terms of binary Littlestone dimension into upper bounds in terms of

multiclass Littlestone dimension. Instantiating this general translation using the algorithm of Ghazi et al. gives the following improved sample complexity upper bound.

Theorem 1.2 (Informal). *Let H be a concept class consisting of $[k]$ -valued functions and let d be the multiclass Littlestone dimension of H . Then H is privately learnable using $\tilde{O}(d^6 \log^8(k+1))$ samples.*

In addition to being conceptually simple and modular, our reduction from multiclass to binary learning means that potential future improvements for binary learning will also automatically give improvements for multiclass learning. For example, if one were able to prove that all binary classes of Littlestone dimension d are privately learnable with $O(d)$ samples, this would imply that every $[k]$ -valued class of multiclass Littlestone dimension d is privately learnable with $\tilde{O}(d \log^3(k+1))$ samples.¹

Finally, in Section 8, we study pure, i.e., $(\epsilon, 0)$ -differentially private PAC learning in the multiclass setting. Beimel et al. [BNS19] characterized the sample complexity of pure private learners in the binary setting using the notion of *probabilistic representation dimension*. We study a generalization of the representation dimension to the multiclass setting and show that it characterizes the sample complexity of pure private multiclass PAC learning up to a logarithmic term in the number of labels $k+1$. Our primary technical contribution in this section is a new and simplified proof of the relationship between representation dimension and Littlestone dimension that readily extends to the multiclass setting. This connection was previously explored by Feldman and Xiao [FX14] in the binary setting, through a connection to randomized one-way communication complexity. We instead use techniques from online learning — specifically, the experts framework and the weighted majority algorithm developed by Littlestone and Warmuth [LW94] for the binary setting and extended to the multiclass setting by Daniely et al. [DSBDSS11].

1.2 Techniques

Theorem 1.1 shows that a multi-valued class H is privately learnable if all of the binary classes $H|_i$ are privately learnable, which in turn holds as long as we can control their (binary) Littlestone dimensions. So the last remaining step in order to conclude Theorem 1.2 is to show that if H has bounded multiclass Littlestone dimension, then all of the classes $H|_i$ have bounded binary Littlestone dimension. At first glance, this may seem to follow immediately from the fact that (multiclass) Littlestone dimension characterizes (multiclass) online learnability — a mistake bounded learner for a multiclass problem is, in particular, able to learn each individual output bit of the function being learned. The problem with this intuition is that the multiclass learner is given more feedback from each example, namely the entire multi-valued class label, than a binary learner for each $H|_i$ that is only given a single bit. Nevertheless, we are still able to use combinatorial methods to show that multiclass online learnability of a class H implies online learnability of all of the binary classes $H|_i$.

Theorem 1.3. *Let H be a $[k]$ -valued concept class with multiclass Littlestone dimension d . Then every binary class $H|_i$ has Littlestone dimension at most $6d \ln(k+1)$.*

Moreover, this result is nearly tight. In Section 6, we show that for every $k, d \geq 1$ there is a $[k]$ -valued class with multiclass Littlestone dimension d such that at least one of the classes $H|_i$ has Littlestone dimension at least $\Omega(d \log(k+1))$.

Theorem 1.3 is the main technical contributions of this work. The proof adapts techniques introduced by Ben-David et al. [BCHL95] for characterizing the sample complexity of (non-private) multiclass PAC learnability. Specifically, Ben-David et al. introduced a family of combinatorial dimensions, parameterized by collections of maps Ψ and called Ψ -dimensions, associated to classes of multi-valued functions. One choice of Ψ corresponds to the “Natarajan dimension” [Nat89], which was previously known to give a lower bound on the sample complexity of multiclass learnability. Another choice corresponds to the “graph dimension” [Nat89] which was known to give an upper bound. Ben-David et al. gave conditions under which

¹The nearly cubic dependence on $\log(k+1)$ follows from the fact that the accuracy of private learners can be boosted with a sample complexity blowup that is nearly inverse linear in the target accuracy [DRV10, BCS20]. See Theorem A.1.

Ψ -dimensions for different choices of Ψ could be related to each other, concluding that the Natarajan and graph dimensions are always within an $O(\log(k+1))$ factor, and thus characterizing the sample complexity of multiclass learnability up to such a factor.

Our proof of Theorem 1.3 proceeds by extending the definition of Ψ -dimension to online learning. We show that one choice of Ψ corresponds to the multiclass Littlestone dimension, while a different choice corresponds to an upper bound on the maximum Littlestone dimension of any binary class $H|_i$. We relate the two quantities up to a logarithmic factor using a new variant of the Sauer-Shelah-Perles Lemma for the “0-cover numbers” of a class of multi-valued functions. While we were originally motivated by privacy, we believe that Theorem 1.3 and the toolkit we develop for understanding online Ψ -dimensions may be of broader interest in the study of (multiclass) online learnability.

Next, in Section 7 we prove that the multiclass Littlestone dimension of any $[k]$ -valued class H can be no more than a $\log(k+1)$ multiplicative factor larger than the maximum Littlestone dimension over the classes $H|_i$. We also show that this is tight. Hence, our results give a complete characterization of the relationship between the multiclass Littlestone dimension of a class H and the maximum Littlestone dimension over the corresponding binary classes $H|_i$.

Finally, we remark that Theorem 1.3 implies a qualitative converse to Lemma 1.1. If a multi-valued class H is privately learnable, then the lower bound of Jung et al. [JKT20] implies that H has finite multiclass Littlestone dimension. Theorem 1.3 then shows that all of the classes $H|_i$ have finite binary Littlestone dimension, which implies via sample complexity upper bounds for binary private PAC learnability [BLM20, GGKM20] that they are also privately learnable.

2 Background

Differential Privacy Differential privacy is a property of a randomized algorithm guaranteeing that the distributions obtained by running the algorithm on two datasets differing for one individual’s data are indistinguishable up to a multiplicative factor e^ϵ and an additive factor δ . Formally, it is defined as follows:

Definition 2.1 (Differential privacy, [DMNS06]). *Let $n \in \mathbb{N}$. A randomized algorithm $M : \mathcal{X}^n \rightarrow \mathcal{Y}$ is (ϵ, δ) -differentially private if for all subsets $S \subseteq \mathcal{Y}$ of the output space, and for all datasets X and X' containing n elements of the universe \mathcal{X} and differing in at most one element (we call these neighbouring datasets), we have that*

$$\Pr[M(X) \in S] \leq e^\epsilon \Pr[M(X') \in S] + \delta$$

We will also need the closely related notion of (ϵ, δ) -indistinguishability of random variables.

Definition 2.2 ((ϵ, δ) -indistinguishability). *Two random variables a_1 and a_2 defined over the same outcome space \mathcal{Y} are said to be (ϵ, δ) -indistinguishable if for all subsets $S \subseteq \mathcal{Y}$, we have that*

$$\Pr[a_1 \in S] \leq e^\epsilon \Pr[a_2 \in S] + \delta$$

and

$$\Pr[a_2 \in S] \leq e^\epsilon \Pr[a_1 \in S] + \delta$$

One useful property of differential privacy that we will use is that any output of a differentially private algorithm is closed under ‘post-processing’, that is, its cannot be made less private by applying any data-independent transformations.

Lemma 2.3 (Post-processing of differential privacy, [DMNS06]). *If $M : \mathcal{X}^n \rightarrow \mathcal{Y}$ is (ϵ, δ) -differentially private, and $\mathcal{B} : \mathcal{Y} \rightarrow \mathcal{Z}$ is any randomized function, then the algorithm $\mathcal{B} \circ M$ is (ϵ, δ) -differentially private.*

Similarly, (ϵ, δ) -indistinguishability is also preserved under post-processing.

Lemma 2.4 (Post-processing of (ϵ, δ) -indistinguishability). *If a_1 and a_2 are random variables over the same outcome space \mathcal{Y} that are (ϵ, δ) -indistinguishable, then for any possibly randomized function $\mathcal{B} : \mathcal{Y} \rightarrow \mathcal{Z}$, we have that $\mathcal{B}(a_1)$ and $\mathcal{B}(a_2)$ are (ϵ, δ) -indistinguishable.*

PAC learning. PAC learning [Val84] aims at capturing natural conditions under which an algorithm can approximately learn a hypothesis class.

Definition 2.5 (Hypothesis class). A hypothesis class H with input space \mathcal{X} and output space \mathcal{Y} (also called the label space) is a set of functions f mapping \mathcal{X} to \mathcal{Y} .

Where it is clear, we will not explicitly name the input and output spaces. We can now formally define PAC learning.

Definition 2.6 (PAC learning, [Val84]). A learning problem is defined by a hypothesis class H . For any distribution P over the input space \mathcal{X} , consider n independent draws x_1, x_2, \dots, x_n from distribution P . A labeled sample of size n is the set $\{(x_1, f(x_1)), (x_2, f(x_2)), \dots, (x_n, f(x_n))\}$ where $f \in H$. We say an algorithm A taking a labeled sample X of size n is an (α, β) -accurate PAC learner for the hypothesis class H if for all functions $f \in H$ and for all distributions P over the input space, A on being given a labeled sample of size n drawn from P and labeled by f , outputs a hypothesis $h \in H$ such that with probability greater than or equal to $1 - \beta$ over the randomness of the sample and the algorithm,

$$\Pr[h(x) \neq f(x)] \leq \alpha.$$

The definition above defines PAC learning in the *realizable* setting, where all the functions f labeling the data are in H . Two well studied settings for PAC learning are the *binary learning* case, where $\mathcal{Y} = \{0, 1\}$ and the *multiclass learning* case, where $\mathcal{Y} = [k] = \{0, 1, \dots, k\}$ for natural numbers $k > 2$. The natural notion of complexity for PAC learning is *sample complexity*.

Definition 2.7 (Sample complexity). The sample complexity $S_{H, \alpha, \beta}(A)$ of algorithm A with respect to hypothesis class H is the minimum size of the sample that the algorithm requires in order to be an (α, β) -accurate PAC learner for H . The PAC complexity of the hypothesis class H is

$$\inf_A S_{H, \alpha, \beta}(A).$$

In this work, we will be interested in *generic* learners, that work for every hypothesis class.

Definition 2.8 (Generic learners). We say that an algorithm A that additionally takes the hypothesis class as an input, is a **generic** (α, β) -accurate private PAC learner with sample complexity function $SC(H, \alpha, \beta)$, if for every hypothesis class H , it is an (α, β) -accurate private PAC learner for H with sample complexity $SC(H, \alpha, \beta)$.

2.1 Differentially Private PAC Learning

We can now define differentially private PAC learning, by putting together the constraints imposed by differential privacy and PAC learning respectively.

Definition 2.9 (Differentially private PAC learning [KLN⁺08]). An algorithm A is an (ϵ, δ) -differentially private and (α, β) -accurate private PAC learner for the hypothesis class H with sample complexity n if and only if:

1. A is an (α, β) -accurate PAC learner for the hypothesis class H with sample complexity n .
2. A is (ϵ, δ) -differentially private.

In this work, we study the complexity of private PAC learning. Our work focuses on the **multiclass realizable** setting.

2.2 Multiclass Littlestone Dimension

We recall here the definition of multiclass Littlestone dimension [DSBDSS11], which we will use extensively in this work. Unless stated otherwise, we will use the convention that the root of a tree is at depth 0. As a first step, we define a class of labeled binary trees, representing possible input-output label sequences over an input space \mathcal{X} and the label space $[k]$.

Definition 2.10 (Complete io-labeled binary tree). *A complete io-labeled binary tree of depth b with input set \mathcal{X} and output set $[k]$ consists of a complete binary tree of depth b with the following properties:*

1. *Every node of the tree other than the leaves is labeled by an example $x \in \mathcal{X}$.*
2. *The 2 edges going from any parent node to its two children are labeled by two different labels in $[k]$.*
3. *The leaf nodes of the tree are unlabeled.*

We are interested in whether the input-output labelings defined by the complete io-labeled tree can be achieved by some function in the hypothesis class; to this end, we define realizability for root-to-leaf paths.

Definition 2.11. *Given a complete io-labeled binary tree of depth b , consider a root-to-leaf path described as an ordered sequence $S = \{(x_i, y_i) \mid i \in [b]\}$, where x_i is a node label and y_i is the label of the edge between x_i and x_{i+1} , and where x_0 is the root. We say that the root-to-leaf path is realized by a function $f \in H$ if for every (x_i, y_i) in S , we have $x_i \in \mathcal{X}$ and $y_i = f(x_i)$.*

Using this definition we can now define what it means for a hypothesis class of functions to shatter a complete io-labeled binary tree, which helps to capture how expressive the hypothesis class is.

Definition 2.12 (Shattering). *We say that a complete io-labeled binary tree of depth b with label set $[k]$ is shattered by a hypothesis class H if for all 2^b root-to-leaf sequences S of the tree, there exists a function $f \in H$ that realizes S .*

Using this definition of shattering we can finally define the multiclass Littlestone dimension.

