The sample complexity of multi-distribution learning

Binghui Peng BP2601@COLUMBIA.COM
Columbia University, United States

Editors: Shipra Agrawal and Aaron Roth

Abstract

Multi-distribution learning generalizes the classic PAC learning to handle data coming from multiple distributions. Given a set of k data distributions and a hypothesis class of VC dimension d, the goal is to learn a hypothesis that minimizes the maximum population loss over k distributions, up to ϵ additive error. In this paper, we settle the sample complexity of multi-distribution learning by giving an algorithm of sample complexity $\widetilde{O}((d+k)\epsilon^{-2})\cdot(k/\epsilon)^{o(1)}$. This matches the lower bound up to sub-polynomial factor and resolves the COLT 2023 open problem of Awasthi, Haghtalab and Zhao (Awasthi et al., 2023).

Keywords: Learning theory, PAC learning, sample complexity, multi-distribution learning

1. Introduction

Multi-distribution learning is a natural generalization of the classic PAC learning (Valiant, 1984) to multiple distributions setting. Given a hypothesis class $\mathcal{H}$ and a set of k distributions $\mathcal{D}_1,\dots,\mathcal{D}_k$ over the data universe $\mathcal{X}\times\{0,1\}$, multi-distribution learning seeks for a hypothesis f that achieves near optimal worst case guarantee over all distributions

$$\max_{i\in[k]}\ell_{\mathcal{D}_i}(f)\le\arg\min_{h^*\in\mathcal{H}}\max_{i\in[k]}\ell_{\mathcal{D}_i}(h^*)+\epsilon\quad\text{where}\quad\ell_{\mathcal{D}}(h):=\Pr_{(x,y)\sim\mathcal{D}}[h(x)\neq y].$$

The formulation of multi-distribution learning captures many important applications: For fairness consideration, the distributions represent heterogeneous populations of protected attributes and multi-distribution learning yields the minimax group fairness (Mohri et al., 2019; Shekhar et al., 2021; Rothblum and Yona, 2021; Diana et al., 2021; Tosh and Hsu, 2022); In the context of multi-task or federated learning, multi-distribution learning captures the notion of robustness and yields worst case guarantees (Sener and Koltun, 2018); For group distributional robustness optimization, multi-distribution learning obtains uniform guarantee to all pre-defined groups of distributions (Rahimian and Mehrotra, 2019; Sagawa et al., 2019, 2020; Duchi and Namkoong, 2021).

Similar to the study of PAC learning (Blumer et al., 1989; Auer and Ortner, 2004; Hanneke, 2016; Larsen, 2023; Aden-Ali et al., 2023), one important research question is to characterize the sample complexity of multi-distribution learning. It is not hard to see that the learnability is still captured by the VC dimension (Vapnik and Chervonenkis, 1971), and $\widetilde{\Theta}(kd/\epsilon^2)$ samples are both necessary and sufficient to guarantee uniform convergence (Blum et al., 2017). There is a long line of work (Blum et al., 2017; Nguyen and Zakynthinou, 2018; Chen et al., 2018; Haghtalab et al., 2022; Awasthi et al., 2023) that try to pin down the optimal sample complexity. In the realizable setting, where the optimal hypothesis $h^*\in\mathcal{H}$ has zero error, Blum et al. (2017); Nguyen and Zakynthinou (2018); Chen et al. (2018) give algorithms of sample complexity $\widetilde{O}((d+k)/\epsilon)$ using the idea of multiplicative weight update. The sample complexity for the general agnostic learning setting is more challenging. A recent breakthrough of Haghtalab, Jordan and Zhao (Haghtalab et al.,

2022) gives an algorithm of sample complexity $\widetilde{O}((k+\log(|\mathcal{H}|))/\epsilon^2)$, this is optimal assuming the hypothesis class $\mathcal{H}$ is finite. For infinite hypothesis class, Awasthi et al. (2023) gives two algorithms: One bases on the multiplicative weight update and has sample complexity $\widetilde{O}((d+k)/\epsilon^4)$; The other bases on the finite hypothesis algorithm (Haghtalab et al., 2022) and has sample complexity $\widetilde{O}((d+k)/\epsilon^2 + kd/\epsilon)$. Nevertheless, there is still a significant gap between the upper and lower bound ($\widetilde{O}(\min\{(d+k)/\epsilon^4, (d+k)/\epsilon^2 + kd/\epsilon\})$ vs. $\widetilde{\Omega}((d+k)/\epsilon^2)$), and as elaborated in the COLT 2023 open problem publication (Awasthi et al., 2023), fundamental barriers exist for all current approaches. They pose the open question of obtaining the optimal sample complexity for multi-distribution learning.

In this paper, we address the open question of (Awasthi et al., 2023) and give an algorithm of optimal sample complexity (up to sub-polynomial factor). Our result is formally stated as below.

Theorem 1 (Multi-distribution learning) *Let k be the number of distributions, d be the VC dimension of the hypothesis class. For any $\epsilon > 0$, there is an algorithm that outputs an ϵ-optimal classifier with probability $1-\delta$, and has sample complexity*

$$\frac{(d+k)\log(d/\delta)}{\epsilon^2} \cdot (k/\epsilon)^{o(1)}.$$

An immediate implication of Theorem 1 is that *multi-distribution learning is no harder than (single-distribution) PAC learning* for sample complexity consideration.

1.1. Technical overview: Achieving optimal sample complexity via recursive width reduction

We give an overview of our algorithm for Theorem 1, the key ingredient is a recursive width reduction procedure.

The MWU framework The major technique used by all previous works (Blum et al., 2017; Nguyen and Zakynthinou, 2018; Chen et al., 2018; Haghtalab et al., 2022; Awasthi et al., 2023) is the multiplicative weight update (MWU) framework (Arora et al., 2012). We first review this framework. The algorithm views the k distributions $\mathcal{D}_1, \ldots, \mathcal{D}_k$ as k experts and runs MWU for T rounds. At each round $t \in [T]$, the algorithm performs empirical risk minimization (ERM) and obtains an ϵ-optimal hypothesis $f_t \in \mathcal{H}$ over the mixed distribution $\mathcal{D}^{(t)} = \sum_{i\in[k]} p_t(i)\mathcal{D}_i$. Here p_t is the strategy of MWU, and it is updated by the loss of f_t over distributions $(\mathcal{D}_i)_{i\in[k]}$, i.e. $\ell_t = (\ell_{\mathcal{D}_i}(f_t))_{i\in[k]} \in [0,1]^k$. The final output is taken to be $f = \frac{1}{T}\sum_{t\in[T]} f_t$. The regret guarantee of MWU ensures that the worst case error of f over $\mathcal{D}_1, \ldots, \mathcal{D}_k$ is close to the average error of f_t over $\mathcal{D}^{(t)}$, which is at most ϵ. For the sample complexity, in the realizable setting, the sample complexity per round is $\widetilde{O}((k+d)/\epsilon)$ and $T = \Omega(\log(k))$ rounds are needed; under the agnostic learning setting, the sample complexity per round is $\widetilde{O}((k+d)/\epsilon^2)$ and $T = \widetilde{\Omega}(1/\epsilon^2)$ rounds are needed. It is not hard to see that both terms are tight in the worst case and they form the major technical obstacle for obtaining the optimal sample complexity.

Width reduction Our approach also falls into this MWU framework and the key idea for improvement is *recursive width reduction*. In the literature of online learning, width refers to the maximum range of the loss vector, i.e., $\max_{i\in[k]} \ell_t(i) - \min_{i\in[k]} \ell_t(i)$. To get a quick sense of how width reduction works, recall the regret guarantee of MWU equals $O(\sqrt{\log(k)/TB})$, where B is the width of the loss vector ℓ_t and it equals 1 in the above framework. In order to get ϵ-regret, one

needs to take $T \geq \widetilde{\Omega}(1/\epsilon^2)$. If one can reduce the width, then it immediately reduces the number of iterations for MWU, and consequently, reduces the sample complexity.

For now, we assume the optimal error $\mathrm{OPT} := \min_{h^* \in \mathcal{H}} \max_{i \in [k]} \ell_{\mathcal{D}_i}(h^*)$ is known to the algorithm, this assumption can be easily removed and we defer the discussion to the end. Our idea is to reduce the width using the algorithm itself. Recall we have an algorithm of sample complexity $\widetilde{O}((d+k)/\epsilon^4)$ using MWU. Now at each round $t \in [T]$, we first draw $\widetilde{O}(d/\epsilon^2)$ samples from $\mathcal{D}^{(t)}$. Instead of running ERM, we first obtain a subset of the hypothesis $\mathcal{H}' \subseteq \mathcal{H}$ by removing all hypothesis $h \in \mathcal{H}$ that has error more than $\mathrm{OPT} + \epsilon$. We then run the MWU algorithm with error parameter $\epsilon' = \epsilon^{1/2}$ over hypothesis class $\mathcal{H}'$ (so the additional samples it needs is still $\widetilde{O}((d+k)/(\epsilon')^4) = \widetilde{O}((d+k)/\epsilon^2)$), and obtain a hypothesis f_t. The hypothesis f_t has additional guarantees on the maximum loss, i.e., $\max_{i \in [k]} \ell_{\mathcal{D}_i}(f_t) \leq \mathrm{OPT} + \epsilon^{1/2}$. This reduces the maximum loss from 1 to $\mathrm{OPT} + \epsilon^{1/2}$. However, we have no guarantee on the minimum loss, and the width could still be as large as $\mathrm{OPT} + \epsilon^{1/2} \approx \Theta(1)$.