Definition 2.13 (Multiclass Littlestone dimension, [DSBDSS11]). *The **multiclass Littlestone dimension** of a hypothesis class H , denoted $MLD(H)$, is defined to be the maximum b such that there exists a complete io-labeled binary tree of depth b that is shattered by H . If no maximum exists, then we say that the multiclass Littlestone dimension of H is ∞ .*

3 Main Results

3.1 Reduction from multiclass private PAC learning to binary private PAC learning

Our first main result is a reduction from multiclass private PAC learning to binary private PAC learning. Informally, the idea is that every function f mapping examples to labels in $[k]$ can be thought of as a vector of binary functions $(f_1, \dots, f_{\log(k+1)})$. Here, each binary function predicts a bit of the binary representation of the label predicted by f . Then, we can learn these binary functions by splitting the dataset into $\log(k+1)$ parts, and using each part to learn a different f_i . We can learn the binary functions using an (ϵ, δ) -DP binary PAC learner. Then, we can combine the binary hypotheses obtained to get a hypothesis for the multiclass setting, by applying a binary to decimal transformation. This process, described in Figure 1, preserves privacy since changing a single element of the input dataset changes only one of the partitions, and we apply an (ϵ, δ) -DP learning algorithm to each partition. The binary to decimal transformation can be seen as post-processing.

Next, we formalize this idea. Given a hypothesis class H with label set $[k]$, construct the following $\log(k+1)$ hypothesis classes $H|_1, \dots, H|_{\log(k+1)}$. For every function $f \in H$, let $f_i : \mathcal{X} \rightarrow \{0, 1\}$ be the function defined such that $f_i(x)$ is the i^{th} bit of the binary expansion of $f(x)$. Let the hypothesis class $H|_i$ be defined as $\{f_i : f \in H\}$. We will call these the **binary restrictions** of H .

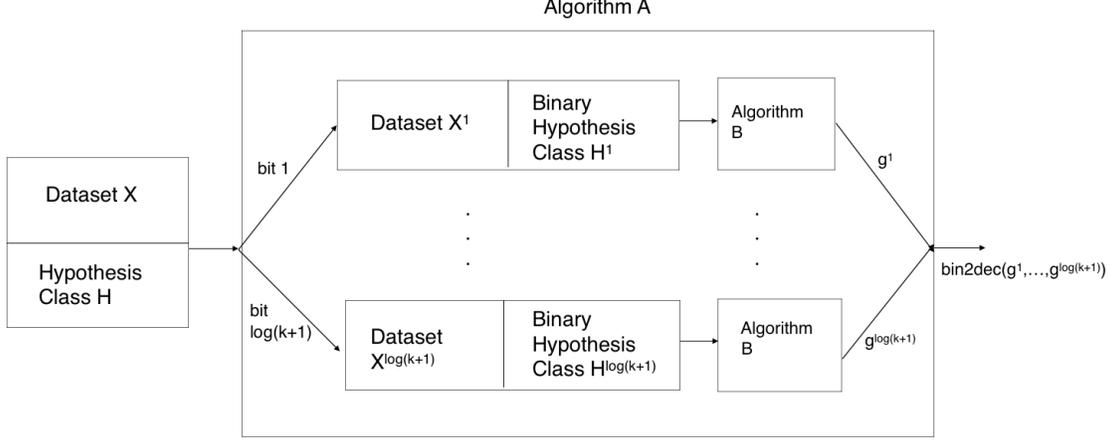


Figure 1: Algorithm A is the (ϵ, δ) -DP PAC learner for hypothesis classes with label set $[k]$. The algorithm B used as a subroutine is an (ϵ, δ) -DP PAC learner for binary hypothesis classes. bin2dec represents a binary to decimal conversion.

Theorem 3.1. *Let H be a hypothesis class with label set $[k]$ and let $H|_1, \dots, H|_{\log(k+1)}$ be its binary restrictions. Assume we have (ϵ, δ) -differentially private, (α, β) -accurate PAC learners $B^1, \dots, B^{\log(k+1)}$ for $H|_1, \dots, H|_{\log(k+1)}$ with sample complexities upper bounded by $SC_{\alpha, \beta}^1, SC_{\alpha, \beta}^2, \dots, SC_{\alpha, \beta}^{\log(k+1)}$. Then, there exists an (ϵ, δ) -differentially private, (α, β) -accurate PAC learner A for the hypothesis class H that has sample complexity upper bounded by $\sum_{i=1}^{\log(k+1)} SC_{\alpha/\log(k+1), \beta/\log(k+1)}^i$.*

Proof. For simplicity, let k be a predecessor of a power of 2. Note that if it is not, the argument below will work by replacing k with the predecessor of the closest power of 2 that is larger than k .

Fix any distribution P over \mathcal{X} and an unknown function $f \in H = \text{bin2dec}(f_1, \dots, f_{\log(k+1)})$ (bin2dec represents a binary to decimal conversion; which in this case will be an output in $[k]$) where $f_i \in H|_i$ predicts the i^{th} bit of the binary expansion of the label predicted by f .

Assuming the algorithm is given a labeled sample X of size $\sum_{i=1}^{\log(k+1)} SC_{\alpha/\log(k+1), \beta/\log(k+1)}^i$ drawn independently from P and labeled by f , split the sample into $\log(k+1)$ smaller samples $X^1, \dots, X^{\log(k+1)}$. The first sample will be of size $SC_{\alpha/\log(k+1), \beta/\log(k+1)}^1$, the second sample will be of size $SC_{\alpha/\log(k+1), \beta/\log(k+1)}^2$ and so on. For each sample X^i , replace the labels of all examples in that sample by the i^{th} bit of the binary expansion of what the label previously was. Note that this is equivalent to getting a sample of size $SC_{\alpha/\log(k+1), \beta/\log(k+1)}^i$ from distribution P that is labeled by function $f_i \in H|_i$.

For all classes $H|_i$, A runs the (ϵ, δ) -DP, $(\alpha/\log(k+1), \beta/\log(k+1))$ -accurate PAC learning algorithm B^i to learn $H|_i$ using the sample X^i . Let the hypothesis output when running the generic binary PAC learner on $H|_i$ be g_i . Then, A outputs the function $g(x) = \text{bin2dec}(g_1(x), \dots, g_{\log(k+1)}(x))$.

First, we argue that A is an (α, β) -accurate PAC learner for H .

$$\Pr[g(x) \neq f(x)] = \Pr[\exists i, g_i(x) \neq f_i(x)] \leq \sum_{i=1}^{\log(k+1)} \Pr[g_i(x) \neq f_i(x)] \quad (1)$$

where the last inequality is by a union bound.

But since g_i is the output of the $(\alpha/\log(k+1), \beta/\log(k+1))$ -accurate PAC learner B^i on $H|_i$, and we feed it a sufficient number of samples, we get that for any i , with probability $\geq 1 - \beta/\log(k+1)$,

$$\Pr[g_i(x) \neq f_i(x)] \leq \alpha/\log(k+1).$$

This means that again by a union bound, we can say that with probability $\geq 1 - \beta$,

$$\forall i, \Pr[g_i(x) \neq f_i(x)] \leq \alpha / \log(k+1) \quad (2)$$

Substituting equation 2 into equation 1, we get that with probability $1 - \beta$ over the randomness of the sample and the algorithm,

$$\Pr[g(x) \neq f(x)] \leq \sum_{i=1}^{\log(k+1)} \Pr[g_i(x) \neq f_i(x)] \leq \alpha \quad (3)$$

which means that A is an (α, β) -accurate PAC learner with sample complexity upper bounded by

$$\sum_{i=1}^{\log(k+1)} SC_{\alpha/\log(k+1), \beta/\log(k+1)}^i.$$

We now argue that A is (ϵ, δ) -DP. This will follow from the ‘parallel composition’ property of (ϵ, δ) -DP.

Claim 3.2. *Let algorithm $M : \mathcal{X}^n \rightarrow \mathcal{Y}^r$ have the following structure: it splits its input data into r disjoint partitions X^1, X^2, \dots, X^r in a data-independent way. It runs r (potentially different) (ϵ, δ) -DP algorithms $M^1 : \mathcal{X}^* \rightarrow \mathcal{Y}, \dots, M^r : \mathcal{X}^* \rightarrow \mathcal{Y}$, one on each partition. It then outputs $(M^1(X^1), M^2(X^2), \dots, M^r(X^r))$. Then, M is (ϵ, δ) -DP.*

Proof. Fix any two neighbouring datasets X and Y . Then, we want to argue that the random variable $M(X)$ is (ϵ, δ) -indistinguishable from the random variable $M(Y)$. Observe that since X and Y differ in only one element, when we partition them, all but one partition is the same. Assume without loss of generality that only the first partition is different, that is $X^1 \neq Y^1$, but $X^2 = Y^2, \dots, X^r = Y^r$. X^1 and Y^1 are neighbouring datasets since they differ in only a single element. Hence since M^1 is (ϵ, δ) -DP, we have that $M^1(X^1)$ is (ϵ, δ) -indistinguishable from $M^1(Y^1)$.

Next, consider a randomized function $f_{X,Y} : \mathcal{Y} \rightarrow \mathcal{Y}^r$ (that depends on the neighbouring dataset pair) to represent the output of M as follows: For any $q \in \mathcal{Y}$, let $f_{X,Y}(q) = (q, M^2(X^2), \dots, M^r(X^r))$. By Claim 2.4, since (ϵ, δ) -indistinguishability is preserved under post-processing, we have that $f_{X,Y}(M^1(X^1))$ is (ϵ, δ) -indistinguishable from $f_{X,Y}(M^1(Y^1))$.

But

$$f_{X,Y}(M^1(X^1)) = (M^1(X^1), M^2(X^2), \dots, M^r(X^r)) = M(X),$$

and

$$f_{X,Y}(M^1(Y^1)) = (M^1(Y^1), M^2(X^2), \dots, M^r(X^r)) = (M^1(Y^1), M^2(Y^2), \dots, M^r(Y^r)) = M(Y).$$

where the second equality follows because $X^2 = Y^2, X^3 = Y^3, \dots, X^r = Y^r$. Hence, we get that $M(X)$ and $M(Y)$ are (ϵ, δ) -indistinguishable. This argument works for any pair of databases; hence, we get that M is (ϵ, δ) -DP. \square

Note that algorithm A follows a similar structure to that described in Claim 3.2; it divides the dataset into $\log(k+1)$ partitions, runs an (ϵ, δ) -PAC learning algorithm for binary hypothesis classes on each partition and post-processes the outputs. Hence, by Claim 3.2 and by the fact that (ϵ, δ) -DP is closed under postprocessing (Claim 2.3), we get that A is (ϵ, δ) -DP. \square

Next, we recall that the sample complexity of privately learning binary hypothesis classes can be characterized by the Littlestone dimension of the hypothesis class [ALMM19, BLM20]. That is, there exists an (α, β) -accurate, (ϵ, δ) -DP PAC learning algorithm for any binary hypothesis class G with sample complexity upper and lower bounded by a function only depending on $\alpha, \beta, \epsilon, \delta$ and d where d is the Littlestone dimension of G . Using this characterization, we directly obtain the following corollary to Theorem 3.1.

Corollary 3.3. *Let H be a hypothesis class with label set $[k]$ and let $H|_1, \dots, H|_{\log(k+1)}$ be its binary restrictions. Let the Littlestone dimensions of $H|_1, \dots, H|_{\log(k+1)}$ be $d_1, \dots, d_{\log(k+1)}$. Assume we have a generic (ϵ, δ) -differentially private, (α, β) -accurate PAC learner B for binary hypothesis classes G that has sample complexity upper bounded by a function $SC_{\epsilon, \delta}(d', \alpha, \beta)$ where d' is the Littlestone dimension of G . Then, there exists an (ϵ, δ) -differentially private, (α, β) -accurate PAC learner A for H that has sample complexity upper bounded by $\sum_{i=1}^{\log(k+1)} SC_{\epsilon, \delta}(d_i, \alpha/\log(k+1), \beta/\log(k+1))$.*

Corollary 3.3 shows that the sample complexity of privately PAC learning a hypothesis class in the multiclass setting can be upper bounded by a function depending on the Littlestone dimensions of its binary restrictions. However, as described earlier, Jung et al. [JKT20] showed that the sample complexity of private multiclass PAC learning could be characterized by the multiclass Littlestone dimension. Hence, an immediate question is what the relationship between the multiclass Littlestone dimension of a class and the Littlestone dimensions of its binary restrictions is.

3.2 Connection between Multiclass and Binary Littlestone Dimension

We show that the multiclass Littlestone dimension $MLD(H)$ of a hypothesis class is intimately connected to the maximum Littlestone dimension over its binary restrictions.