To get a lower bound on ℓ_t, we can truncate small entries: If an entry $\ell_t(i) = \ell_{\mathcal{D}_i}(f_t) \leq \mathrm{OPT} - \epsilon^{1/2}$ is small, then we take it as $\ell_t(i) = \mathrm{OPT} - \epsilon^{1/2}$. In this way, the width reduces to $(\mathrm{OPT} + \epsilon^{1/2}) - (\mathrm{OPT} - \epsilon^{1/2}) = 2\epsilon^{1/2}$. However, there is a fatal issue here: There is no reason we can arbitrarily truncate the loss. Recall we need the average loss $\sum_{i \in [k]} p_t(i) \ell_t(i)$ to be close to $\mathrm{OPT} + \epsilon$. If we truncate the small entries, then we increase its value. As a concrete example, if $\mathrm{OPT} = 1/2$, there are $2\epsilon^{1/2}$-fraction of $(\ell_{\mathcal{D}_i}(f_t))_{i \in [k]}$ equal 0, and the other $1 - 2\epsilon^{1/2}$ fraction equal $\mathrm{OPT} + \epsilon^{1/2}$, then there is no way one can truncate the loss.

The next idea is, instead of relying on uniform convergence and selecting the hypothesis class $\mathcal{H}'$ that are ϵ-optimal on $\mathcal{D}^{(t)}$, we need more refined properties of $\mathcal{H}'$ that make the loss $(\ell_{\mathcal{D}_i}(f_t))_{i \in [k]}$ more balanced. To this end, we want the hypothesis class $\mathcal{H}'$ satisfies the following two properties:

- **Soundness.** The optimal classifier survives, i.e., $h^* \in \mathcal{H}'$, where $h^* = \arg\min_{h \in \mathcal{H}} \max_{i \in [k]} \ell_{\mathcal{D}_i}(h)$.
- **Completeness.** For any hypothesis $h \in \mathcal{H}'$ that survives, it satisfies the following guarantee. For any subset of distributions $\mathcal{I} \subseteq [k]$, if their weights $\sum_{i \in [n]} p_t(i) \geq 1/2$, then the loss of h on the distribution $\sum_{i \in \mathcal{I}} \frac{p_t(i)}{\sum_{i \in \mathcal{I}} p_t(i)} \mathcal{D}_i$ is at most $\mathrm{OPT} + O(\epsilon)$.

The first property states that the optimal classifier h^* survives, this ensures that it is safe to work with $\mathcal{H}'$ instead of $\mathcal{H}$. The second property is more complicated, but from a high level, it says that any surviving hypothesis in $\mathcal{H}'$ is robust – not only their loss is small on the entire distribution $\mathcal{D}^{(t)} = \sum_{i \in [k]} p_t(i) \mathcal{D}_i$, but also small on any sub-populations $\{\mathcal{D}_i\}_{i \in \mathcal{I}}$ of mass at least $1/2$. Suppose for now, we have achieved these two properties with $\widetilde{O}((d+k)/\epsilon^2)$ samples. Then we can safely truncate the loss $\ell_t(i) = \max\{\ell_{\mathcal{D}_i}(f_t), \mathrm{OPT} - \epsilon^{1/2}\}$ and reduce the width of $\ell_t \in [\mathrm{OPT} - \epsilon^{1/2}, \mathrm{OPT} + \epsilon^{1/2}]^k$ to $2\epsilon^{1/2}$. It remains to argue that the average loss satisfies $\sum_{i \in [k]} p_t(i) \ell_t(i) \leq \mathrm{OPT} + O(\epsilon)$.[1] To this end, we sort the loss $\{\ell_{\mathcal{D}_i}(f_t)\}_{i \in [k]}$ and assume $\ell_{\mathcal{D}_1}(f_t) \geq \cdots \geq \ell_{\mathcal{D}_k}(f_t)$ w.l.o.g. Suppose $k' \in [k]$ is the smallest index such that $\sum_{i \leq k'} p_t(i) \geq 1/2$.

- **Case 1.** If $\ell_{\mathcal{D}_{k'}}(f_t) \geq \mathrm{OPT} - \epsilon^{1/2}$ (i.e., no truncation at the larger half), then by the completeness property,[2] the larger half has loss at most $\mathrm{OPT} + O(\epsilon)$, since there is no truncation.

1. We also need to ensure $\ell_t(i)$ is an overestimate of $\ell_{\mathcal{D}_i}(f_t)$, but this is trivial from the definition.
2. We actually need the hypothesis f_t to be a weighted average of hypothesis in $\mathcal{H}'$ in order to inherit the completeness property of $\mathcal{H}'$, this is naturally satisfied by algorithms in the MWU framework.

Meanwhile, the loss of the smaller half is no more than the larger half, so the average loss is at most $\mathrm{OPT} + O(\epsilon)$.

- **Case 2.** Otherwise, if $\ell_{\mathcal{D}_{k'}}(f_t) < \mathrm{OPT} - \epsilon^{1/2}$, then performing truncation is still fine, because more than $1/2$-fraction of distributions have loss smaller than $\mathrm{OPT} - \epsilon^{1/2}$, while the rest of them (at most $1/2$-fraction) have loss at most $\mathrm{OPT} + \epsilon^{1/2}$.

Now we elaborate a bit on how we achieve both soundness and completeness. It proceeds in two steps. First, we draw $\widetilde{O}(d/\epsilon)$ samples $S_1^{(t)}$ from $\mathcal{D}^{(t)}$ and look at the projection of $\mathcal{H}$ on $S_1^{(t)}$. We construct an ϵ-cover $\mathcal{C}_{\mathcal{H}}$ of $\mathcal{H}$ by including an arbitrary hypothesis for each projection. This is a fairly standard trick (e.g. see Alon et al. (2019)). Next, we draw another $\widetilde{O}((d+k)/\epsilon^2)$ samples $S_2^{(t)}$ from $\mathcal{D}^{(t)}$ and run the following test on $S_2^{(t)}$. For each hypothesis $h \in \mathcal{C}_{\mathcal{H}}$ in the ϵ-cover, if there exists a subset $\mathcal{I} \subseteq [k]$ of distributions such that (1) $\sum_{i\in\mathcal{I}} p_t(i) \geq 1/2$ and (2) the empirical loss of h on $\sum_{i\in\mathcal{I}} \frac{p_t(i)}{\sum_{i\in\mathcal{I}} p_t(i)} \mathcal{D}_i$ is larger that $\mathrm{OPT} + O(\epsilon)$, then we remove h, as well as all hypothesis that have the same projection as h on $S_1^{(t)}$, from $\mathcal{H}'$. In the proof, we show that this test guarantees soundness and completeness with high probability.

Recursive width reduction The width reduction procedure described above reduces the width from 1 to $2\epsilon^{1/2}$. The regret now becomes $\widetilde{O}(\epsilon^{1/2}/\sqrt{T})$, and it suffices to take $T = \widetilde{O}(1/\epsilon)$. The sample complexity per round remains $\widetilde{O}((d+k)/\epsilon^2)$ and there are $T = \widetilde{O}(1/\epsilon)$ rounds, so we improve the sample complexity from $\widetilde{O}((d+k)/\epsilon^4)$ to $\widetilde{O}((d+k)/\epsilon^3)$. We can continue this process, and use this new algorithm for width reduction. In particular, at each round, we can take the error parameter $\epsilon' = \epsilon^{2/3}$ and use $\widetilde{O}((d+k)/(\epsilon')^3) = \widetilde{O}((d+k)/(\epsilon)^2)$ samples to reduce the maximum loss to $\mathrm{OPT} + \epsilon' = \mathrm{OPT} + \epsilon^{2/3}$ (instead of $\mathrm{OPT} + \epsilon^{1/2}$). The regret now becomes $\widetilde{O}(\epsilon^{2/3}/\sqrt{T})$ and we can further reduce the number of rounds to $T = \widetilde{O}(\epsilon^{-2/3})$ and the sample complexity to $\widetilde{O}((d+k)/\epsilon^{8/3})$. We repeat the above process and obtain an algorithm of sample complexity $O((d+k)/\epsilon^2) \cdot (k/\epsilon)^{o(1)}$.

Remove prior knowledge on OPT The above algorithm requires prior knowledge on the optimal value (for both the testing step and the truncation step), we next remove this assumption. It is not hard to see that the above algorithm succeeds with an ϵ-approximate $\mathrm{OPT}' \in [\mathrm{OPT} - \epsilon, \mathrm{OPT} + \epsilon]$ (i.e., no need for the exact value of OPT). Hence, we can run $1/\epsilon$ threads of the algorithm with $\mathrm{OPT}' = \epsilon, 2\epsilon, \ldots, 1$ and take the best one. This has sample complexity $(k+d)/\epsilon^2 \cdot (1/\epsilon) = (k+d)/\epsilon^3$ (we omit the $o(1)$ term for simplicity) and does not require any knowledge of OPT. Next, we take $\epsilon' = \epsilon^{2/3}$ and runs the algorithm with $(k+d)/(\epsilon')^3 = (k+d)/\epsilon^2$ samples . The output hypothesis has error at most $\mathrm{OPT} + O(\epsilon') = \mathrm{OPT} + O(\epsilon^{2/3})$. Now, we can reduce the size of the grid search and only search for $\epsilon^{2/3}/\epsilon = \epsilon^{-1/3}$ possible value of OPT' (instead of $1/\epsilon$), this reduces the sample complexity from $(k+d)/\epsilon^3$ to $(k+d)/\epsilon^{7/3}$. Again, we repeat this process and get an algorithm of sample complexity $O((d+k)/\epsilon^2) \cdot (k/\epsilon)^{o(1)}$ without any knowledge of OPT.