Theorem 3.4. *Let H be a hypothesis class with input set \mathcal{X} and output set $[k]$. Let the multiclass Littlestone dimension of H be d . Let $H|_1, H|_2, \dots, H|_{\log(k+1)}$ be the binary restrictions of H . Let the Littlestone dimensions of $H|_1, H|_2, \dots, H|_{\log(k+1)}$ be $d_1, \dots, d_{\log(k+1)}$. Then,*

$$\max_{i=1, \dots, \log(k+1)} d_i \leq 6d \ln(k+1).$$

A similar-looking theorem relating the Natarajan dimension of a hypothesis class with the maximum VC dimension over its binary restrictions was proved in Ben-David et al. [BCHL95] using the notion of Ψ -dimension. Our proof of Theorem 3.4 is inspired by this strategy. It will proceed by defining and analyzing a notion of dimension that we call Ψ -Littlestone dimension. It will also use the 0-cover function of a hypothesis class defined in Rakhlin et al. [RST15]. The details of the proof are described in Section 5.

This theorem is tight; for all $d \geq 0$ and $k \geq 1$, there exists a hypothesis class H with label set $[k]$ and multiclass Littlestone dimension d such the maximum Littlestone dimensions over the binary restrictions of H is $O(d \ln(k+1))$. We prove this in Section 6. Additionally, the reverse direction is also true, the multiclass Littlestone dimension of any hypothesis class H with label set $[k]$ is at most a $\log(k+1)$ factor larger than the maximum Littlestone dimension over its binary restrictions (this is also tight). We prove this in Section 7.

These arguments together completely describe the relationship between the multiclass Littlestone dimension of a hypothesis class H with label set $[k]$ and the maximum Littlestone dimension over its binary restrictions.

Finally, combining Theorem 3.4 and Corollary 3.3, we can directly obtain the following corollary to Theorem 3.1.

Corollary 3.5. *Assume we have a generic (ϵ, δ) -differentially private, (α, β) -accurate PAC learner B for binary hypothesis classes G that has sample complexity upper bounded by a function $SC_{\epsilon, \delta}(d', \alpha, \beta)$ where d' is the Littlestone dimension of G . Then, there exists a generic (ϵ, δ) -differentially private, (α, β) -accurate PAC learner A for multi-valued hypothesis classes H (label set $[k]$) that has sample complexity upper bounded by $\log(k+1)SC_{\epsilon, \delta}(6d \ln(k+1), \alpha/\log(k+1), \beta/\log(k+1))$ where d is the multiclass Littlestone dimension of H .*

We now consider an application of this result. The best known sample complexity bound for (ϵ, δ) -DP binary PAC learning is achieved by a learner described in Ghazi et al. [GGKM20]. We state a slightly looser version of their result here.

Theorem 3.6 (Theorem 6.4 [GGKM20]). *Let G be any binary hypothesis class with Littlestone dimension d_L . Then, for any $\epsilon, \delta, \alpha, \beta \in [0, 1]$, for some*

$$n = O\left(\frac{d_L^6 \log^2\left(\frac{d_L}{\alpha\beta\epsilon\delta}\right)}{\epsilon\alpha^2}\right),$$

there is an (ϵ, δ) -differentially private, (α, β) -accurate PAC learning algorithm B for G with sample complexity upper bounded by n .

Now, applying the reduction described in Theorem 3.1, with this learner as a subroutine, we get the following theorem. (Instead of directly applying Theorem 3.6, we will instead first use a boosting procedure described in Appendix A.)

Theorem 3.7. *Let H be a concept class over \mathcal{X} with label set $[k]$ and multiclass Littlestone dimension d . Then, for any $\epsilon \in [0, 1/4]$, $\delta, \alpha, \beta \in [0, 1]$, for some*

$$n = O\left(\frac{d^6 (\log(k+1))^8 \log^4\left(\frac{d \log^3(k+1)}{\epsilon\delta\alpha\beta}\right)}{\epsilon\alpha}\right)$$

there is an (ϵ, δ) -differentially private, (α, β) -accurate PAC learning algorithm A for H with sample complexity upper bounded by n .

Proof. We will use the fact that the binary PAC learner from Ghazi et al. can be boosted to give a learner for binary hypothesis classes H with Littlestone dimension d' with sample complexity upper bounded by $O\left(\frac{d'^6 \log^4\left(\frac{d'}{\alpha\beta\epsilon\delta}\right)}{\epsilon\alpha}\right)$. The main difference is that the sample complexity is nearly inverse linear in the term α versus inverse quadratic. This boosting procedure is discussed in detail in Section A and the sample complexity bound we use here is derived in Corollary A.3.

Substituting into Corollary 3.5 with $d' = 6d \ln(k+1)$, $\alpha' = \alpha / \log(k+1)$, $\beta' = \beta / \log(k+1)$ gives the result. \square

4 Ψ -Littlestone Dimension

4.1 Definition

In this section, we define an online analog of the Ψ -dimension [BCHL95] that will help us prove Theorem 3.4. The main intuition is that similar to in the definition of Ψ -dimension, we can use what we term *collapsing maps* to reason about the multiclass setting while working with binary outputs. Let $\phi : [k] \rightarrow \{0, 1, *\}$ represent a function that maps labels to $\{0, 1, *\}$, which we call a *collapsing map*. We refer to a set of collapsing maps Ψ as a *family*. The definitions of labeled trees will be the only distinction from the regular definition of multiclass Littlestone dimension, and every node will have not only an example, but also a collapsing map assigned to it.

Definition 4.1 (Ψ -labeled binary tree). *A complete Ψ -labeled binary tree of depth b with label set $[k]$ and mapping set Ψ on input space \mathcal{X} consists of a complete binary tree of depth b with the following labels:*

1. *Every node of the tree other than the leaves is labeled by an example $x \in \mathcal{X}$, and a collapsing map $\phi \in \Psi$.*
2. *The left and right edges going from any parent node to its two children are labeled by 0 and 1 respectively.*
3. *The leaf nodes of the tree are unlabeled.*

A complete Ψ -uniformly labeled binary tree of depth b with label set $[k]$ and mapping set Ψ on input space \mathcal{X} is defined in the same way, with the additional property that all nodes at the same depth are labeled by the same collapsing map.

Where the input space, label space and mapping set are obvious, we will omit them and simply refer to a complete Ψ -labeled binary tree or Ψ -uniformly labeled binary tree.

Definition 4.2. Consider a root-to-leaf path in a complete Ψ -labeled binary tree described as an ordered sequence $S = ((x_0, \phi_0, y_0), \dots, (x_{b-1}, \phi_{b-1}, y_{b-1}))$, where each $x_i \in \mathcal{X}$ is an input, ϕ_i is a collapsing map, and $y_i \in \{0, 1\}$ is an edge label. We say that this path is realized by a function $f \in H$ if $y_i = \phi_i(f(x_i))$ for every triple in the ordered sequence S .

We can now define what it means for a class of functions to Ψ -shatter a complete Ψ -labeled binary tree.

Definition 4.3 (Ψ -shattering). We say that a complete Ψ -labeled binary tree of depth b with label set $[k]$ is Ψ -shattered by a hypothesis class H if for all 2^b root-to-leaf sequences S of the tree, there exists a function $f \in H$ that realizes S . Similarly, we say that a complete binary Ψ -uniformly labeled tree of depth b with label set $[k]$ is Ψ -shattered by a hypothesis class H if for all 2^b root-to-leaf sequences S of the tree, there exists a function $f \in H$ that realizes S .

Finally, we are in a position to define the Ψ -Littlestone dimension.

Definition 4.4 (Ψ -Littlestone dimension). The Ψ -**Littlestone dimension** $\Psi_{LD}(H)$ of a hypothesis class H is defined to be the maximum depth b such that there is a complete Ψ -labeled binary tree of depth b that is Ψ -shattered by H . If no maximum exists, then we say that the Ψ -Littlestone dimension of H is $d = \infty$. The **uniform Ψ -Littlestone dimension** $\Psi_{LDU}(H)$ is defined similarly (using the definition of Ψ -shattering for complete Ψ -uniformly labeled binary trees instead).

4.2 Properties of Ψ -Littlestone Dimension

In this section, we begin our investigation of the Ψ -Littlestone dimensions by discussing a few simple and useful properties. We first define three important families of collapsing maps Ψ^N , Ψ^{bin} and Ψ^B that will play an important role in our results.

Consider a collapsing map $\phi_{w,w'}$ defined by $\phi_{w,w'}(\ell) = 0$ if $\ell = w$, $\phi_{w,w'}(\ell) = 1$ if $\ell = w'$, and $\phi_{w,w'}(\ell) = *$ otherwise. Then, Ψ^N is defined to be $\{\phi_{w,w'} \mid w \neq w', w, w' \in [k]\}$. Similarly, let ϕ_i be a collapsing map that maps a label in $[k]$ to the i^{th} bit of its $\log(k+1)$ -bit binary expansion. Then, Ψ^{bin} is defined to be $\{\phi_i \mid i = 1, \dots, \log(k+1)\}$. Finally, Ψ^B is defined as the family of all collapsing maps from $[k]$ to $\{0, 1, *\}$.

We first show that the multiclass Littlestone dimension of a hypothesis class H (denoted $MLD(H)$) is equivalent to $\Psi_{LD}^N(H)$.

Lemma 4.5. For all hypothesis classes H , $\Psi_{LD}^N(H) = MLD(H)$.

Proof. Consider any complete io-labeled binary tree T of depth $MLD(H)$ that is shattered by H . Construct a complete Ψ^N -labeled binary tree T' as follows. The tree will be of the same depth as T . If in T , for a particular parent node, the two edges from a parent to a child are labeled by w, w' , then let the collapsing map labeling the parent node in T' be $\phi_{w,w'}$. The edge labeled w in T will be labeled by 0 in T' and the other edge will be labeled by 1. Also, label the nodes of T' with examples in exactly the same way as T . The leaves remain unlabeled. By the definition of shattering, for every root-to-leaf path in T , there is a function that realizes that path. This function will continue to realize the corresponding path in T' . Hence, T' is Ψ^N -shattered by H . This implies that

$$MLD(H) \leq \Psi_{LD}^N(H).$$

The other direction performs this construction in reverse: it takes a complete Ψ^N -labeled binary tree T' that is Ψ^N -shattered by H and creates a complete io-labeled binary tree T of the same depth that is shattered by H . For any node in T' , if the collapsing map assigned to that node is $\phi_{k,k'}$, the edges of that node to its children in T will be labeled k and k' respectively (the edge labeled 0 in T' will be labeled by k in T and the other edge will be labeled by k'). The nodes of T are labeled with the same examples as T' . The leaves

remain unlabeled. By a similar argument to that in the previous paragraph, we have that T is shattered by H , which means that

$$\Psi_{LD}^N(H) \leq MLD(H).$$

This proves the claim. \square

Next, we connect the Littlestone dimension of the binary restrictions of a hypothesis class H with label set $[k]$ to the Ψ^{bin} -Littlestone dimension of the class.

Claim 4.6. *Consider any hypothesis class H with label set $[k]$, and let $H|_1, H|_2, \dots, H|_{\log(k+1)}$ be the binary restrictions of H . Let the Littlestone dimension of $H|_j$ be d_j . Then,*

$$\max_j d_j \leq \Psi_{LD}^{bin} U(H) \leq \Psi_{LD}^{bin}(H).$$

Proof. The second inequality follows immediately from the fact that for any Ψ , if there exists a complete Ψ -uniformly labeled binary tree that is Ψ -shattered by H , then there exists a complete Ψ -labeled binary tree that is Ψ -shattered by H .

To prove the first inequality, fix a class $H|_i$ such that $d_i = \max_j d_j$. Consider a complete, io-labeled binary tree T of depth d_i that is shattered by $H|_i$. Then, construct the following complete Ψ^{bin} -labeled binary tree T' of the same depth d_i . For every node, label it with the same example as in tree T . Every node in T' is labeled with the collapsing map ϕ_i which maps a label to the i^{th} bit of its binary expansion. The leaves remain unlabeled. Then, we have that H Ψ^{bin} -shatters T' . Additionally, T' is of the same depth as T and all nodes at the same depth are labeled by the same collapsing map. Hence,

$$\max_j d_j \leq \Psi_{LD}^{bin} U(H).$$

\square

Finally, we relate the notions of Ψ -Littlestone dimension we have obtained with the families Ψ^N , Ψ^B and Ψ^{bin} .

Claim 4.7. *For all hypothesis class H ,*

$$\Psi_{LD}^N(H) \leq \Psi_{LD}^{bin}(H) \leq \Psi_{LD}^B(H).$$

Proof. Consider any complete Ψ^N -labeled binary tree of depth $\Psi_{LD}^N(H)$ that is Ψ^N -shattered by H . Construct a complete Ψ^{bin} -labeled binary tree T' of the same depth as follows. Label the nodes of T' with examples exactly as in T . Consider a node in T and the collapsing map $\phi_{w,w'}$ that labels the node. There is at least one bit in which the binary expansions of w and w' vary. Let this bit be the i^{th} bit. Then, label the corresponding node in T' with the collapsing map ϕ_i , which maps every label to the i^{th} bit of its binary expansion. Consider the two edges emanating from this node. If the i^{th} bit of the binary expansion of w is 0, then in T' , label the edge that was labeled 0 in T by 0 and the other by 1. Else, label the edge that was labeled 0 in T by 1 and the other by 0. Perform this transformation for every labeled node in T to obtain a corresponding labeled node in T' . The leaves of T' will remain unlabeled.