1.2. Related work

The sample complexity of multi-distribution learning has been extensively studied in the past decade (Blum et al., 2017; Nguyen and Zakynthinou, 2018; Qiao, 2018; Chen et al., 2018; Tao et al., 2019; Blum et al., 2021; Haghtalab et al., 2022; Awasthi et al., 2023). The optimal sample complexity has been derived in the realizable setting (Blum et al., 2017; Nguyen and Zakynthinou, 2018; Chen

et al., 2018). For the more general agnostic learning setting, the optimal sample complexity has been obtained for finite hypothesis class (Haghtalab et al., 2022) but the question is widely open for VC classes, we refer interesting readers for the open problem publication of (Awasthi et al., 2023) for an excellent coverage on the literature.

The multi-distribution learning has applications to fairness (Hébert-Johnson et al., 2018; Mohri et al., 2019; Shekhar et al., 2021; Rothblum and Yona, 2021; Tosh and Hsu, 2022) and group distributional robust optimization (Ben-Tal et al., 2009; Rahimian and Mehrotra, 2019; Sagawa et al., 2019, 2020; Duchi and Namkoong, 2021). It is also closely related to multi-task learning (Caruana, 1997), distributed learning (Balcan et al., 2012), federated learning (McMahan et al., 2017), meta learning (Finn et al., 2017) and continual learning (Chen et al., 2022).

Our approach can be seen as a boosting framework for (agnostic) multi-distribution learning. It converts a weak multi-distribution learner into one with better sample complexity guarantee. There is a vast literature on boosting (Schapire, 1990; Freund and Schapire, 1997; Freund et al., 1999; Ben-David et al., 2001; Mansour and McAllester, 2002; Kalai and Servedio, 2003; Kalai et al., 2008; Balcan et al., 2012; Schapire, 2013; Beygelzimer et al., 2015; Brukhim et al., 2020; Alon et al., 2021; Brukhim et al., 2021, 2023), but to the best of our knowledge, it is the first time that width reduction has been used – we hope it could find broad applications for boosting. The idea of width reduction traces back to the seminal work of positive LP solver (Garg and Könemann, 2007) and approximate max flow (Christiano et al., 2011), which use separate subroutines for width reduction. The idea of recursive width reduction (recursively applying the algorithm itself to reduce the width) has been introduced recently by Peng and Zhang (2023) and it is crucial for the recent development of low memory online learning algorithm (Peng and Zhang, 2023; Peng and Rubinstein, 2023). These previous work are very inspiring, but our way of width reduction, which forms the major challenging part of the proof, is unique and different.

Concurrent and independent work We were recently made aware of the concurrent and independent work of Zhang, Zhan, Chen, Du and Lee (Zhang et al., 2023), which obtains the similar result as Theorem 1. Moreover, their result has the optimal sample complexity up to *polylogarithmic* factor, and their algorithm is oracle efficient. Their result is derived via a different set of technique, which relies on sample reuse.

2. Preliminary

Let $\mathcal{X}$ be the data universe and $\mathcal{Y} = \{0, 1\}$ be binary labels. Let $\mathcal{H}$ be a hypothesis class of VC dimension d, a hypothesis $h \in \mathcal{H}$ maps the data universe $\mathcal{X}$ to the binary label $\mathcal{Y}$. For any hypothesis $f : \mathcal{X} \to \mathcal{Y}$ and any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, the population loss of f over $\mathcal{D}$ equals $\ell_{\mathcal{D}}(f) := \Pr_{(x,y)\sim\mathcal{D}}[f(x) \neq y]$.

Definition 2 (Multi-distribution learning) *Let $\epsilon > 0$ be the error parameter. In the task of multi-distribution learning, there are k distributions $\mathcal{D}_1, \ldots, \mathcal{D}_k$ over $\mathcal{X} \times \mathcal{Y}$. The goal is to learn a hypothesis f that minimizes the maximum loss, i.e.,*

$$\max_{i\in[k]} \ell_{\mathcal{D}_i}(f) \leq \min_{h^*\in\mathcal{H}} \max_{i\in[k]} \ell_{\mathcal{D}_i}(h^*) + \epsilon. \tag{1}$$

We say an algorithm is an (ϵ, δ)-multi-distribution learner if its output satisfies Eq. (1) with probability at least $1 - \delta$. In the rest of this paper, we write $h^* \in \mathcal{H}$ to be the hypothesis that obtains the

minimum loss and OPT be the minimum loss, i.e.,

$$h^* = \arg\min_{h\in\mathcal{H}} \max_{i\in[k]} \ell_{\mathcal{D}_i}(h) \qquad \text{and} \qquad \mathrm{OPT} = \max_{i\in[k]} \ell_{\mathcal{D}_i}(h^*).$$

For any set $S = \{x_1, \ldots, x_n\} \in \mathcal{X}^d$, let $\mathcal{H}(S) := \{(h(x_1), \ldots, h(x_n)) : h \in \mathcal{H}\} \subseteq \{0,1\}^n$ be the projection of $\mathcal{H}$ onto S. The Sauer–Shelah Lemma gives an upper bound on the size $|\mathcal{H}(S)|$.

Lemma 3 (Sauer–Shelah Lemma (Sauer, 1972; Shelah, 1972)) *Let $\mathcal{H}$ be a hypothesis class with VC dimension d, then for any $S \subseteq \mathcal{X}$ with $|S| = n$, $|\mathcal{H}(S)| \le \sum_{i=0}^{d} \binom{n}{i}$. In particular, $|\mathcal{H}(S)| \le (en/d)^d$ if $n \ge d$.*

The multiplicative weight updating (Littlestone and Warmuth, 1994) is a classic algorithm for online learning. An online learning task can be seen as a repeated game between an algorithm and the nature for a sequence of T rounds. Let $[n] = \{1, 2, \ldots, n\}$ and let Δ_n be all probability distributions over $[n]$. The MWU algorithm commits a distribution $p_t \in \Delta_n$ over a set of n experts at each round $t \in [T]$, and then the nature reveals the loss $\ell_t \in \mathbb{R}^n$ for experts $[n]$. The goal is to minimize the regret $\sum_{t\in[T]} \langle p_t, \ell_t\rangle - \min_{i^*\in[n]} \sum_{t\in[T]} \ell_t(i^*)$

Algorithm 1 Multiplicative weight update

1: **for** $t = 1, 2, \ldots, T$ **do**
2: Compute $p_t \in \Delta_n$ over experts such that $p_t(i) \propto \exp(-\eta \sum_{\tau=1}^{t-1} \ell_\tau(i))$ for $i \in [n]$
3: Observe the loss vector ℓ_t and receives loss $\langle p_t, \ell_t\rangle$
4: **end for**

Lemma 4 (Regret guarantee of MWU (Arora et al., 2012)) *Let n be the number of experts, T be the number of days, B be the width of the loss sequence, i.e., the loss vector $\ell_t \in [\rho_t, \rho_t + B]^n$ at each day $t \in [T]$. Let $\eta = \sqrt{\log(n)/T}/B$ be the learning rate, then the MWU algorithm guarantees*

$$\sum_{t\in[T]} \langle p_t, \ell_t\rangle - \min_{i^*\in[n]} \sum_{t\in[T]} \ell_t(i^*) \le \frac{\log n}{\eta} + \eta T B^2 = 2\sqrt{\log(n) T} B.$$

3. The boosting framework

We provide a general boosting framework that takes an arbitrary multi-distribution learning algorithm, reduces its error while incurring a mild overhead on the sample complexity. The boosting framework is formally described in Algorithm 2. It contains several subroutines, whose pseudocodes are presented in Algorithm 3-5. The input of BOOSTLEARNER (Algorithm 2) consists of the hypothesis class $\mathcal{H}$, k distributions $\mathcal{D}_1, \ldots, \mathcal{D}_k$, the multi-distribution learning algorithm MULTILEARNERORACLE, as well as an estimate OPT' on the optimal loss.

BOOSTLEARNER views $\mathcal{D}_1, \ldots, \mathcal{D}_k$ as k experts, and runs MWU over them for T rounds. At each round $t \in [T]$, BOOSTLEARNER maintains a strategy $p_t \in \Delta_k$ over k data distributions, and let $\mathcal{D}^{(t)} = \sum_{i\in[k]} p_t(i)\mathcal{D}_i$ be the mixed distribution of $\mathcal{D}_1, \ldots, \mathcal{D}_k$. BOOSTLEARNER proceeds in a few steps.

Construct ϵ-cover of $\mathcal{H}$ The first step is to construct an ϵ-cover of the hypothesis $\mathcal{H}$ on the distribution $\mathcal{D}^{(t)}$ (Line 3 of Algorithm 2). CONSTRUCTCOVER (Algorithm 3) samples $m_1 = \widetilde{O}(d/\epsilon)$

data points $S_1^{(t)}$ from $\mathcal{D}^{(t)}$, the cover $\mathcal{C}_{\mathcal{H}} \subseteq \mathcal{H}$ is constructed by including an arbitrary hypothesis $h \in \mathcal{H}$ for each projection of $\mathcal{H}(S_1^{(t)})$.