Then, T' is Ψ^{bin} -shattered by H . This gives that $\Psi_{LD}^N(H) \leq \Psi_{LD}^{bin}(H)$. The second inequality follows because $\Psi^{bin} \subseteq \Psi^B$, and so a Ψ^{bin} -labeled tree that is Ψ^{bin} -shattered by H is automatically also a Ψ^B -labeled tree that is Ψ^B -shattered by H . \square

5 Proof of Theorem 3.4

In this section, we use the concept of Ψ -Littlestone dimension to prove Theorem 3.4.

5.1 Sauer’s Lemma for Multiclass Littlestone Dimension

In this section, we will describe a version of Sauer’s Lemma that will suffice for our application. This argument is essentially due to Rakhlin et al. [RST15]. Theorem 7 in that paper states a Sauer’s lemma style upper bound for a quantity they introduce called the “0-cover function”, for hypothesis classes with bounded “sequential fat-shattering dimension.” We show that this argument applies almost verbatim for hypothesis classes with bounded multiclass Littlestone dimension.

5.1.1 0-Cover Function

We start by recalling the definition of 0-cover from Rakhlin et al.

Definition 5.1 (output-labeled trees, input-labeled trees). *A complete output-labeled binary tree of depth b with label set $[k]$ is a complete binary tree of depth b such that every node of the tree is labeled with an output in $[k]$. A complete input-labeled binary tree of depth b with input set \mathcal{X} is a complete binary tree of depth b such that every node of the tree is labeled with an input in \mathcal{X} .*

The convention we will use is that output and input-labeled binary trees have root at depth 1 (as opposed to io-labeled trees and Ψ -labeled trees, where we use the convention that root has depth 0). Consider a set V of complete output-labeled binary trees of depth b with label set $[k]$. Consider a hypothesis class H consisting of functions from input space \mathcal{X} to label set $[k]$. Fix a complete input-labeled binary tree z of depth b with input space \mathcal{X} and a complete output-labeled tree $v \in V$.

Definition 5.2. *We say that a root-to-leaf path A in z **corresponds** to a root-to-leaf path B in v if for all $1 \leq i \leq b - 1$, if node $i + 1$ in A is the left child of node i in A , then node $i + 1$ in B is the left child of node i in B and likewise for the case where node $i + 1$ is the right child of node i .*

Definition 5.3. *Let A be a root-to-leaf path in z and let the labels of the nodes in A be (x_1, \dots, x_b) where $x_i \in \mathcal{X}$. The function $f \in H$ applied to A , denoted by $f(A)$, is the sequence $(f(x_1), \dots, f(x_b))$.*

Definition 5.4 (0-cover, [RST15]). *We say that V forms a **0-cover** of hypothesis class H on tree z if, for every function $f \in H$ and every root-to-leaf path A in z , there exists a complete output-labeled tree $v \in V$, such that for the corresponding root-to-leaf path $B \in v$ with the labels of nodes in B denoted by a tuple $C \in [k]^b$ (call this the **label sequence** of B), we have that $f(A) = C$.*

Definition 5.5 (0-cover function, [RST15]). *Let $N(0, H, z)$ denote the size of the smallest 0-cover of hypothesis class H on tree z . Let $T_b^{\mathcal{X}}$ be the set of all complete input-labeled binary trees of depth b with input space \mathcal{X} . Then, the **0-cover function** $N(0, H, b)$ of hypothesis class H is defined as $\sup_{z \in T_b^{\mathcal{X}}} N(0, H, z)$.*

We use the convention that $N(0, H, 0) = 1$.

5.1.2 Statement of theorem

The following theorem is essentially Theorem 7 of Rakhlin et al. [RST15] (with multiclass Littlestone dimension in place of sequential fat shattering dimension).

Theorem 5.6. *Let hypothesis class H be a set of functions $f : \mathcal{X} \rightarrow [k]$. Let the multiclass Littlestone dimension of H be d . Then, for all natural numbers $n \geq d$, with $d \geq 0$,*

$$N(0, H, n) \leq \sum_{i=0}^d \binom{n}{i} k^i \tag{4}$$

For all natural numbers $n \geq d$, with $d > 0$, we additionally have the following:

$$N(0, H, n) \leq \sum_{i=0}^d \binom{n}{i} k^i \leq \left(\frac{ekn}{d} \right)^d. \tag{5}$$

Finally, for all $d \geq 0$, for all natural numbers $n < d$, we have $N(0, H, n) \leq (k + 1)^n$.

Proof. Firstly, observe that for all $n \geq d$, $d > 0$,

$$\begin{aligned} \sum_{i=0}^d \binom{n}{i} k^i &= \sum_{i=0}^d \binom{n}{i} k^i \left(\frac{n}{d}\right)^i \left(\frac{d}{n}\right)^i \\ &\leq \left(\frac{kn}{d}\right)^d \sum_{i=0}^d \binom{n}{i} \left(\frac{d}{n}\right)^i \\ &\leq \left(\frac{kn}{d}\right)^d \left(1 + \frac{d}{n}\right)^n \\ &\leq \left(\frac{ekn}{d}\right)^d. \end{aligned}$$

This proves the second inequality in expression 5.

The proof of the rest of the theorem will be by double induction on n and d .

First base case ($d = 0, n \geq 1$): Observe that when $d = 0$, the class H consists of only a single distinct function. Call this function f . Then, for any complete, input-labeled binary tree z of depth n on input set \mathcal{X} , create a complete, output-labeled binary tree v of depth n on output set $[k]$ as follows: for every node in z labeled by input $x \in \mathcal{X}$, label the corresponding node in v by $f(x)$. Then the set consisting of just one tree v is a 0-cover for z . Thus we have that $N(0, H, n) = 1 = \sum_{i=0}^0 \binom{n}{i} k^i$, verifying this base case.

Second base case ($0 < n \leq d$): We will prove a stronger statement; we will show that for any complete input-labeled binary tree z of depth n (for any natural number n), there is a 0-cover of hypothesis class H on z of size $(k+1)^n$. This also proves the final part of the theorem corresponding to $n < d$. We start by observing that there are $(k+1)^n$ sequences of n elements from $[k]$. For every such sequence, create a complete output-labeled binary tree v of depth n as follows: label all nodes at depth i by the i^{th} element of the sequence. In this way, we create $(k+1)^n$ different trees. This set of trees V will form a 0-cover for H on z . To see this, fix a root-to-leaf path A in z and a function $f \in H$ and consider the sequence $f(A) \in [k]^n$. Then by construction, there is a tree $v \in V$ such that every root-to-leaf path B in v has label sequence $f(A)$. This implies that V is a 0-cover of hypothesis class H on z . Thus, we have that $N(0, H, n) \leq (k+1)^n = \sum_{i=0}^n \binom{n}{i} k^i \leq \sum_{i=0}^d \binom{n}{i} k^i$ for $n \leq d$, verifying the second base case.

Inductive case: Fix a $d \geq 1, n \geq 2$ such that $n > d$ (note that the base cases handle other values of d and n). Assume that the theorem is true for all pairs of values (d', n') where $d' \leq d$ and $n' \leq n-1$. We will prove it is true for values d, n . Consider a complete, input-labeled binary tree z of depth n with input set \mathcal{X} . Let the root node of z be labeled by example $x_r \in \mathcal{X}$. Divide hypothesis class H into $k+1$ subclasses H^0, \dots, H^k as follows,

$$H^i = \{f \in H : f(x_r) = i\}.$$

That is, H^i is the subclass of functions in H that output label i on example x_r .

Claim 5.7. *There exists at most one $i \in [k]$ such that H^i has multiclass Littlestone dimension d . Every other subclass has multiclass Littlestone dimension at most $d-1$.*

Proof. Assume by way of contradiction that there are two hypothesis classes H^i and H^j that both have multiclass Littlestone dimension d . Then there are complete io-labeled, binary trees T^L and T^R of depth d with input set \mathcal{X} and output set $[k]$ that are shattered by H^i and H^j respectively. Construct a complete io-labeled binary tree T of depth $d+1$ with input set \mathcal{X} and output set $[k]$ as follows: set the root node to be x_r , and label the two edges emanating from the root by i and j respectively. Set the left sub-tree of the root to be T^L and the right sub-tree to be T^R . Then H shatters T since H^i and H^j shatter T^L and T^R respectively. However, this is a contradiction since H has multiclass Littlestone dimension d , and the shattered tree T has depth $d+1$. \square

Next, consider any hypothesis class H^i with multiclass Littlestone dimension equal to d . If no such class exists, simply choose the class H^i with maximum multiclass Littlestone dimension (note that $MLD(H^i)$ will be upper bounded by d). Let z^L and z^R be the left and right sub-trees of depth $n-1$ of the root of z . By the inductive hypothesis, there are 0-covers V^L and V^R of H^i on z^L and z^R each of size at most $\sum_{i=0}^d \binom{n-1}{i} k^i$. We will now stitch together trees from V^L and V^R to create a set of trees V that will form a 0-cover of H^i on z . Informally, we do this as follows. Every tree in V will have root labeled by i . The left sub-tree of the root will be assigned to be some tree from V^L and the right sub-tree of the root will be assigned to be some tree from V^R .

Formally, without loss of generality, let $|V^L| \geq |V^R|$. Then, there exists a surjective function ξ from V^L to V^R . For every tree $v^L \in V^L$, construct a tree in V as follows, the root will be labeled by i , the left subtree will be v^L and the right subtree will be labeled by $\xi(v^L)$. Clearly, the size of V is equal to the size of V^L , which is at most $\sum_{i=0}^d \binom{n-1}{i} k^i$. Next, we argue that the set V is a 0-cover for H^i on z .

Claim 5.8. V is a 0-cover of H^i on z .

Proof. Fix a root-to-leaf path A in z and fix a function $f \in H^i$. Let $A = (x_r, A|_{2:n})$ where $A|_{2:n}$ is the root-to-leaf path omitting the root. Consider $f(A) = (f(x_r), f(A|_{2:n})) = (i, f(A|_{2:n}))$. Note that $A|_{2:n}$ is a root-to-leaf path of either z^L or z^R . Without loss of generality, assume it is a root-to-leaf path of z^L . Hence, there exists a tree v^L in V^L such that the root-to-leaf path B in v^L corresponding to root-to-leaf path $A|_{2:n}$ in z^L has label sequence C such that $f(A|_{2:n}) = C$. (This is true since V^L is a 0-cover of H^i on z^L). By the construction of V , there is a tree v in V that has a root-to-leaf path B' (with label sequence C') corresponding to root-to-leaf path A in z such that $f(A) = (i, f(A|_{2:n})) = (i, C) = C'$. Hence, we have that V is a 0-cover of H^i on z . \square

A very similar argument can also be used to construct 0 covers of size at most $\sum_{i=0}^{d-1} \binom{n-1}{i} k^i$ for hypothesis classes H^i with multiclass Littlestone dimension at most $d-1$.

We now use the fact that the covers we constructed for H^0, \dots, H^k can be combined into a cover for H .

Claim 5.9. Let hypothesis class $G = G^1 \cup G^2 \cup \dots \cup G^l$ for some positive integer l . Let V^1, \dots, V^l be 0-covers of G^1, \dots, G^l on z . Then $V' = V^1 \cup \dots \cup V^l$ is a 0-cover of G on z .

Proof. Consider $f \in G$. Then, there exists an $i \in \{1, \dots, l\}$ such that $f \in G^i$. Thus, for every root-to-leaf path in z , there is a tree $v \in V^i$ that is consistent with $f(z)$. By the definition of V' , $v \in V'$ as well. This argument works for any function $f \in G$. Hence V' is a 0-cover of G on z . \square

Using Claim 5.9, we can construct a 0-cover V' for H on z by taking the union of the 0-covers for H^0, \dots, H^k on z . This means that the size of V' is less than or equal to the sums of sizes of the 0-covers for H^0, \dots, H^k on z . Additionally, by Claim 5.7, we have that at least k of the hypothesis classes H^i have multiclass Littlestone dimension at most $d-1$. This implies that

$$|V'| \leq k \sum_{i=0}^{d-1} \binom{n-1}{i} k^i + \sum_{i=0}^d \binom{n-1}{i} k^i.$$

Finally, we simplify this using the following claim:

Claim 5.10. For all natural numbers k, n and for all integers d such that $0 < d < n$,

$$\sum_{i=0}^d \binom{n-1}{i} k^i + k \sum_{i=0}^{d-1} \binom{n-1}{i} k^i = \sum_{i=0}^d \binom{n}{i} k^i.$$

Proof. Regrouping the terms in the sums, and using the fact that $\binom{n}{i} = \binom{n-1}{i-1} + \binom{n-1}{i}$, we get that

$$\begin{aligned}
& \sum_{i=0}^d \binom{n-1}{i} k^i + k \sum_{i=0}^{d-1} \binom{n-1}{i} k^i \\
&= \left(\binom{n-1}{d} k^d + \binom{n-1}{d-1} k \cdot k^{d-1} \right) + \left(\binom{n-1}{d-1} k^{d-1} + \binom{n-1}{d-2} k \cdot k^{d-2} \right) + \dots \\
&\quad \dots + \left(\binom{n-1}{1} k^1 + \binom{n-1}{0} k \cdot k^0 \right) + \binom{n}{0} k^0 \\
&= \binom{n}{d} k^d + \binom{n}{d-1} k^{d-1} + \dots + \binom{n}{0} k^0 \\
&= \sum_{i=0}^d \binom{n}{i} k^i.
\end{aligned}$$

□

The argument applies for any complete binary labeled tree z of depth n with input space \mathcal{X} , which means that the 0-cover number $N(0, H, n) \leq \sum_{i=0}^d \binom{n}{i} k^i$. This completes the inductive argument and proves the theorem. □

5.2 Lower Bound for 0-Cover Function

To complement the upper bound given by our variant of Sauer's Lemma, we give a lower bound showing that the 0-cover function must grow exponentially in the Ψ^B -Littlestone dimension of a class.

Lemma 5.11. *Let the Ψ^B -Littlestone dimension of hypothesis class H be d . Then,*

$$N(0, H, d) \geq 2^d.$$

Proof. The case where $d = 0$ follows trivially from the convention that $N(0, H, 0) = 1$. Hence, we consider $d > 0$.

As a reminder, we note that the convention for input-labeled and output-labeled trees is that the root is at depth 1, whereas the convention for Ψ -labeled trees is that the root is at depth 0.

Since H has Ψ^B -Littlestone dimension d , there is a complete Ψ^B -labeled binary tree T of depth d that is Ψ^B -shattered by H . Since T is Ψ^B -shattered by H , for every root-to-leaf path A of T , there is at least one function $f \in H$ that realizes that path. For each root-to-leaf path A , choose such a function and denote it by f_A .

Construct an input labeled tree T' of depth d as follows: simply remove the unlabeled leaves of T , the labels on the edges and the collapsing map assigned to each node. Note that T' has depth d by the convention used that input-labeled trees have roots at depth 1. Observe that the process of going from T to T' removes a layer of leaf nodes from T , and therefore each root-to-leaf path B in T' corresponds to two root-to-leaf paths, say B_1 and B_2 , in T . Then, map functions f_{B_1} and f_{B_2} both to B . Hence, each root-to-leaf path of T' has two functions mapped to it. See Figure 2 for a visual depiction of this. We argue that any 0-cover of T' must contain a distinct tree for each of the 2^d functions mapped to root-to-leaf paths of T' .

Fix any 2 root-to-leaf paths A_1 and A_2 in the complete Ψ^B -labeled tree T . Then there exists some node in T where the two paths diverge. Let that node be labeled by example x_{div} and collapsing map ϕ_{div} . Then we have that $\phi_{div}(f_{A_1}(x_{div})) \neq \phi_{div}(f_{A_2}(x_{div}))$ since f_{A_1} and f_{A_2} realize A_1 and A_2 respectively. However, since ϕ_{div} is a function, this implies that $f_{A_1}(x_{div}) \neq f_{A_2}(x_{div})$.

Now, consider the input-labeled tree T' . The node at which A_1 and A_2 diverge is in T' as well.

First, consider the case where it is a leaf node of T' . This means that both f_{A_1} and f_{A_2} are mapped to the same root-to-leaf path in T' . Hence, since $f_{A_1}(x_{div}) \neq f_{A_2}(x_{div})$, we cannot construct a single output-labeled tree that covers both functions since then a single leaf node would need two labels.

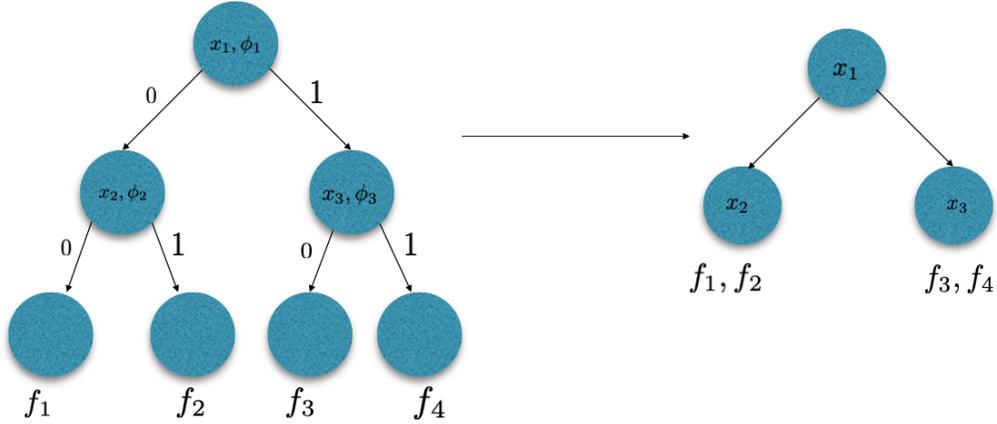


Figure 2: The tree on the left represents the Ψ^B -labeled tree T and the tree on the right represents the input-labeled tree T' created from T . Functions f_1, \dots, f_4 are each assigned to a different root-to-leaf path in T as shown, whereas in T' , two functions are assigned to each root-to-leaf path

On the other hand, if the node at which A_1 and A_2 diverge is not a leaf node of T' , then f_{A_1} and f_{A_2} are mapped to different root-to-leaf paths B_1 and B_2 in T' , which diverge at a node labeled by x_{div} . Hence, if we were trying to construct a single output-labeled tree v such that the root-to-leaf paths C_1 and C_2 in v corresponding to B_1 and B_2 in T' had label sequences s_1 and s_2 such that $s_1 = f_{A_1}(B_1)$ and $s_2 = f_{A_2}(B_2)$, then at the node at which B_1 and B_2 diverge, we would need two different labels in v , which is impossible.

This argument works for any pair A_1 and A_2 of root-to-leaf paths in T , and shows that the functions f_{A_1} and f_{A_2} require different trees in the 0-cover of T' . The number of root-to-leaf paths in T is 2^d . Hence, we have that $N(0, H, d) \geq 2^d$. \square

5.3 Putting the Pieces Together

In this section, we prove Theorem 3.4 using the techniques we have built up.

Let the Ψ^B -Littlestone dimension of H be d_B . The theorem is trivially true for $d_B = 0$, since by Lemmas 4.5 and 4.7, multiclass Littlestone dimension of H (which is d) is a lower bound for d_B .

Hence, let $d_B > 0$. By Lemmas 4.5 and 4.7, we have that $d \leq d_B$. Additionally, using Lemma 5.11 and Theorem 5.6 with $n = d_B \geq d$, we have that

$$2^{d_B} \leq N(0, H, d_B) \leq \left(\frac{ekd_B}{d} \right)^d. \quad (6)$$

We will use the fact that for all positive real numbers x, y , $\ln x \leq xy - \ln(ey)$. Fix some constant $y < \ln 2$ to be chosen later. Removing the middle man from equation 6 and simplifying, we can write the following chain of inequalities.

$$\begin{aligned}
2^{d_B} &\leq \left(\frac{ekd_B}{d}\right)^d \implies d_B \ln 2 \leq d \left(\ln\left(\frac{d_B}{d}\right) + \ln(ek)\right) \\
&\implies d_B \ln 2 \leq d \left(y \cdot \frac{d_B}{d} - \ln(ey) + \ln(ek)\right) \\
&\implies d_B(\ln 2 - y) \leq d \ln\left(\frac{k}{y}\right) \\
&\implies d_B \leq \frac{1}{\ln 2 - y} d \ln\left(\frac{k}{y}\right).
\end{aligned}$$

Setting $y = \frac{1}{5} < \ln 2$, we get that

$$d_B \leq 6d \ln(k+1).$$

Next, by Lemmas 4.6 and 4.7, we get that

$$\max_{i=1, \dots, \log(k+1)} d_i \leq \Psi_{LD}^{bin}(H) \leq d_B \leq 6d \ln(k+1).$$

This proves the theorem.

6 Tightness of Theorem 1.3

In this section, we show that Theorem 1.3 is tight up to constant factors.

Theorem 6.1. *For all integers $k \geq 1, d \geq 0$, there exists a hypothesis class H with label set $[k]$ such that the multiclass Littlestone dimension of H is d and the maximum Littlestone dimension over the binary restrictions of H is at least $\frac{d}{10} \log_2(k+1)$.*

Proof. First, we will deal with trivial cases. When $d = 0$, and $k \geq 1$, observe that any hypothesis class H with $MLD(H) = d = 0$ consists of a single function, which means each of the binary restrictions consists of a single function and the maximum Littlestone dimension over the binary restrictions is 0.

Next, when $k = 1$, and $d \geq 0$, any hypothesis class H with multiclass Littlestone dimension d has only one binary restriction, and $H|_1 = H$, which means that the multiclass Littlestone dimension of H is equal to the Littlestone dimension of $H|_1$. For $k = 2, 3, 4$, $d \geq 0$ we can consider any binary hypothesis class H , and consider it as a multi-valued hypothesis class, with no functions in the class ever outputting the labels 2, 3, 4. The multiclass Littlestone dimension in this case remains unchanged, and for $k = 2, 3$ the second binary restriction $H|_2$ is equal to H , while for $k = 4$, the third binary restriction $H|_3$ is equal to H . Hence, the maximum Littlestone dimension over the binary restrictions also remains unchanged. Additionally $\log(k+1) \leq \log(5) < 10$, so the theorem is true in these cases as well.

Thus, it suffices to consider $k \geq 5, d \geq 1$. The proof strategy will start by proving the theorem for the case where $d = 1$ and $k \geq 5$. We will then create an ‘‘amplified’’ class which will prove the theorem for arbitrary $d \geq 1$ and $k \geq 5$.

Fix $k \geq 5$. Fix $k' = k$ if k is even, and $k' = k + 1$ if k is odd. Consider an input domain $\mathcal{X} = [k'/2 - 1]$. Consider the class of threshold functions over the domain \mathcal{X} , where threshold functions g_t are parametrized by a number $t \in \mathcal{X}$ and $g_t(x) = 1[t \geq x]$ indicates whether the parameter t is at least the input x .

Then, the hypothesis class F consists of functions parametrized by $t \in \mathcal{X}$, such that $f_t(x) = (g_t(x), t)$. Note that the number of labels in the label set of F is $2(k'/2) = k'$, (the threshold function has binary outputs and $t \in [k'/2 - 1]$), and hence the output set can be encoded by $[k' - 1]$. It is immediate that the multiclass Littlestone dimension of F is 1, since any label completely specifies the function f_t . Next, observe that the binary restriction $F|_1$ corresponds to the class of thresholds over $[k'/2 - 1]$, which is known to have Littlestone dimension lower bounded by $\log(k'/4)$ [Lit87]. If k is odd, then the number of labels in the label set of F is $k + 1$; however, if k is even, then the number of labels is k . In order to make the number of

labels $k + 1$ so that the output set can be encoded by $[k]$, we add an extra label that is never used. (If k is odd, we keep the hypothesis class F unchanged). Note that adding unused labels does not change the multiclass Littlestone dimension of a class, nor does it change the maximum Littlestone dimension over its binary restrictions.

Let the maximum Littlestone dimension over the binary restrictions of F be d_1 . Hence, we get that $d_1 \geq MLD(F) \log(k/4) \geq \frac{1}{10} MLD(F) \log(k + 1)$ for $k \geq 5$.

Next, we “amplify” the gap.

Claim 6.2 (Gap amplification). *Let H be a hypothesis class with input space \mathcal{X} and output space $[k]$ such that $MLD(H) = d$ and let the maximum Littlestone dimension over its binary restrictions be at least D . Then, for all $\ell \geq 1$, there exists a hypothesis class H' with output space $[k]$ such that $MLD(H') = \ell d$ and the maximum Littlestone dimension over the binary restrictions of H' is at least ℓD .*

Proof. The hypothesis class H' will have input space $\{1, \dots, \ell\} \times \mathcal{X}$ and output space $[k]$. It is constructed as follows:

$$H' = \{h : \exists f_1, \dots, f_\ell \in H \text{ such that } \forall x \in \mathcal{X}, \forall j \in \{1, \dots, \ell\}, h((j, x)) = f_j(x)\}. \quad (7)$$

This class was used in Ghazi et al. [GGKM21] in a different context.

First, we argue that the multiclass Littlestone dimension of H' is upper bounded by ℓD . To see this, consider an online learner A for H' that runs ℓ copies of the multiclass Standard Optimal Algorithm [DSBDSS11], A_1, \dots, A_ℓ for H in parallel. Note that the multiclass Standard Optimal Algorithm is a deterministic online learning algorithm that makes at most $MLD(H)$ mistakes while online learning H . On getting an example (j, x) , A outputs the prediction of A_j on input x , and updates A_j based on the feedback from the environment. It is immediate that the worst-case number of mistakes made by A while online learning H' is upper bounded by the sum over j of the worst-case number of mistakes made by each A_j while online learning H .