Filter $\mathcal{H}$ Given the ϵ-cover $\mathcal{C}_{\mathcal{H}}$, the next step is to filter $\mathcal{H}$ and only keep a subset of good hypothesis $\mathcal{H}' \subseteq \mathcal{H}$ (Line 4 of Algorithm 2). The FILTER procedure (Algorithm 4) draws $m_2 = \widetilde{O}(\frac{d+k}{\epsilon^2})$ samples $S_2^{(t)}$ from $\mathcal{D}^{(t)}$ as a test set. For each hypothesis h in the cover $\mathcal{C}_{\mathcal{H}}$, it goes through all subsets $\mathcal{I}$ of $[k]$. If the probability mass $\sum_{i\in\mathcal{I}} p_t(i)$ is large enough (i.e., greater than $1/2$) and the empirical loss of h on the mixture distribution $\sum_{i\in\mathcal{I}} \frac{p_t(i)}{\sum_{i\in\mathcal{I}} p_t(i)}\mathcal{D}_i$ is large (i.e., great than $\mathrm{OPT}' + 8\epsilon$), then it removes h, as well as any hypothesis $h' \in \mathcal{H}$ that has the same projection as h on $S_1^{(t)}$, from $\mathcal{H}'$.

Evoke the oracle After obtaining the new hypothesis class $\mathcal{H}' \subseteq \mathcal{H}$, BOOSTLEARNER evokes the oracle MULTILEARNERORACLE with hypothesis $\mathcal{H}'$ and distributions $\mathcal{D}_1, \ldots, \mathcal{D}_k$, and obtains a hypothesis f_t.

Construct the loss vector Given the hypothesis f_t, BOOSTLEARNER constructs the loss vector ℓ_t and feeds it to MWU (Line 6-7 of Algorithm 2). ESTIMATE (Algorithm 5) draws $\widetilde{O}(1/\epsilon^2)$ samples from each distribution $\mathcal{D}_i$ and compute the empirical loss $\hat{\ell}_{\mathcal{D}_i}(f_t)$ of f_t on $\mathcal{D}_i$. Instead directly using this empirical loss, ESTIMATE further truncates loss entries that are below $\mathrm{OPT}' - \alpha$ (Line 4 of Algorithm 5), here α is the error of MULTILEARNERORACLE.

Final output The final output is taken to be the average of $\{f_t\}_{t\in[T]}$. In particular, the output $f = \frac{1}{T}\sum_{t\in[T]} f_t$ is defined as

$$\Pr[f(x) = 1] = \frac{1}{T}\sum_{t\in[T]} \Pr[f_t(x) = 1] \qquad \forall x \in \mathcal{X}.$$

Algorithm 2 BOOSTLEARNER$(\mathcal{H}, \mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_k,$ MULTILEARNERORACLE$, \mathrm{OPT}')$

1: **for** $t = 1, 2, \ldots, T$ **do**
2: $\quad \mathcal{D}^{(t)} \leftarrow \sum_{i\in[k]} p_t(i)\mathcal{D}_i$ ▷ $p_t \in \Delta_k$ is the strategy of MWU
3: $\quad \mathcal{C}_{\mathcal{H}} \leftarrow$ CONSTRUCTCOVER$(\mathcal{H}, \mathcal{D}^{(t)})$
4: $\quad \mathcal{H}' \leftarrow$ FILTER$(\mathcal{H}, \mathcal{D}^{(t)}, \mathcal{C}_{\mathcal{H}}, \mathrm{OPT}')$
5: $\quad f_t \leftarrow$ MULTILEARNERORACLE$(\mathcal{H}', \mathcal{D}_1, \ldots, \mathcal{D}_k, \mathrm{OPT}')$
6: $\quad \ell_t \leftarrow$ ESTIMATE(f_t)
7: $\quad$ Update the strategy of MWU with loss vector $-\ell_t$
8: **end for**
9: **return** $f = \frac{1}{T}\sum_{t\in[T]} f_t$

3.1. Analysis

Given an (infinite) hypothesis class $\mathcal{H}$, let $\Delta(\mathcal{H})$ be all distributions over $\mathcal{H}$ with finite support. Our goal is to prove

Lemma 5 (Boosting framework) *Suppose* $\mathrm{OPT}' \in [\mathrm{OPT} - \epsilon, \mathrm{OPT} + \epsilon]$ *and* MULTILEARNERORACLE *is an* $(\alpha, \delta/16T)$*-multi-distribution learner whose output* $f_t \in \Delta(\mathcal{H})$*. Let* $T = \log(k)(\alpha/\epsilon)^2$*, then*

Algorithm 3 CONSTRUCTCOVER$(\mathcal{H}, \mathcal{D}^{(t)})$

Sample $m_1 = O(\frac{d\log(kd/\epsilon\delta)}{\epsilon})$ data points $S_1^{(t)} = \{(x_j, y_j)\}_{j\in[m_1]}$ from $\mathcal{D}^{(t)}$
$\mathcal{C}_\mathcal{H} \leftarrow \emptyset$
for $(z_1, \ldots, z_{m_1}) \in \mathcal{H}(S_1^{(t)})$ **do** ▷ $\mathcal{H}(S_1^{(t)})$ is the projection of $\mathcal{H}$ onto $S_1^{(t)}$
    Let $h \in \mathcal{H}$ be an arbitrary hypothesis that satisfies $h(x_j) = z_j$ for all $j \in [m_1]$
    $\mathcal{C}_\mathcal{H} \leftarrow \mathcal{C}_\mathcal{H} \cup \{h\}$
end for
return $\mathcal{C}_\mathcal{H}$

Algorithm 4 FILTER$(\mathcal{H}, \mathcal{D}^{(t)}, \mathcal{C}_\mathcal{H}, \mathrm{OPT}')$

Sample $m_2 = O(\frac{(k+d)\log(kd/\epsilon\delta)}{\epsilon^2})$ data points $S_2^{(t)} = \{(x_j, y_j)\}_{j\in[m_2]}$ from $\mathcal{D}^{(t)}$
$\mathcal{H}' \leftarrow \mathcal{H}$
for $h \in \mathcal{C}_\mathcal{H}$ **do**
    for $\mathcal{I} \subseteq [k]$ **do**
        if $\sum_{i\in\mathcal{I}} p_t(i) \geq 1/2$ **and** $\frac{\sum_{j\in[m_2]} 1\{x_j\in\mathcal{D}_\mathcal{I} \wedge h(x_j)\neq y_j\}}{\sum_{j\in[m_2]} 1\{x_j\in\mathcal{D}_\mathcal{I}\}} \geq \mathrm{OPT}' + 8\epsilon$ **then** ▷ $\mathcal{D}_\mathcal{I} := \cup_{i\in\mathcal{I}}\mathcal{D}_i$
            $\mathcal{H}' \leftarrow \mathcal{H}' \backslash \{h' \in \mathcal{H} : h(x) = h'(x)\, \forall x \in S_1^{(t)}\}$
        end if
    end for
end for
return $\mathcal{H}'$

Algorithm 5 ESTIMATE(f_t)

for $i = 1, 2, \ldots, k$ **do**
    Sample $m_3 = O(\log(kd/\epsilon\delta)/\epsilon^2)$ data points $S_{3,i}^{(t)}$ from $\mathcal{D}_i$
    $\hat{\ell}_{\mathcal{D}_i}(f_t) \leftarrow \Pr_{(x,y)\sim S_{3,i}^{(t)}}[f_t(x) \neq y]$
    $\ell_t(i) \leftarrow \max\{\hat{\ell}_{\mathcal{D}_i}(f_t), \mathrm{OPT}' - \alpha\}$ ▷ α is the error of MULTILEARNERORACLE
end for
return ℓ_t

with probability at least $1-\delta$, BOOSTLEARNER *guarantees*

$$\max_{i\in[k]} \ell_{\mathcal{D}_i}(f) \le \mathrm{OPT} + 32\epsilon.$$

We devote to prove Lemma 5 in the rest of this section, and we always make the assumptions that $\mathrm{OPT}' \in [\mathrm{OPT}-\epsilon, \mathrm{OPT}+\epsilon]$ and MULTILEARNERORACLE is an $(\alpha, \delta/32T)$-multi-distribution learner whose output $f_t \in \Delta(\mathcal{H})$. We further assume $\epsilon \le \alpha/32$, otherwise we do not need BOOSTLEARNER.

We first state the guarantee of CONSTRUCTCOVER.

Lemma 6 (Guarantee of CONSTRUCTCOVER, adapted from Lemma 3.3 of Alon et al. (2019)) *For any $t \in [T]$, with probability at least $1-\delta/32T$, $\mathcal{C}_{\mathcal{H}}$ is an ϵ-cover of $\mathcal{H}$. Moreover, for any hypothesis $h \in \mathcal{H}$, let $h' \in \mathcal{C}_{\mathcal{H}}$ be the hypothesis with the same projection over $S_1^{(t)}$, we have*

$$\Pr_{x\sim\mathcal{D}^{(t)}}[h(x) \ne h'(x)] \le \epsilon.$$

We next provide the guarantee of FILTER, the proof can be found at Appendix.