By work of Daniely et al. [DSBDSS11], it is known that the multiclass Littlestone dimension of a hypothesis class lower bounds the worst-case number of mistakes of any deterministic online learner for that hypothesis class.

Combining the arguments of the above two paragraphs, we get that

$$MLD(H') \leq \ell \cdot MLD(H) = \ell d. \quad (8)$$

We postpone the proof that $MLD(H') \geq \ell d$ till later since it will follow by an argument very similar to the argument below.

Next, we argue that the maximum Littlestone dimension over the binary restrictions of H' is greater than or equal to ℓD . We will do this by relating it to the maximum Littlestone dimension over the binary restrictions of H . Let the maximum Littlestone dimension over the binary restrictions of H be achieved by the i^{th} binary restriction $H|_i$. Let its Littlestone dimension be $d_i \geq D$.

Next, we make the following observation.

$$H'|_i = \{h : \exists g_1, \dots, g_\ell \in H|_i \text{ such that } \forall x \in \mathcal{X}, \forall j \in \{1, \dots, \ell\}, h((j, x)) = g_j(x)\}. \quad (9)$$

We will represent any hypothesis $h \in H'|_i$ by ℓ hypotheses in $H|_i$. That is we will represent hypothesis h by (g_1, \dots, g_ℓ) where g_1, \dots, g_ℓ are ℓ hypotheses guaranteed by equation 9.

We now argue that there exists a complete io-labeled binary tree of depth ℓD with input set $\{1, \dots, \ell\} \times \mathcal{X}$ that is shattered by $H'|_i$. First, note that since d_i is lower bounded by D , there exists a complete io-labeled binary tree T of depth D with input set \mathcal{X} that is shattered by $H|_i$. We will derive a complete io-labeled tree T' of depth ℓD with input set $\{1, \dots, \ell\} \times \mathcal{X}$ from T .

First, create ℓ copies of T and call them T_1, \dots, T_ℓ . Remove the unlabeled leaves of $T_1, \dots, T_{\ell-1}$ (keep the unlabeled leaves of T_ℓ) and change the node labelings of T_j from x to (j, x) . Keep the edge labels as is.

Next, define concatenation as follows. Let T_1 be the top-most tree. Create tree T' by letting T_2 be both the left and right sub-tree to every leaf of T_1 . In T' , label the two edges emanating out of each leaf of T_1 by 0 and 1 respectively. We will call T' the *concatenation* of T_1 and T_2 , which we denote by $T' = T_1 \otimes T_2$.

Construct T'' as follows; let $T'' = T_1 \otimes (T_2 \otimes (\dots \otimes T_\ell))$. Observe that T'' is a complete io-labeled binary tree of depth ℓD with input set $\{1, \dots, \ell\} \times \mathcal{X}$.

Consider root-to-leaf paths represented as a sequence of tuples of the form (node label, edge label). Then, any root-to-leaf path B of T'' is of the following form.

$$B = (((1, x_1), edge_{e_1}), \dots, ((1, x_\ell), edge_{e_\ell}), ((2, x'_1), edge'_{e'_1}), \dots, ((2, x'_\ell), edge'_{e'_\ell}), \dots, ((\ell, x''_1), edge''_{e''_1}), \dots, ((\ell, x''_\ell), edge''_{e''_\ell})).$$

In this sense, B is the concatenation of ℓ root-to-leaf paths C_1, \dots, C_ℓ of T where

$$C_1 = ((x_1, edge_{e_1}), \dots, (x_\ell, edge_{e_\ell})),$$

$$C_2 = ((x'_1, edge'_{e'_1}), \dots, (x'_\ell, edge'_{e'_\ell})),$$

and so on.

Next, we argue that T'' is shattered by $H'|_i$. First, since T is shattered by $H|_i$, every root-to-leaf path C of T is realized by a function $g_C \in H|_i$. Now, fix any root-to-leaf path B of T'' . By the argument described in the preceding paragraph, B is the concatenation of ℓ root-to-leaf paths C_1, \dots, C_ℓ of T . The individual paths C_1, \dots, C_ℓ are realized by functions $g_{C_1}, \dots, g_{C_\ell} \in H|_i$ respectively. By construction, the hypothesis $h = (g_{C_1}, \dots, g_{C_\ell}) \in H'|_i$ realizes B . This argument works for every root-to-leaf path $B \in T''$, and hence we have that $H'|_i$ shatters T'' . This proves that the Littlestone dimension of $H'|_i$ is at least ℓD .

Finally, a similar argument can be used to show that the multiclass Littlestone dimension of H , i.e. $MLD(H)$ is at least ℓd . Hence, since the multiclass Littlestone dimension of H' is at least ℓd and at most ℓd , we have that it is exactly equal to ℓd . \square

Now, applying the gap amplification Claim 6.2 to hypothesis class F with multiclass Littlestone dimension 1 and maximum Littlestone dimension over its binary restrictions at least $\frac{1}{10} \log(k+1)$, setting $\ell = d$, we get a hypothesis class F' . The maximum Littlestone dimension over the binary restrictions of F' is at least $\frac{d}{10} \log(k+1)$ and the multiclass Littlestone dimension of F' is d , proving the theorem. \square

7 Reverse Direction

Theorem 7.1. *Let H be a hypothesis class with input set \mathcal{X} and output set $[k]$. Let the multiclass Littlestone dimension of H be d . Let $H|_1, H|_2, \dots, H|_{\log(k+1)}$ be the binary restrictions of H . Let the Littlestone dimensions of $H|_1, H|_2, \dots, H|_{\log(k+1)}$ be $d_1, \dots, d_{\log(k+1)}$. Then,*

$$d \leq \left[\max_{i=1, \dots, \log(k+1)} d_i \right] \log(k+1).$$

Proof. We prove the theorem by using the online learners for the binary restrictions $H|_i$ to construct an online learner for H .

The online learner A for H works as follows. It runs $\log(k+1)$ online learners $A_1, \dots, A_{\log(k+1)}$ simultaneously, one for each of the binary restrictions $H|_i$. On getting an example x , it gets the i^{th} online learner to predict the i^{th} bit of the label by sending it x . It then concatenates these predictions and applies a binary to decimal conversion to obtain a prediction for the label in $[k]$. On receiving the true label y from the environment, it updates the i^{th} online learner with the i^{th} bit of the binary expansion of the label.

If $A_1, \dots, A_{\log k}$ are all set to be the Standard Optimal Algorithm, then by the results of Littlestone [Lit87], we have that A_i makes at most d_i mistakes while online learning $H|_i$. Additionally, observe that A makes a mistake if and only if at least one of the learners $A_1, \dots, A_{\log k}$ makes a mistake. Hence, the number of mistakes made by A is upper-bounded by $\sum_{i=1}^{\log(k+1)} d_i \leq [\max_{i=1, \dots, \log(k+1)} d_i] \log(k+1)$. By work of Daniely et al. [DSBDSS11], it is known that the multiclass Littlestone dimension of a hypothesis class lower bounds the worst-case number of mistakes of any deterministic online learner for that hypothesis class. Hence, we get that $d \leq [\max_{i=1, \dots, \log(k+1)} d_i] \log(k+1)$. \square

Next, we describe a hypothesis class for which the above result is tight. The intuition is that such a hypothesis class is one where the information about the label is spread across the bits of the label as opposed to concentrated in a few bits.

Theorem 7.2. *For all integers $k \geq 1, d \geq 0$, there exists a hypothesis class H with label set $[k]$ such that the multiclass Littlestone dimension of H is $d \log(k+1)$ and the maximum Littlestone dimension over the binary restrictions of H is d .*

Proof. Let the binary restrictions $H|_1, \dots, H|_{\log(k+1)}$ all be the class of d -point functions over input space \mathbb{N} , that is for all functions $f_i \in H|_i$, there exist d distinct natural numbers x_1, \dots, x_d such that $f_i(x) = 1$ if and only if $x \in \{x_1, \dots, x_d\}$. Let H be the hypothesis class with binary restrictions $H|_1, \dots, H|_{\log(k+1)}$ as defined above. It is argued in [Lit87] that the class of 1-point functions has Littlestone dimension 1, and a straightforward extension of this argument shows that the Littlestone dimension of each of the binary restrictions $H|_1, \dots, H|_{\log(k+1)}$ is d . Hence, we are left to show that the multiclass Littlestone dimension of H is $d \log(k+1)$.

First, an application of Theorem 7.1 shows that $MLD(H) \leq d \log(k+1)$. Hence, we are left to show that $MLD(H) \geq d \log(k+1)$. To prove this, we construct an io-labeled tree with label space $[k]$ of depth $d \log(k+1)$ that is shattered by H .

To start, we observe that since $H|_i$ has multiclass Littlestone dimension d , there exists a tree T_1 of depth d that is shattered by $H|_i$. Let the set of examples labeling nodes of this tree be X_1 . Then, consider the subclass of $H|_2$ that corresponds to d -point functions over $\mathbb{N} \setminus X_1$ (that always predict 0 on points from X_1). It is straightforward to see that this class has Littlestone dimension d , and so there exists a tree T_2 of depth d that is shattered by $H|_2$, labeled only with examples from $\mathbb{N} \setminus X_1$. Let the set of examples labeling nodes of T_2 be X_2 . Then, consider the subclass of $H|_3$ that corresponds to d -point functions over $\mathbb{N} \setminus (X_1 \cup X_2)$. This class too has Littlestone dimension d , and so there exists a tree T_3 of depth d that is shattered by $H|_3$, labeled only with examples from $\mathbb{N} \setminus (X_1 \cup X_2)$. We can define $T_4, \dots, T_{\log(k+1)}$ similarly.

For all j , modify T_j , such that the edge labels of T_j are $\log(k+1)$ -bit binary strings as follows: for every edge label that is 1, replace it by $(0, \dots, 0, 1, 0, \dots, 0)$ where the only 1 is in the j^{th} position. Similarly, replace all edge labels that are 0 by the all-zero string. Remove the unlabeled leaf nodes of $T_1, \dots, T_{\log(k)}$ (keeping the leaf nodes of only a single tree $T_{\log(k+1)}$). Then, define tree T of depth $d \log(k+1)$ as follows.

$$T = T_1 \otimes (T_2 \otimes (T_3 \otimes \dots) \dots).$$

where the \otimes operator is as defined in the proof of Claim 6.2 in Section 6. Next, we argue that T is shattered by H . Consider root-to-leaf paths represented as a sequence of tuples of the form (node label, edge label). Then, any root-to-leaf path B of T is of the following form.

$$B = ((x_1, \text{edge}_1), \dots, (x_d, \text{edge}_d), (x'_1, \text{edge}'_1), \dots, (x'_d, \text{edge}'_d), \dots, (x''_1, \text{edge}''_1), \dots, (x''_d, \text{edge}''_d)).$$

In this sense, B is the concatenation of $\log(k+1)$ root-to-leaf paths $C_1, \dots, C_{\log(k+1)}$ of $T_1, T_2, \dots, T_{\log(k+1)}$ respectively where

$$C_1 = ((x_1, \text{edge}_1), \dots, (x_d, \text{edge}_d)),$$

$$C_2 = ((x'_1, \text{edge}'_1), \dots, (x'_d, \text{edge}'_d)),$$

and so on.

Consider root-to-leaf path C_1 with the edge label $(1, 0, \dots, 0)$ replaced by 1 and the all-zeros label replaced by 0. Since $H|_1$ is the class of all d -point functions, there exists a function $f_1 \in H|_1$ that predicts 0 on all inputs in the input sets $X_2, \dots, X_{\log(k+1)}$ that realizes C_1 . By a similar argument, there exists a function $f_2 \in H|_2$ that predicts 0 on all inputs in the input sets $X_1, X_3, X_4, \dots, X_{\log(k+1)}$ that realizes C_2 (with edge label $(0, 1, 0, \dots, 0)$ replaced by 1 and the all zeros label replaced by 0). This argument can be extended to all root-to-leaf paths C_j for $1 \leq j \leq \log(k+1)$. Hence, we have that the function $f = (f_1, \dots, f_{\log(k+1)}) \in H$ realizes root-to-leaf path B . Hence, H shatters T , which implies that $MLD(H) \geq d \log(k+1)$, completing the proof. \square

8 Pure Differential Privacy

In this section, we discuss multiclass private PAC learning under the constraint of $(\epsilon, 0)$ -differential privacy. We first discuss how work done by Beimel et al. [BNS19] for the binary case applies to this setting. Specifically, the *probabilistic representation dimension* characterizes the sample complexity of pure private PAC learning in the multiclass setting upto logarithmic factors in the number of labels $k + 1$. In the binary case, Feldman and Xiao [FX14] showed that for any hypothesis class H , the representation dimension $RepDim(H)$ is asymptotically lower bounded by the Littlestone dimension of the class, that is, $RepDim(H) = \Omega(LDim(H))$. Their proof was via a beautiful connection to the notion of randomized one-way communication complexity. We will show the same result in the multiclass setting, through a (in our opinion, simpler) proof using the experts framework from online learning.

First, we recall the notion of representation dimension, appropriately generalized to the multiclass setting.