Lemma 7 (Guarantee of FILTER, Part 1) *For each $t \in [T]$, with probability at least $1-\delta/16T$, we have $h^* \in \mathcal{H}'$.*

Lemma 8 (Guarantee of FILTER, Part 2) *For each $t \in [T]$, with probability at least $1-\delta/16T$, it holds that for every hypothesis $h \in \mathcal{H}'$ and every set $\mathcal{I} \subseteq [k]$, if $\sum_{i\in\mathcal{I}} p_t(i) \ge 1/2$, then*

$$\frac{\sum_{i\in\mathcal{I}} p_t(i)\ell_{\mathcal{D}_i}(h)}{\sum_{i\in\mathcal{I}} p_t(i)} \le \mathrm{OPT} + 16\epsilon.$$

Next we make some observations on the output f_t of MULTILEARNERORACLE.

Lemma 9 *For each $t \in [T]$, with probability at least $1-\delta/4T$, we have*

- $\max_{i\in[k]} \ell_{\mathcal{D}_i}(f_t) \le \mathrm{OPT} + \alpha$
- *For any set $\mathcal{I} \subseteq [k]$ with $\sum_{i\in\mathcal{I}} p_t(i) \ge 1/2$, $\frac{\sum_{i\in\mathcal{I}} p_t(i)\ell_{\mathcal{D}_i}(f_t)}{\sum_{i\in\mathcal{I}} p_t(i)} \le \mathrm{OPT} + 16\epsilon$.*

Proof We condition on the high probability event of Lemma 7 and Lemma 8. The first claim follows from

$$\max_{i\in[k]} \ell_{\mathcal{D}_i}(f_t) \le \arg\min_{h\in\mathcal{H}'} \max_{i\in[k]} \ell_{\mathcal{D}_i}(h) + \alpha = \mathrm{OPT} + \alpha$$

The first step follows from the guarantee of MULTILEARNERORACLE, the second step holds since $h^* \in \mathcal{H}'$.

For the second claim, since $f_t \in \Delta(\mathcal{H}')$, we can write $f_t = \sum_j q_j h_j$ for some $h_j \in \mathcal{H}'$ and $\sum_j q_j = 1$. Then for any set $\mathcal{I} \subseteq [k]$ with $\sum_{i\in\mathcal{I}} p_t(i) \ge 1/2$, we have

$$\frac{\sum_{i\in\mathcal{I}} p_t(i)\ell_{\mathcal{D}_i}(f_t)}{\sum_{i\in\mathcal{I}} p_t(i)} = \frac{\sum_j q_j \sum_{i\in\mathcal{I}} p_t(i)\ell_{\mathcal{D}_i}(h_j)}{\sum_{i\in\mathcal{I}} p_t(i)} \le \sum_j q_j(\mathrm{OPT} + 16\epsilon) = \mathrm{OPT} + 16\epsilon.$$

Here the first step holds since

$$\ell_{\mathcal{D}_i}(f_t) = \Pr_{(x,y)\sim\mathcal{D}_i}[f_t(x) \neq y] = \sum_j q_j \Pr_{(x,y)\sim\mathcal{D}_i}[h_j(x) \neq y] = \sum_j q_j \ell_{\mathcal{D}_i}(h_j).$$

and the second step holds due to Lemma 8. ■

Finally, we make some observations on the loss vector ℓ_t constructed by ESTIMATE.

Lemma 10 (Guarantee of ESTIMATE) *For any $t \in [T]$, with probability at least $1 - \frac{\delta}{2T}$, we have*

- $\ell_t(i) \geq \ell_{\mathcal{D}_i}(f_t) - \epsilon$
- $\ell_t(i) \in [\mathrm{OPT} - 2\alpha, \mathrm{OPT} + 2\alpha]$
- $\sum_{i\in[k]} p_t(i)\ell_t(i) \leq \mathrm{OPT} + 20\epsilon$

Proof For each $t \in [T]$, we condition on the high probability event of Lemma 9. For each $i \in [k]$, since $m_3 \geq \Omega(\log(kd/\epsilon\delta)/\epsilon^2)$, by Chernoff bound, with probability at least $1 - \frac{\delta}{32kT}$, the empirical loss $\hat{\ell}_{\mathcal{D}_i}(f_t)$ is ϵ-close to the population loss $\ell_{\mathcal{D}_i}(f_t)$. Taking a union bound, we have

$$\hat{\ell}_{\mathcal{D}_i}(f_t) \in [\ell_{\mathcal{D}_i}(f_t) - \epsilon, \ell_{\mathcal{D}_i}(f_t) + \epsilon] \quad \forall i \in [k] \tag{2}$$

holds with probability at least $1 - \frac{\delta}{32T}$.

For the first claim, we have

$$\ell_t(i) = \max\{\hat{\ell}_{\mathcal{D}_i}(f_t), \mathrm{OPT}' - \alpha\} \geq \hat{\ell}_{\mathcal{D}_i}(f) \geq \ell_{\mathcal{D}_i}(f_t) - \epsilon.$$

For the second claim, we have

$$\ell_t(i) = \max\{\hat{\ell}_{\mathcal{D}_i}(f_t), \mathrm{OPT}' - \alpha\} \geq \mathrm{OPT}' - \alpha \geq \mathrm{OPT} - 2\alpha$$

and

$$\ell_t(i) = \max\{\hat{\ell}_{\mathcal{D}_i}(f_t), \mathrm{OPT}' - \alpha\} \leq \max\{\ell_{\mathcal{D}_i}(f_t), \mathrm{OPT} - \alpha\} + \epsilon \leq \mathrm{OPT} + 2\alpha,$$

where the second step follows from Eq. (2) and $\mathrm{OPT}' \leq \mathrm{OPT} + \epsilon$, the third step holds since $\ell_{\mathcal{D}_i}(f_t) \leq \mathrm{OPT} + \alpha$ (Lemma 9).

For the last claim, w.l.o.g., we can assume $\ell_{\mathcal{D}_1}(f_t) \geq \cdots \geq \ell_{\mathcal{D}_k}(f_t)$. Let $k' \in [k]$ be the smallest index such that $\sum_{i\leq k'} p_t(i) \geq 1/2$. We divide into two cases based on the value of $\ell_{D_{k'}}(f_t)$.

If $\ell_{D_{k'}}(f_t) \geq \mathrm{OPT} - \alpha$, then we have

$$\begin{aligned}\sum_{i\in[k]} p_t(i)\ell_t(i) &= \sum_{i\leq k'} p_t(i)\ell_t(i) + \sum_{i\geq k'+1} p_t(i)\ell_t(i) \\ &= \sum_{i\leq k'} p_t(i) \cdot \max\{\hat{\ell}_{\mathcal{D}_i}(f_t), \mathrm{OPT}' - \alpha\} + \sum_{i\geq k'+1} p_t(i) \cdot \max\{\hat{\ell}_{\mathcal{D}_i}(f_t), \mathrm{OPT}' - \alpha\}\end{aligned} \tag{3}$$

For the first term, we have

$$\begin{aligned}\sum_{i\le k'} p_t(i)\cdot\max\{\hat{\ell}_{\mathcal{D}_i}(f_t),\mathrm{OPT}'-\alpha\} &\le \sum_{i\le k'} p_t(i)\cdot(\max\{\ell_{\mathcal{D}_i}(f_t),\mathrm{OPT}-\alpha\}+\epsilon)\\ &= \sum_{i\le k'} p_t(i)\cdot(\ell_{\mathcal{D}_i}(f_t)+\epsilon)\\ &\le \sum_{i\le k'} p_t(i)\cdot(\mathrm{OPT}+17\epsilon). \end{aligned} \tag{4}$$

The first step follows from $\mathrm{OPT}'\le\mathrm{OPT}+\epsilon$ and Eq. (2), the second step follows from the assumption that $\ell_{D_{k'}}(f_t)\ge\mathrm{OPT}-\alpha$, the third step holds due to Lemma 9.

For the second term, we have

$$\begin{aligned}\sum_{i\ge k'+1} p_t(i)\cdot\max\{\hat{\ell}_{\mathcal{D}_i}(f_t),\mathrm{OPT}'-\alpha\} &\le \sum_{i\ge k'+1} p_t(i)(\max\{\ell_{\mathcal{D}_i}(f_t),\mathrm{OPT}-\alpha\}+\epsilon)\\ &\le \sum_{i\ge k'+1} p_t(i)(\max\{\ell_{\mathcal{D}_{k'}}(f_t),\mathrm{OPT}-\alpha\}+\epsilon)\\ &= \sum_{i\ge k'+1} p_t(i)(\ell_{\mathcal{D}_{k'}}(f_t)+\epsilon)\\ &\le \sum_{i\ge k'+1} p_t(i)(\mathrm{OPT}+17\epsilon) \end{aligned} \tag{5}$$

The first step follows from $\mathrm{OPT}'\le\mathrm{OPT}+\epsilon$ and Eq. (2), the third step follows from the assumption that $\ell_{D_{k'}}(f_t)\ge\mathrm{OPT}-\alpha$, the last step follows from $\ell_{\mathcal{D}_{k'}}(f_t)\le\ell_{\mathcal{D}_i}(f_t)$ $(i\le k')$ and Lemma 9.