Definition 8.1 (Probabilistic Representation [BNS19]). *Let \mathcal{G} be a family of hypothesis classes $\{G_1, \dots, G_r\}$ with label set $[k]$ and let P be a distribution over $\{1, \dots, r\}$. We say (P, \mathcal{G}) is an (α, β) -probabilistic representation for a hypothesis class H with label set $[k]$ and input set \mathcal{X} if for every function $f \in H$, and every distribution D over \mathcal{X} ,*

$$\Pr_P[\exists f_i \in G_i \text{ such that } \Pr_{x \sim D}[f(x) \neq f_i(x)] \leq \alpha] \geq 1 - \beta. \quad (10)$$

where the outer probability is over randomly choosing a hypothesis class $G_i \in \mathcal{G}$ according to P .

Definition 8.2 (Representation Dimension [BNS19]). *Let \mathcal{G} be a family of hypothesis classes $\{G_1, \dots, G_r\}$ with label set $[k]$. Let $size(\mathcal{G}) = \max_{G_i \in \mathcal{G}} \{\ln |G_i|\}$. Then, the Representation Dimension of a hypothesis class H with label set $[k]$ is defined as follows.*

$$RepDim(H) = \min\{size(\mathcal{G}) : \exists P \text{ such that } (P, \mathcal{G}) \text{ is a } (1/4, 1/8)\text{-probabilistic representation for } H\}. \quad (11)$$

Note that the constant $1/8$ chosen here (for the value of β) is smaller than the constant $1/4$ chosen in the definition of representation dimension in Beimel et al. However, as they point out, $1/4$ is an arbitrary choice and their results are only changed by a constant factor by changing the constant. We choose $1/8$ since it simplifies a later argument concerning the connection between the representation dimension and multiclass Littlestone dimension.

Beimel et al. proved the following two lemmas, which continue to apply in the multiclass setting via the same proofs.

Lemma 8.3 (Lemma 14, [BNS19]). *If there exists a pair (\mathcal{G}, P) that (α, β) -probabilistically represents a hypothesis class H with label set $[k]$, then for every $\epsilon > 0$, there exists an $(\epsilon, 0)$ -DP, $(6\alpha, 4\beta)$ -accurate PAC learner A for hypothesis class H that has sample complexity $O(\frac{1}{\alpha\epsilon}(RepDim(H) + \ln(1/\beta)))$.*

Lemma 8.4 (Lemma 15, [BNS19]). *For any hypothesis class H , with label set $[k]$, if there exists an $(\epsilon, 0)$ -DP, $(\alpha, 1/2)$ -accurate PAC learner A for H that has sample complexity upper bounded by m , then there exists a $(\alpha, 1/4)$ -probabilistic representation of H such that $size(H) = O(m\epsilon)$.*

The latter lemma gives a lower bound of $\Omega(RepDim(H)/\epsilon)$ on the sample complexity of pure-private PAC learning in the multiclass setting. Note that this lower bound is independent of α . Next, we add in a dependence on α exactly as in Beimel et al. Unfortunately, this direct adaptation of the proof of Beimel et al. weakens the lower bound by an additive logarithmic term in the number of labels $k + 1$.

Lemma 8.5. *For any hypothesis class H , with label set $[k]$ and input set \mathcal{X} , if there exists an $(\epsilon, 0)$ -DP, $(\alpha, 1/2)$ -accurate PAC learner A for H that has sample complexity upper bounded by m , then there exists a $(1/4, 1/8)$ -probabilistic representation of H such that $size(H) = O(m\alpha\epsilon + \ln(k + 1))$.*

Proof. Assume the existence of an $(\epsilon, 0)$ -DP, $(\alpha, 1/2)$ -accurate PAC learner A for H with input space \mathcal{X} . We assume that $m \geq 3 \ln(4)/4\alpha$ (this is without loss of generality since A can ignore part of its sample).

Fix a target function $f \in H'$ and a distribution D on input space \mathcal{X} . Let 0 be an element of \mathcal{X} . Define the following distribution \tilde{D} on input space \mathcal{X}' .

$$\Pr_{\tilde{D}}[x] = \begin{cases} 1 - 4\alpha + 4\alpha \Pr_D[x], & x = 0 \\ 4\alpha \Pr_D[x], & x \neq 0. \end{cases} \quad (12)$$

Next, define $G_D^\alpha = \{\text{function } g : \Pr_{x \sim D}(g(x) \neq f(x)) \leq \alpha\}$. Since A is $(\alpha, 1/2)$ -accurate,

$$\Pr_{X \sim \tilde{D}^m, A} [A(X) \in G_D^\alpha] \geq \frac{1}{2}.$$

In addition, by Equation 12, for every function $h : \mathcal{X}' \rightarrow [k]$ such that $\Pr_{x \sim D}[h(x) \neq f(x)] \geq \frac{1}{4}$,

$$\Pr_{x \sim \tilde{D}} [f(x) \neq h(x)] \geq 4\alpha \Pr_{x \sim D} [f(x) \neq h(x)] \geq \alpha. \quad (13)$$

Hence, $\Pr_{x \sim \tilde{D}}[f(x) \neq h(x)] \leq \alpha \implies \Pr_{x \sim D}[f(x) \neq h(x)] \leq \frac{1}{4}$, that is $G_D^\alpha \subseteq G_D^{1/4}$. Therefore, $\Pr_{X \sim \tilde{D}^m, A}[A(X) \in G_D^{1/4}] \geq \frac{1}{2}$. We call a dataset of m labeled examples *good* if the unlabeled example 0 appears at least $(1 - 8\alpha)m$ times in the dataset. Let X be a dataset constructed by taking m i.i.d samples from \tilde{D} labeled by f . By a Chernoff bound, X is good with probability at least $1 - e^{-4\alpha m/3}$. Hence, by a union bound,

$$\Pr_{A, \tilde{D}} [(A(X) \in G_D^{1/4}) \wedge (X \text{ is good})] \geq \frac{1}{2} - e^{-4\alpha m/3} \geq \frac{1}{4}.$$

Therefore, there exists a dataset X_{good} of m examples (labeled by concept f) that is good such that $\Pr_A[A(X) \in G_D^{1/4}] \geq \frac{1}{4}$ where the probability is only over the randomness of the algorithm. For $\sigma \in [k]$, let $\vec{0}_0$ represent a dataset of size m consisting only of the special element, such that every example is labeled by σ . Then, by group privacy, the fact that X_{good} is good, there exists a $\sigma \in [k]$ such that

$$\Pr_A[A(\vec{0}_\sigma) \in G_D^{1/4}] \geq e^{-8\epsilon\alpha m} \Pr_A[A(X_{\text{good}}) \in G_D^{1/4}] \geq e^{-8\epsilon\alpha m} \cdot \frac{1}{4}. \quad (14)$$

Consider a set G consisting of the outcomes of $4 \ln(8) \cdot e^{8\epsilon\alpha m}$ executions of $A(\vec{0}_0), A(\vec{0}_1), \dots, A(\vec{0}_k)$. The probability that G does not contain a hypothesis $h \in G_D^{1/4}$, is then at most $(1 - e^{-8\epsilon\alpha m} \cdot \frac{1}{4})^{4 \ln(8) \cdot e^{8\epsilon\alpha m}} \leq \frac{1}{8}$. Thus, if we let \mathcal{G} be the set of all hypothesis classes of size at most $4(k+1) \ln(8) \cdot e^{8\epsilon\alpha m}$ with label set $[k]$ and input set \mathcal{X}' , and set the distribution P to be the distribution on \mathcal{G} induced by $A(\vec{0}_0)$, then (P, \mathcal{G}) is a $(1/4, 1/8)$ -probabilistic representation of H , and $\text{size}(\mathcal{G}) = \max_{G_i \in \mathcal{G}} \{\ln |G_i|\} = \ln(4(k+1) \ln(8)) + 8m\epsilon\alpha = O(m\alpha\epsilon + \ln(k+1))$. \square

The above lemma gives us that the sample complexity of any $(\epsilon, 0)$ -DP, $(\alpha, 1/2)$ -accurate learning algorithm for a hypothesis class H with label set $[k]$ is $m = \Omega\left(\frac{\text{RepDim}(H) - \ln(k+1)}{\alpha\epsilon}\right)$. Combined with Lemma 8.3, this proves that the representation dimension captures the sample complexity of pure DP PAC learning in the multiclass setting up to logarithmic factors in $k+1$.

Finally, we prove that the representation dimension of a hypothesis class H is asymptotically lower bounded by the multiclass Littlestone dimension of H . To prove this, we will use the following lemmas from Daniely et al. [DSBDSS11].

Lemma 8.6 (Theorem 5.1, [DSBDSS11]). *Let H be a hypothesis class with label set $[k]$, such that $\text{MLD}(H) = d$. Then, for every online learning algorithm A for H (in the realizable setting), there exists a sequence of d examples, such that A makes at least $d/2$ mistakes in expectation on this sequence.*

Our proof makes use of the experts framework in online learning. In this framework, at each time step t , before the online learner chooses its prediction, it gets N pieces of advice from experts with opinions about what the correct prediction should be. There is a classical algorithm called the Weighted Majority Algorithm [LW94, DSBDS11] that achieves the following guarantee in this framework for every sequence of length T .

Lemma 8.7 (Page 20, [DSBDS11]). *Consider N experts. Let A be the Weighted Majority Algorithm. Then, for all sequences S of length T , labeled by an unknown hypothesis in a hypothesis class H with label set $[k]$, if the number of mistakes made by the i^{th} expert on S is $L_{i,T}$ and the number of mistakes made by A on S is $L_{A,T}$, then,*

$$\mathbb{E}_A[L_{A,T} - \min_{i \in [N]} \{L_{i,T}\}] \leq \sqrt{\frac{1}{2} \ln(N)T}. \quad (15)$$

Next, we prove the main result.

Theorem 8.8. *For all $k \in \mathbb{N}$, for any hypothesis class H with label set $[k]$, $\text{RepDim}(H) = \Omega(\text{MLD}(H))$.*

Proof. We can assume $\text{MLD}(H) > 0$, since when $\text{MLD}(H) = 0$, the result is vacuously true. By the definition of representation dimension, there exists a $(1/4, 1/8)$ -probabilistic representation (P, \mathcal{G}) for hypothesis class H where $\text{size}(\mathcal{G}) = \text{RepDim}(H)$.

Now, consider the following online learner A for H . It first samples a hypothesis class $G_i \in \mathcal{G}$ from P . It then runs the Weighted Majority Algorithm with the set of experts being the functions of hypothesis class G_i in order to make predictions at every timestep.

Consider the worst case sequence S for A of length $T = \text{MLD}(H)$ guaranteed by Lemma 8.6. Let D be the empirical distribution corresponding to this sequence. Fix an unknown hypothesis $f \in H$ labeling this sequence. By the definition of probabilistic representation, if a hypothesis class G_i is sampled from P , with probability at least $7/8$, there exists a hypothesis $g \in G_i$ such that $\Pr_{x \sim D}[g(x) \neq f(x)] \leq 1/4$. This implies that the number of mistakes that g makes on S labeled by f is at most $\text{MLD}(H)/4$.

Let L_A be a random variable denoting the number of mistakes that A makes on S . Define the event E as follows: "Class G_i sampled from P is such that for all distributions D' and for all functions $f' \in H$, there exists a function $g \in G_i$ such that $\Pr_{x \sim D'}[g(x) \neq f'(x)] \leq 1/4$." Then, $\Pr[E] \geq 7/8$. Using the law of total expectation, and the fact that the maximum number of mistakes is the length of the sequence S , we can then write that

$$\mathbb{E}_A[L_A] = \mathbb{E}_A[L_A | E] \Pr[E] + \mathbb{E}_A[L_A | \bar{E}] \Pr[\bar{E}] \quad (16)$$

$$\leq \mathbb{E}_A[L_A | E] + \frac{\text{MLD}(H)}{8}. \quad (17)$$

Finally, observe that conditioned on event E , there is an expert in G_i that makes at most $\text{MLD}(H)/4$ mistakes on S . Hence, applying Theorem 8.7, we get that

$$\mathbb{E}_A[L_A | E] \leq \text{MLD}(H)/4 + \sqrt{\frac{1}{2} \text{RepDim}(H) \text{MLD}(H)},$$

where we have used the fact that the natural logarithm of the number of experts in G_i is at most the representation dimension of hypothesis class H . Hence, putting together the arguments from the previous two paragraphs, we have that

$$\mathbb{E}_A[L_A] \leq 3\text{MLD}(H)/8 + \sqrt{\frac{1}{2} \text{RepDim}(H) \text{MLD}(H)}.$$

Next, by Lemma 8.6, we have that $\mathbb{E}_A[L_A] \geq \text{MLD}(H)/2$. Combining the above two equations, we get that

$$\frac{\text{MLD}(H)}{2} \leq 3\text{MLD}(H)/8 + \sqrt{\frac{1}{2} \text{RepDim}(H) \text{MLD}(H)} \implies \frac{\text{MLD}(H)}{32} \leq \text{RepDim}(H).$$

□

References

- [ALMM19] 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*, STOC 2019, page 852–860, New York, NY, USA, 2019. Association for Computing Machinery.
- [BCHL95] Shai Ben-David, Nicolò Cesa-Bianchi, David Haussler, and Philip M. Long. Characterizations of learnability for classes of $\{0, \dots, n\}$ -valued functions. *J. Comput. Syst. Sci.*, 50(1):74–86, 1995.
- [BCS20] Mark Bun, Marco Leandro Carmosino, and Jessica Sorrell. Efficient, noise-tolerant, and private learning via boosting. In Jacob D. Abernethy and Shivani Agarwal, editors, *Conference on Learning Theory, COLT 2020, 9-12 July 2020, Virtual Event [Graz, Austria]*, volume 125 of *Proceedings of Machine Learning Research*, pages 1031–1077. PMLR, 2020.
- [BEHW89] Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K. Warmuth. Learnability and the vapnik-chervonenkis dimension. *J. ACM*, 36(4):929–965, 1989.
- [BLM20] Mark Bun, Roi Livni, and Shay Moran. An equivalence between private classification and online prediction, 2020.
- [BMNS19] Amos Beimel, Shay Moran, Kobbi Nissim, and Uri Stemmer. Private center points and learning of halfspaces. In Alina Beygelzimer and Daniel Hsu, editors, *Conference on Learning Theory, COLT 2019, 25-28 June 2019, Phoenix, AZ, USA*, volume 99 of *Proceedings of Machine Learning Research*, pages 269–282. PMLR, 2019.
- [BNS13] Amos Beimel, Kobbi Nissim, and Uri Stemmer. Private learning and sanitization: Pure vs. approximate differential privacy. In Prasad Raghavendra, Sofya Raskhodnikova, Klaus Jansen, and José D. P. Rolim, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 363–378, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
- [BNS19] 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.
- [BNSV15] 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.
- [DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In Shai Halevi and Tal Rabin, editors, *Theory of Cryptography*, pages 265–284, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
- [DRV10] Cynthia Dwork, Guy N. Rothblum, and Salil P. Vadhan. Boosting and differential privacy. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA*, pages 51–60. IEEE Computer Society, 2010.
- [DSBDSS11] Amit Daniely, Sivan Sabato, Shai Ben-David, and Shai Shalev-Shwartz. Multiclass learnability and the erm principle. In Sham M. Kakade and Ulrike von Luxburg, editors, *Proceedings of the 24th Annual Conference on Learning Theory*, volume 19 of *Proceedings of Machine Learning Research*, pages 207–232, Budapest, Hungary, 09–11 Jun 2011. PMLR.

- [FX14] Vitaly Feldman and David Xiao. Sample complexity bounds on differentially private learning via communication complexity. In Maria Florina Balcan, Vitaly Feldman, and Csaba Szepesvári, editors, *Proceedings of The 27th Conference on Learning Theory*, volume 35 of *Proceedings of Machine Learning Research*, pages 1000–1019, Barcelona, Spain, 13–15 Jun 2014. PMLR.
- [GGKM20] Badih Ghazi, Noah Golowich, Ravi Kumar, and Pasin Manurangsi. Sample-efficient proper pac learning with approximate differential privacy, 2020.
- [GGKM21] Badih Ghazi, Noah Golowich, Ravi Kumar, and Pasin Manurangsi. Near-tight closure bounds for the littlestone and threshold dimensions. In Vitaly Feldman, Katrina Ligett, and Sivan Sabato, editors, *Proceedings of the 32nd International Conference on Algorithmic Learning Theory*, volume 132 of *Proceedings of Machine Learning Research*, pages 686–696. PMLR, 16–19 Mar 2021.
- [JKT20] Young Hun Jung, Baekjin Kim, and Ambuj Tewari. On the equivalence between online and private learnability beyond binary classification, 2020.
- [KLM⁺20] Haim Kaplan, Katrina Ligett, Yishay Mansour, Moni Naor, and Uri Stemmer. Privately learning thresholds: Closing the exponential gap. In Jacob D. Abernethy and Shivani Agarwal, editors, *Conference on Learning Theory, COLT 2020, 9-12 July 2020, Virtual Event [Graz, Austria]*, volume 125 of *Proceedings of Machine Learning Research*, pages 2263–2285. PMLR, 2020.
- [KLN⁺08] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith. What can we learn privately? In *2008 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 531–540, 2008.
- [KMST20] Haim Kaplan, Yishay Mansour, Uri Stemmer, and Eliad Tsfadia. Private learning of half-spaces: Simplifying the construction and reducing the sample complexity. In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.
- [Lit87] Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Mach. Learn.*, 2(4):285–318, 1987.
- [LW94] N. Littlestone and M.K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.
- [MT07] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS’07)*, pages 94–103, 2007.
- [Nat89] B. K. Natarajan. On learning sets and functions. *Mach. Learn.*, 4(1):67–97, October 1989.
- [RST15] Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Sequential complexities and uniform martingale laws of large numbers. *Probability theory and related fields*, 161(1):111–153, 2015.
- [Val84] L. G. Valiant. A theory of the learnable. *Commun. ACM*, 27(11):1134–1142, November 1984.

A Boosting Private Learners

The following result shows that differentially private learners with constant accuracy and confidence parameters can be generically boosted to make these parameters arbitrarily small, with only a mild dependence on these parameters.

Theorem A.1. *Let $\epsilon \leq 1/4$ and let H be a binary concept class that has an (ϵ, δ) -differentially private and $(\alpha = 1/4, \beta = 1/4)$ -accurate PAC learner using $SC_{1/4, 1/4}$ samples. Then for every $\alpha, \beta > 0$, there is an (ϵ, δ) -differentially private learner for H using*

$$O\left(\frac{SC_{1/4, 1/4} \cdot \log(\log(1/\alpha)/\beta) \cdot \log(1/\alpha)}{\alpha}\right).$$

samples.

Proof sketch. We apply the framework of private boosting via lazy Bregman projections described in Bun et al. [BCS20]. The hypotheses of the theorem guarantee the existence of a “weak”, i.e., $(1/4, 1/4)$ -accurate, learner A for the class H . Boosting repeatedly runs (a modification of) A on a sequence μ_1, \dots, μ_T of reweightings of the input sample, and aggregates the resulting hypotheses h_1, \dots, h_T into a new hypothesis h^* with arbitrarily small accuracy parameters α, β .

First, we explain how to boost the confidence parameter of this learner from $1/3$ to an arbitrary $\beta > 0$. We do this by repeating the algorithm A some $k = O(\log(1/\beta))$ times, producing a sequence of candidate hypotheses h_1, \dots, h_k . We then use the *exponential mechanism* [MT07] to identify an h_i with approximately minimal error with respect to a fresh sample of size $O(\log(k/\beta)/\epsilon)$. Overall this results in a $(2\epsilon, \delta)$ -differentially private and $(1/3, \beta)$ -accurate PAC learner A' using $O(SC_{1/4, 1/4} \cdot \log(1/\beta))$ samples.

We now use A' to construct a weak learner satisfying the conditions needed to apply the framework in Bun et al. [BCS20]. Namely, the weak learner must be able to generate accurate hypotheses with respect to any distribution over the input sample, as well as a generalized form of differential privacy that ensures similar output distributions given neighboring datasets *and* reweightings. Below, let Δ_n denote the set of probability distributions over $\{1, \dots, n\}$. Moreover, we say that a distribution $\mu \in \Delta_n$ is s -smooth if $\mu(i) \leq s$ for every $i \in \{1, \dots, n\}$.

Claim A.2. *For parameters $n, T \in \mathbb{N}$ and $s \in (0, 1)$, there exists a randomized algorithm $W : \mathcal{X}^n \times \Delta_n \rightarrow \{0, 1\}^{\mathcal{X}}$ with the following properties:*

- *Accuracy: If $1/s \geq O(SC_{1/4, 1/4} \cdot \log(T/\beta))$, then for every sample $D = ((x_1, y_1), \dots, (x_n, y_n))$ and distribution $\mu \in \Delta_n$, with probability at least $1 - \beta/2T$, the weak learner outputs a hypothesis h such that*

$$\sum_{i=1}^n |h(x_i) - y_i| \mu(i) \leq \frac{1}{3}.$$

- *Privacy: For every pair of neighboring samples D, D' and s -smooth distributions μ, μ' at statistical distance at most s , we have that $W(D, \mu)$ and $W(D', \mu')$ are $(2\epsilon, 3\delta)$ -indistinguishable.*

We obtain the algorithm W from A' as follows. On input a training set D and distribution μ , sample $m = 1/16s$ examples without replacement from D according to μ and run A' on the result. By the accuracy guarantee of A' , this meets the desired accuracy condition as long as $m \geq O(SC_{1/4, 1/4} \cdot \log(T/\beta))$. Moreover, by a privacy amplification by subsampling argument [DRV10, Lemma 6.5], we have that $W(D, \mu)$ and $W(D', \mu')$ are $(32\epsilon sm, \delta(1 + e^{8\epsilon sm}))$ -indistinguishable for every $D \sim D'$ and s -smooth μ, μ' at statistical distance s . Our choice of $m = 1/16s$ and $\epsilon \leq 1$ prove the claim.

The lazy Bregman boosting procedure of Bun et al. [BCS20] shows that after $T = O(\log(1/\alpha))$ rounds of boosting, the aggregated hypothesis has training error at most $\alpha/2$ except with probability at most $\beta/2$. Moreover, for every pair of neighboring inputs D, D' , the sequences of distributions μ_1, \dots, μ_T and μ'_1, \dots, μ'_T constructed over the course of boosting are all $(1/\alpha n)$ -smooth and element-wise at statistical distance at

most $1/\alpha n$. The accuracy condition of Claim A.2 kicks in as long as $1/s = \alpha n \geq O(SC_{1/4,1/4} \cdot \log(T/\beta))$, i.e., if $n \geq O(\frac{1}{\alpha} \cdot SC_{1/4,1/4} \cdot \log(T/\beta))$. Meanwhile, by the privacy condition of Claim A.2, each individual round of boosting is $(2\epsilon, 3\delta)$ -differentially private, so by the basic composition theorem for differential privacy, the algorithm as a whole is $(2T\epsilon, 3T\delta)$ -differentially private.

Finally, we apply a (different) privacy amplification by subsampling argument [BNSV15, Lemma 4.12] one more time to convert this algorithm from a $(2T\epsilon, 3T\delta)$ -differentially private algorithm with sample complexity $O(\frac{1}{\alpha} \cdot SC_{1/4,1/4} \cdot \log(T/\beta))$ into a (ϵ, δ) -differentially private algorithm with sample complexity

$$O\left(\frac{SC_{1/4,1/4} \cdot \log(T/\beta) \cdot T}{\alpha}\right) = O\left(\frac{SC_{1/4,1/4} \cdot \log(\log(1/\alpha)/\beta) \cdot \log(1/\alpha)}{\alpha}\right).$$

By standard generalization bounds for approximate empirical risk minimization [BEHW89], achieving low error on a training set of size $O(\frac{1}{\alpha} \cdot (VC(H) + \log(1/\alpha\beta)))$ suffices to achieve low error on the population, where VC denotes the Vapnik-Chervonenkis dimension. This is already true using the number of samples given above, using the fact that the private learner we started with is a PAC learner and hence $SC_{1/4,1/4} \geq \Omega(VC(H))$. \square

Corollary A.3 (Boosted Learner from [GGKM20]). *Let G be any binary hypothesis class with Littlestone dimension $d_L \geq 0$. Then, for any $\epsilon \in [0, 1/4]$, $\delta, \alpha, \beta \in [0, 1]$, for some*

$$n = O\left(\frac{d_L^6 \log^4\left(\frac{d_L}{\alpha\beta\epsilon\delta}\right)}{\epsilon\alpha}\right),$$

there is an (ϵ, δ) -differentially private, (α, β) -accurate PAC learning algorithm B for G with sample complexity upper bounded by n .

Proof. Using the binary PAC learner from Ghazi et al. [GGKM20], by the sample complexity bound given in theorem 3.6, for $\epsilon \in [0, 1/4]$, $\delta \in [0, 1]$ we get an (ϵ, δ) -differentially private $(1/4, 1/4)$ -accurate PAC learner with sample complexity upper bounded by $O\left(\frac{d_L^6 \log^2\left(\frac{d_L}{\epsilon\delta}\right)}{\epsilon}\right)$. For the case where $d_L = 0$, this directly proves the corollary. So assume $d_L > 0$. Applying the boosting procedure described in Theorem A.1, we get an (ϵ, δ) -differentially private (α, β) -accurate PAC learner with sample complexity upper bounded by $O\left(\frac{d_L^6 \log^2\left(\frac{d_L}{\epsilon\delta}\right) \cdot \log(\log(1/\alpha)/\beta) \cdot \log(1/\alpha)}{\epsilon\alpha}\right) = O\left(\frac{d_L^6 \log^4\left(\frac{d_L}{\epsilon\delta\alpha\beta}\right)}{\epsilon\alpha}\right)$. \square