Combining Eq. (3)(4)(5), we have proved $\sum_{i\in[k]} p_t(i)\ell_t(i)\le\mathrm{OPT}+17\epsilon$.

If $\ell_{D_{k'}}(f_t)<\mathrm{OPT}-\alpha$, then we have

$$\begin{aligned}\sum_{i\in[k]} p_t(i)\ell_t(i) &= \sum_{i\le k'-1} p_t(i)\ell_t(i)+\sum_{i\ge k'} p_t(i)\ell_t(i)\\ &= \sum_{i\le k'-1} p_t(i)\cdot\max\{\hat{\ell}_{\mathcal{D}_i}(f_t),\mathrm{OPT}'-\alpha\}+\sum_{i\ge k'} p_t(i)\cdot\max\{\hat{\ell}_{\mathcal{D}_i}(f_t),\mathrm{OPT}'-\alpha\}\\ &\le \sum_{i\le k'-1} p_t(i)\cdot(\mathrm{OPT}+\alpha+\epsilon)+\sum_{i\ge k'} p_t(i)\cdot(\mathrm{OPT}-\alpha+\epsilon)\\ &\le \mathrm{OPT}+\epsilon. \end{aligned}$$

Here the third step holds since (1) $\hat{\ell}_{\mathcal{D}_i}(f_t)\le\ell_{\mathcal{D}_i}(f_t)+\epsilon\le\mathrm{OPT}+\alpha+\epsilon$ for any $i\in[k'-1]$ (Lemma 9), and (2) $\hat{\ell}_{\mathcal{D}_i}(f_t)\le\ell_{\mathcal{D}_i}(f_t)+\epsilon\le\ell_{D_{k'}}(f_t)+\epsilon\le\mathrm{OPT}-\alpha+\epsilon$ for any $i\ge k'$ due to the assumption $\ell_{D_{k'}}(f_t)<\mathrm{OPT}-\alpha$. The last step holds since $\sum_{i\le k'-1} p_t(i)<1/2$. This completes the proof for all three claims. ■

Finally, we can prove our main Lemma 5.

Proof [Proof of Lemma 5] We condition on the high probability events of Lemma 6 – 10. For any $i \in [k]$, due to the regret guarantee of MWU, we have

$$\begin{aligned}(\mathrm{OPT}+20\epsilon)T &\geq \sum_{t\in[T]} \langle p_t, \ell_t\rangle \geq \sum_{t\in[T]} \ell_t(i) - 2\sqrt{\log(k)T}\cdot 4\alpha \\ &\geq \sum_{t\in[T]} \ell_{\mathcal{D}_i}(f_t) - \epsilon T - 8\sqrt{\log(k)T}\alpha.\end{aligned}$$

The first step follows from the third claim of Lemma 10, the second step follows from the regret guarantee of MWU (Lemma 4) and the width is at most 4α (the second claim of Lemma 10). The third step follows from the first claim of Lemma 10.

Hence, we have

$$\ell_{\mathcal{D}_i}(f) = \frac{1}{T}\sum_{t\in[T]} \ell_{\mathcal{D}_i}(f_t) \leq \mathrm{OPT}+21\epsilon + 8\sqrt{\log(k)/T}\alpha \leq \mathrm{OPT}+32\epsilon.$$

Here the first step holds since

$$\ell_{\mathcal{D}_i}(f) = \Pr_{(x,y)\sim\mathcal{D}_i}[f(x)\neq y] = \frac{1}{T}\sum_{t\in[T]} \Pr_{(x,y)\sim\mathcal{D}_i}[f_t(x)\neq y] = \frac{1}{T}\sum_{t\in[T]} \ell_{\mathcal{D}_i}(f_t).$$

and the last step holds due to the choice of $T = \log(k)(\alpha/\epsilon)^2$. We complete the proof here. ■

4. Final algorithm

BOOSTLEARNER gives a way of converting a weak multi-distribution learner into a strong one. Recursively evoking itself, we have

Lemma 11 (Recursive application of BOOSTLEARNER) *Let $\mathcal{H}$ be a hypothesis class of VC dimension at most d and $\mathcal{D}_1,\ldots,\mathcal{D}_k$ be k distributions. Given $\mathrm{OPT}' \in [\mathrm{OPT}-\epsilon, \mathrm{OPT}+\epsilon]$, for any integer $r \geq 1$, there is an algorithm with sample complexity*

$$O\left(\frac{(k+d)(\log(k))^{2r}\log(kd/\epsilon\delta)}{\epsilon^{2(1+1/r)}}\right)$$

and with probability at least $1-\delta$, returns a hypothesis $f \in \Delta(\mathcal{H})$ such that

$$\max_{i\in[k]} \ell_{\mathcal{D}_i}(f) \leq \mathrm{OPT}+32\epsilon.$$

Proof We prove by induction. For $r=1$, we run BOOSTLEARNER with MULTILEARNERORACLE selecting an arbitrary hypothesis in $\mathcal{H}'$. In this way, MULTILEARNERORACLE takes 0 additional samples and $\alpha = 1$. By Lemma 5, the output $f \in \Delta(\mathcal{H})$ satisfies

$$\max_{i\in[k]} \ell_{\mathcal{D}_i}(f) \leq \mathrm{OPT}+32\epsilon.$$

The total number of sample it takes equals

$$T\cdot(m_1+m_2+km_3) = O(\log(k)\epsilon^{-2}\cdot(k+d)\epsilon^{-2}\log(kd/\epsilon\delta)) = O((k+d)\epsilon^{-4}\log(k)\log(kd/\epsilon\delta)).$$

Suppose the claim continues to hold up to r, then for $r+1$, we run BOOSTLEARNER and set MULTILEARNERORACLE to be the level r algorithm, with error parameter $\epsilon' = \epsilon^{\frac{r}{r+1}}$ and confidence parameter $\delta' = \delta/16T$. At each round $t \in [T]$, the VC dimension of $\mathcal{H}'$ is at most d, and with high probability, $h^* \in \mathcal{H}'$ (Lemma 7). Therefore, MULTILEARNERORACLE draws

$$m = O\left(\frac{(k+d)(\log(k))^{2r}\log(kd/\epsilon'\delta')}{(\epsilon')^{2(1+1/r)}}\right) = O\left(\frac{(k+d)(\log(k))^{2r}\log(kd/\epsilon\delta)}{\epsilon^2}\right)$$

samples, and with probability at least $1-\delta/16T$, the hypothesis $f_t \in \Delta(\mathcal{H})$ it returns has error at most

$$\alpha = 32\epsilon' = 32\epsilon^{\frac{r}{r+1}}.$$

Therefore, by Lemma 5, we obtain an $(32\epsilon, \delta)$-multi-distribution learner and its sample complexity equals

$$\begin{aligned} T(m_1 + m_2 + km_3 + m) &= O\left(\log(k)\alpha^2\epsilon^{-2} \cdot \frac{(k+d)(\log(k))^{2r}\log(kd/\epsilon\delta)}{\epsilon^2}\right) \\ &= O\left(\frac{(k+d)(\log(k))^{2r+2}\log(kd/\epsilon\delta)}{\epsilon^{2(1+\frac{1}{r+1})}}\right). \end{aligned}$$

This completes the proof. ■

The algorithm described in Lemma 11 still requires the prior knowledge of OPT. Next, we give a way of removing this prior knowledge.

Lemma 12 (Remove prior knowledge of OPT) *For any $\kappa \geq 2$, suppose there exists an algorithm that receives $\mathrm{OPT}' \in [\mathrm{OPT}-\epsilon, \mathrm{OPT}+\epsilon]$, returns a hypothesis of error at most 32ϵ and has sample complexity $g(k,d,\delta)\epsilon^{-\kappa}$. Then there is an algorithm of sample complexity $g(k,d,\epsilon^2\delta/80)\cdot\epsilon^{-\kappa}\log(1/\epsilon)$ and returns a hypothesis of error at most 33ϵ. Here $g(k,d,\delta)$ is a function of k,d,δ.*

Proof We prove the following claim by induction: For any $r \geq 1$, let $\delta_r = \epsilon\delta/2^r 80$, there is an algorithm that draws $O\left(g(k,d,\delta_r)\cdot 40r\cdot\epsilon^{-\kappa-\frac{1}{\kappa^{r-1}}}\right)$ samples and obtains a hypothesis f such that $\max_{i\in[k]}\ell_{\mathcal{D}_i}(f) \leq \mathrm{OPT}+33\epsilon$, without knowing OPT.

Let ALG be the input algorithm that requires prior knowledge of OPT. For the base case $r=1$, we instantiate $B = 1/\epsilon$ threads of ALG, with $\mathrm{OPT}' = b\cdot\epsilon$ ($b \in [B]$), and obtain $\{f_b\}_{b\in[B]}$. We select the best hypothesis among $\{f_b\}_{b\in[B]}$, by drawing $O(\log(k/\epsilon\delta)/\epsilon^2)$ samples from each distribution and estimating the empirical loss of $\{f_b\}_{b\in[B]}$. The output hypothesis f satisfies

$$\max_{i\in[k]}\ell_{\mathcal{D}_i}(f) \leq \min_{b\in[B]}\max_{i\in[k]}\ell_{\mathcal{D}_i}(f_b) + \epsilon \leq \mathrm{OPT}+33\epsilon.$$

since one of the guess OPT' has error at most ϵ. The sample complexity equals $g(k,d,\epsilon\delta/2)\epsilon^{-\kappa-1} + O(k\log(k/\epsilon\delta)/\epsilon^2) \leq g(k,d,\delta_1)\epsilon^{-\kappa-1}$.

Suppose the claim continues to hold for r, then for $r+1$, the algorithm first runs the level r algorithm with error parameter $\epsilon' = \epsilon^{(\kappa+\frac{1}{\kappa^r})/(\kappa+\frac{1}{\kappa^{r-1}})}$. In particular, it draws

$$n_1 = g(k,d,\delta_r/2)\cdot 40r\cdot(\epsilon')^{-\kappa-\frac{1}{\kappa^{r-1}}} = g(k,d,\delta_{r+1})\cdot 40r\cdot\epsilon^{-\kappa-\frac{1}{\kappa^r}}$$

samples and obtains a hypothesis f' of error at most $33\epsilon'$. It then draws $n_2 = O(\log(k/\epsilon\delta)/\epsilon^2)$ samples from each distribution and estimates the empirical loss $\hat{\ell}_{\mathcal{D}_i}(f')$ of f' on each distribution $\mathcal{D}_i$ ($i \in [n]$). Next, it instantiates $B = 33\epsilon'/\epsilon$ threads of ALG, with $\mathrm{OPT}' = \max_{i\in[k]} \hat{\ell}_{\mathcal{D}_i}(f') - b\epsilon$ ($b \in [B]$), and obtains $\{f_b\}_{b\in[B]}$. The number of samples taken in this step equals

$$\begin{aligned} n_3 &= 33(\epsilon'/\epsilon) \cdot g(k, d, \epsilon\delta/80) \cdot \epsilon^{-\kappa} = 33g(k, d, \delta_1) \cdot \epsilon^{-\kappa-1+(\kappa+\frac{1}{\kappa^r})/(\kappa+\frac{1}{\kappa^{r-1}})} \\ &\le 33g(k, d, \delta_{r+1}) \cdot \epsilon^{-\kappa-\frac{1}{\kappa^r}}. \end{aligned}$$

The final output f is the best hypothesis among f' and $\{f_b\}_{b\in[B]}$, measured with their empirical loss. The sample complexity of the algorithm equals

$$n_1 + kn_2 + n_3 \le g(k, d, \delta_{r+1}) \cdot 40(r+1) \cdot \epsilon^{-\kappa-\frac{1}{\kappa^r}}. \tag{6}$$

For the output hypothesis f, if $\max_{i\in[k]} \ell_{\mathcal{D}_i}(f') \le \mathrm{OPT} + 30\epsilon$, then we have

$$\max_{i\in[k]} \ell_{\mathcal{D}_i}(f) \le \max_{i\in[k]} \ell_{\mathcal{D}_i}(f') + \epsilon \le \mathrm{OPT} + 31\epsilon. \tag{7}$$

Otherwise, if $\max_{i\in[k]} \ell_{\mathcal{D}_i}(f') \ge \mathrm{OPT} + 30\epsilon$, the one of the guess $\{\max_{i\in[k]} \hat{\ell}_{\mathcal{D}_i}(f') - b\epsilon\}_{b\in[B]}$ of OPT' is ϵ-close to OPT, and therefore, we have

$$\max_{i\in[k]} \ell_{\mathcal{D}_i}(f) \le \min_{b\in[B]} \max_{i\in[k]} \ell_{\mathcal{D}_i}(f_b) + \epsilon \le \mathrm{OPT} + 33\epsilon \tag{8}$$

where the last step follows from the guarantee of ALG. Combining Eq. (6)(7)(8), we complete the induction. Taking $r = \log(1/\epsilon)$, we finish the proof. ■

Combining Lemma 11, Lemma 12, and taking $r = \omega(1)$, we complete the proof of Theorem 1.

Acknowledgments

The research work is supported by NSF CCF-1703925, IIS-1838154, CCF-2106429, CCF-2107187, CCF-1763970, AF2212233, COLL2134095, COLL2212745.

References

Ishaq Aden-Ali, Yeshwanth Cherapanamjeri, Abhishek Shetty, and Nikita Zhivotovskiy. Optimal pac bounds without uniform convergence. In *2023 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, 2023.

Noga Alon, Raef Bassily, and Shay Moran. Limits of private learning with access to public data. *Advances in neural information processing systems*, 32, 2019.

Noga Alon, Alon Gonen, Elad Hazan, and Shay Moran. Boosting simple learners. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 481–489, 2021.

Sanjeev Arora, Elad Hazan, and Satyen Kale. The multiplicative weights update method: a meta-algorithm and applications. *Theory of Computing*, 8(1):121–164, 2012.

Peter Auer and Ronald Ortner. A new pac bound for intersection-closed concept classes. In *International Conference on Computational Learning Theory*, pages 408–414. Springer, 2004.

Pranjal Awasthi, Nika Haghtalab, and Eric Zhao. Open problem: The sample complexity of multi-distribution learning for vc classes. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 5943–5949. PMLR, 2023.

Maria Florina Balcan, Avrim Blum, Shai Fine, and Yishay Mansour. Distributed learning, communication complexity and privacy. In *Conference on Learning Theory*, pages 26–1. JMLR Workshop and Conference Proceedings, 2012.

Shai Ben-David, Philip M Long, and Yishay Mansour. Agnostic boosting. In *Computational Learning Theory: 14th Annual Conference on Computational Learning Theory, COLT 2001 and 5th European Conference on Computational Learning Theory, EuroCOLT 2001 Amsterdam, The Netherlands, July 16–19, 2001 Proceedings 14*, pages 507–516. Springer, 2001.

Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust optimization*, volume 28. Princeton university press, 2009.

Alina Beygelzimer, Satyen Kale, and Haipeng Luo. Optimal and adaptive algorithms for online boosting. In *International Conference on Machine Learning*, pages 2323–2331. PMLR, 2015.

Avrim Blum, Nika Haghtalab, Ariel D Procaccia, and Mingda Qiao. Collaborative pac learning. *Advances in Neural Information Processing Systems*, 30, 2017.

Avrim Blum, Nika Haghtalab, Richard Lanas Phillips, and Han Shao. One for one, or all for all: Equilibria and optimality of collaboration in federated learning. In *International Conference on Machine Learning*, pages 1005–1014. PMLR, 2021.

Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth. Learnability and the vapnik-chervonenkis dimension. *Journal of the ACM (JACM)*, 36(4):929–965, 1989.

Nataly Brukhim, Xinyi Chen, Elad Hazan, and Shay Moran. Online agnostic boosting via regret minimization. *Advances in Neural Information Processing Systems*, 33:644–654, 2020.

Nataly Brukhim, Elad Hazan, Shay Moran, Indraneel Mukherjee, and Robert E Schapire. Multiclass boosting and the cost of weak learning. *Advances in Neural Information Processing Systems*, 34: 3057–3067, 2021.

Nataly Brukhim, Steve Hanneke, and Shay Moran. Improper multiclass boosting. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 5433–5452. PMLR, 2023.

Rich Caruana. Multitask learning. *Machine learning*, 28:41–75, 1997.

Jiecao Chen, Qin Zhang, and Yuan Zhou. Tight bounds for collaborative pac learning via multiplicative weights. *Advances in neural information processing systems*, 31, 2018.

Xi Chen, Christos Papadimitriou, and Binghui Peng. Memory bounds for continual learning. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 519–530. IEEE, 2022.

Paul Christiano, Jonathan A Kelner, Aleksander Madry, Daniel A Spielman, and Shang-Hua Teng. Electrical flows, laplacian systems, and faster approximation of maximum flow in undirected graphs. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 273–282, 2011.

Emily Diana, Wesley Gill, Michael Kearns, Krishnaram Kenthapadi, and Aaron Roth. Minimax group fairness: Algorithms and experiments. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 66–76, 2021.

John C Duchi and Hongseok Namkoong. Learning models with uniform performance via distributionally robust optimization. *The Annals of Statistics*, 49(3):1378–1406, 2021.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

Yoav Freund, Robert Schapire, and Naoki Abe. A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612, 1999.

Naveen Garg and Jochen Könemann. Faster and simpler algorithms for multicommodity flow and other fractional packing problems. *SIAM Journal on Computing*, 37(2):630–652, 2007.

Nika Haghtalab, Michael Jordan, and Eric Zhao. On-demand sampling: Learning optimally from multiple distributions. *Advances in Neural Information Processing Systems*, 35:406–419, 2022.

Steve Hanneke. The optimal sample complexity of pac learning. *The Journal of Machine Learning Research*, 17(1):1319–1333, 2016.

Ursula Hébert-Johnson, Michael Kim, Omer Reingold, and Guy Rothblum. Multicalibration: Calibration for the (computationally-identifiable) masses. In *International Conference on Machine Learning*, pages 1939–1948. PMLR, 2018.

Adam Kalai and Rocco A Servedio. Boosting in the presence of noise. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, pages 195–205, 2003.

Adam Tauman Kalai, Yishay Mansour, and Elad Verbin. On agnostic boosting and parity learning. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 629–638, 2008.

Kasper Green Larsen. Bagging is an optimal pac learner. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 450–468. PMLR, 2023.

Nick Littlestone and Manfred K Warmuth. The weighted majority algorithm. *Information and computation*, 108(2):212–261, 1994.

Yishay Mansour and David McAllester. Boosting using branching programs. *Journal of Computer and System Sciences*, 64(1):103–112, 2002.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In *International Conference on Machine Learning*, pages 4615–4625. PMLR, 2019.

Huy Nguyen and Lydia Zakynthinou. Improved algorithms for collaborative pac learning. *Advances in Neural Information Processing Systems*, 31, 2018.

Binghui Peng and Aviad Rubinstein. Near optimal memory-regret tradeoff for online learning. In *2023 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, 2023.

Binghui Peng and Fred Zhang. Online prediction in sub-linear space. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1611–1634. SIAM, 2023.

Mingda Qiao. Do outliers ruin collaboration? In *International Conference on Machine Learning*, pages 4180–4187. PMLR, 2018.

Hamed Rahimian and Sanjay Mehrotra. Distributionally robust optimization: A review. *arXiv preprint arXiv:1908.05659*, 2019.

Guy N Rothblum and Gal Yona. Multi-group agnostic pac learnability. In *International Conference on Machine Learning*, pages 9107–9115. PMLR, 2021.

Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2019.

Shiori Sagawa, Aditi Raghunathan, Pang Wei Koh, and Percy Liang. An investigation of why overparameterization exacerbates spurious correlations. In *International Conference on Machine Learning*, pages 8346–8356. PMLR, 2020.

Norbert Sauer. On the density of families of sets. *Journal of Combinatorial Theory, Series A*, 13 (1):145–147, 1972.

Robert E Schapire. The strength of weak learnability. *Machine learning*, 5:197–227, 1990.

Robert E Schapire. Explaining adaboost. In *Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik*, pages 37–52. Springer, 2013.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31, 2018.

Shubhanshu Shekhar, Greg Fields, Mohammad Ghavamzadeh, and Tara Javidi. Adaptive sampling for minimax fair classification. *Advances in Neural Information Processing Systems*, 34:24535–24544, 2021.

Saharon Shelah. A combinatorial problem; stability and order for models and theories in infinitary languages. *Pacific Journal of Mathematics*, 41(1):247–261, 1972.

Chao Tao, Qin Zhang, and Yuan Zhou. Collaborative learning with limited interaction: Tight bounds for distributed exploration in multi-armed bandits. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 126–146. IEEE, 2019.

Christopher J Tosh and Daniel Hsu. Simple and near-optimal algorithms for hidden stratification and multi-group learning. In *International Conference on Machine Learning*, pages 21633–21657. PMLR, 2022.

Leslie G Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

VN Vapnik and A Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971.

Zihan Zhang, Wenhao Zhan, Yuxin Chen, Simon S Du, and Jason D Lee. Optimal multi-distribution learning. *arXiv preprint arXiv:2312.05134*, 2023.

Appendix A. Missing proof

We first provide the proof of Lemma 7.

Proof [Proof of Lemma 7] For each $t \in [T]$, we condition on the high probability event of Lemma 6. Suppose $h^*_{\mathcal{C}} \in \mathcal{C}_{\mathcal{H}}$ has the same projection as h^* on $S_1^{(t)}$, it suffices to prove $h^*_{\mathcal{C}} \in \mathcal{H}'$. By Lemma 6, we have

$$\Pr_{x\sim\mathcal{D}^{(t)}}[h^*(x) \neq h^*_{\mathcal{C}}(x)] \leq \epsilon. \tag{9}$$

For any set $\mathcal{I} \subseteq [k]$ with $\sum_{i\in\mathcal{I}} p_t(i) \geq 1/2$, we have

$$\frac{\sum_{i\in\mathcal{I}} p_t(i)\ell_{\mathcal{D}_i}(h^*_{\mathcal{C}})}{\sum_{i\in\mathcal{I}} p_t(i)} \leq \frac{\sum_{i\in\mathcal{I}} p_t(i)\ell_{\mathcal{D}_i}(h^*) + \epsilon}{\sum_{i\in\mathcal{I}} p_t(i)} \leq \mathrm{OPT} + 2\epsilon \tag{10}$$

Here the first step follows from Eq. (9), the second step holds since $\ell_{\mathcal{D}_i}(h^*) \leq \mathrm{OPT}$ ($\forall i \in [k]$) and $\sum_{i\in\mathcal{I}} p_t(i) \geq 1/2$.

Next, we have

$$\begin{aligned}
&\Pr\left[\frac{\sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_{\mathcal{I}} \wedge h^*_{\mathcal{C}}(x_j) \neq y_j\}}{\sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_{\mathcal{I}}\}} \geq \mathrm{OPT}' + 8\epsilon\right] \\
\leq\ &\Pr\left[\frac{\sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_{\mathcal{I}} \wedge h^*_{\mathcal{C}}(x_j) \neq y_j\}}{\sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_{\mathcal{I}}\}} \geq \mathrm{OPT} + 7\epsilon\right] \\
\leq\ &\Pr\left[\sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_{\mathcal{I}}\} \leq \frac{1}{4}m_2\right] \\
&+ \Pr\left[\frac{\sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_{\mathcal{I}} \wedge h^*_{\mathcal{C}}(x_j) \neq y_j\}}{\sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_{\mathcal{I}}\}} \geq \mathrm{OPT} + 7\epsilon \mid \sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_{\mathcal{I}}\} \geq \frac{1}{4}m_2\right] \\
\leq\ &\exp(-m_2/8) + \exp(-2\cdot(m_2/4)\cdot(5\epsilon)^2) \\
\leq\ &2^{-k}\cdot\frac{\delta}{32T}.
\end{aligned}$$

The first step follows from $\mathrm{OPT} \leq \mathrm{OPT}' + \epsilon$, the third step follows from Chernoff bound, $\sum_{i\in\mathcal{I}} p_t(i) \geq 1/2$ and Eq. (10). The last step follows from the choice of $m_2 \geq \Omega(k\log(kd/\epsilon\delta)/\epsilon^2)$.

Taking a union bound over all subsets $\mathcal{I} \subseteq [k]$, we have

$$\Pr[h^* \in \mathcal{H}'] = \Pr[h^*_{\mathcal{C}} \in \mathcal{H}'] \geq 1 - 2^k\cdot 2^{-k}\cdot\frac{\delta}{32T} \geq 1 - \frac{\delta}{32T}.$$

This finishes the proof. ■

We then provide the proof of Lemma 8.

Proof [Proof of Lemma 8] For each $t \in [T]$, we condition on the high probability event of Lemma 6. For each $h \in \mathcal{C}_{\mathcal{H}}$, if there exists $h' \in \mathcal{H}$ that has the same projection as h on $S_1^{(t)}$, and there exists a subset $\mathcal{I} \subseteq [k]$ with $\sum_{i\in\mathcal{I}} p_t(i) \geq 1/2$, such that

$$\frac{\sum_{i\in\mathcal{I}} p_t(i)\ell_{\mathcal{D}_i}(h')}{\sum_{i\in\mathcal{I}} p_t(i)} \geq \mathrm{OPT} + 16\epsilon \tag{11}$$

then we prove h would be removed from $\mathcal{H}'$ with high probability.

On the same subset $\mathcal{I}$, we have

$$\frac{\sum_{i\in\mathcal{I}} p_t(i)\ell_{\mathcal{D}_i}(h)}{\sum_{i\in\mathcal{I}} p_t(i)} \geq \frac{\sum_{i\in\mathcal{I}} p_t(i)\ell_{\mathcal{D}_i}(h') - \epsilon}{\sum_{i\in\mathcal{I}} p_t(i)} \geq \mathrm{OPT} + 14\epsilon \geq \mathrm{OPT}' + 13\epsilon \tag{12}$$

where the first step holds from Lemma 6, the second step holds since $\sum_{i\in\mathcal{I}} p_t(i) \geq 1/2$, the third step follows from Eq. (11), and the last step follows from $\mathrm{OPT}' \leq \mathrm{OPT} + \epsilon$.

Now, we have

$$\begin{aligned}
&\Pr\left[\frac{\sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_\mathcal{I} \wedge h(x_j) \neq y_j\}}{\sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_\mathcal{I}\}} < \mathrm{OPT}' + 8\epsilon\right] \\
\leq\ &\Pr\left[\sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_\mathcal{I}\} < \frac{1}{4}m_2\right] \\
&+ \Pr\left[\frac{\sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_\mathcal{I} \wedge h(x_j) \neq y_j\}}{\sum_{j\in[m_2]} 1\{x_j \in \mathcal{D}_\mathcal{I}\}} < \mathrm{OPT}' + 8\epsilon \mid \sum_{j\in[n]} 1\{x_j \in \mathcal{D}_\mathcal{I}\} \geq \frac{1}{4}m_2\right] \\
\leq\ &\exp(-m_2/8) + \exp(-2\cdot(m_2/4)\cdot(5\epsilon)^2) \\
\leq\ &(kd/\epsilon^2\delta)^{-d}\cdot\frac{\delta}{32T}.
\end{aligned}$$

The second step follows from Chernoff bound, $\sum_{i\in\mathcal{I}} p_t(i) \geq 1/2$ and Eq. (12). The third step holds from the choice $m_2 \geq \Omega(d\log(kd/\epsilon\delta)/\epsilon^2)$.

Take a union bound over $\mathcal{C}_\mathcal{H}$ and note that $|C_\mathcal{H}| \leq (kd/\epsilon^2\delta)^d$ by Sauer–Shelah Lemma (see Lemma 3), we complete the proof. ■