Simple and near-optimal algorithms for hidden stratification and multi-group learning

Christopher Tosh [1] **Daniel Hsu** [2]

Abstract

Multi-group agnostic learning is a formal learning criterion that is concerned with the conditional risks of predictors within subgroups of a population. The criterion addresses recent practical concerns such as subgroup fairness and hidden stratification. This paper studies the structure of solutions to the multi-group learning problem, and provides simple and near-optimal algorithms for the learning problem.

1. Introduction

Despite its status as the de facto selection criterion for machine learning models, accuracy is an aggregate statistic that often obscures the underlying structure of mistaken predictions. Oakden-Rayner et al. (2020) recently raised this concern in the context of medical image analysis. Consider the problem of diagnosing an image as cancerous or not. Certain types of aggressive cancers may be less common than some non-aggressive types. Thus, classifiers that concentrate their errors on images of these rarer and more aggressive cancers can achieve higher overall accuracy than classifiers that spread their errors more evenly over all types of cancer. However, choosing classifiers that concentrate their errors on these rarer cancers is clearly not ideal, as it could lead to harmful misdiagnoses for those who need treatment the most. Oakden-Rayner et al. refer to this general phenomenon as the *hidden stratification* problem, where there is some latent grouping of the data domain and performance on these latent groups is just as important as performance on the entire domain.

Similar scenarios arise in areas where *fairness* is a concern (see, e.g., Hardt et al., 2016). Here, the concern is that for applications such as credit recommendation or loan approval the errors of models may be concentrated on certain demographic groups and potentially exacerbate pre-existing social disadvantages. This issue can persist even when protected class information such as race or age is not explicitly included in the model, as other features often serve as good proxies for protected class information.

The *multi-group (agnostic) learning* setting, formalized by Rothblum & Yona (2021a), is a learning-theoretic model for addressing these scenarios. This setting is specified by a collection of groups $\mathcal{G}$, where each group $g \in \mathcal{G}$ is a subset of the input space, and a set of reference predictors $\mathcal{H}$. Here, the groups in $\mathcal{G}$ can overlap in arbitrary ways, and $|\mathcal{G}|$ need not be finite. The multi-group learning objective is to find a predictor f such that, for all groups $g \in \mathcal{G}$ simultaneously, the average loss of f among examples in g is comparable with that of the best predictor $h_g \in \mathcal{H}$ specific to g. This learning objective thus pays attention to the group that is worst off with respect to $\mathcal{H}$. Note that because a good reference predictor h_g for one group g may be very poor for another group g', a successful multi-group learner may need to choose its predictor f from outside of $\mathcal{H}$.

Rothblum & Yona (2021a) obtained initial results for multi-group learning (and, in fact, study a broader class of objectives compared to what we consider here), but they leave open several theoretical questions that we address in this paper. First, while it is known that uniform convergence of empirical risks with respect to $\mathcal{H}$ is necessary and sufficient for the standard agnostic learning model, it is not clear whether the same holds for multi-group learning, let alone whether the sample complexities are equivalent. Second, Rothblum & Yona focus on finite $\mathcal{G}$, but the motivation from hidden stratification may require the use of rich and infinite families of groups in order to well-approximate the potentially unknown strata of importance. Finally, Rothblum & Yona obtained their algorithm via a blackbox reduction to a more general problem, leaving open the possibility of simpler algorithms and prediction rules with multi-group learning guarantees.

1.1. Summary of results

We introduce some notation in order to state our results. Given n i.i.d. training examples drawn from a distribution

[1]Memorial Sloan Kettering Cancer Center, New York, NY [2]Department of Computer Science, Columbia University, New York, NY. Correspondence to: Christopher Tosh <christopher.j.tosh@gmail.com>, Daniel Hsu <djhsu@cs.columbia.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

$\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, let $\#_n(g)$ for a group $g \subseteq \mathcal{X}$ denote the number of training examples (x, y) with $x \in g$. For a predictor $f \colon \mathcal{X} \to \mathcal{Z}$ and a group $g \subseteq \mathcal{X}$, let

$$L(f \mid g) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(f(x), y) \mid x \in g]$$

denote the *conditional risk of f on g*, and $\ell \colon \mathcal{Z} \times \mathcal{Y} \to [0, 1]$ is a bounded loss function.

Our first multi-group learning result is an algorithm to learn simple predictors with per-group conditional risk guarantees. Here, the class of 'simple predictors' we consider is the collection of decision lists in which internal (decision) nodes are associated with membership tests for groups from $\mathcal{G}$, and leaf (prediction) nodes are associated with reference predictors from $\mathcal{H}$.

Theorem 4 (Informal)**.** *There is an algorithm $\mathcal{A}$ such that the following holds for any hypothesis set $\mathcal{H}$ and set of groups $\mathcal{G}$. Given n i.i.d. training examples from $\mathcal{D}$, $\mathcal{A}$ produces a decision list f such that, with high probability,*

$$L(f \mid g) \;\leq\; \inf_{h\in\mathcal{H}} L(h \mid g) + O\left(\left(\frac{\log|\mathcal{H}||\mathcal{G}|}{\gamma_n \cdot \#_n(g)}\right)^{1/3}\right)$$

for all $g \in \mathcal{G}$, where $\gamma_n := \min_{g\in\mathcal{G}} \#_n(g)/n$ is the minimum empirical probability mass among groups in $\mathcal{G}$.[1]

When $\mathcal{H}$ and $\mathcal{G}$ are infinite, a version of Theorem 4 also holds when the complexities of $\mathcal{H}$ and $\mathcal{G}$ are appropriately bounded; see Theorem 6 for a formal statement.

Though the algorithm and resulting predictors of Theorem 4 and Theorem 6 are quite simple, the per-group excess error rates are suboptimal. Statistical learning theory suggests that if we knew *a priori* which group g we would be tested on, empirical risk minimization (ERM) on the i.i.d. training examples restricted to g would lead to an excess risk of $O(\sqrt{\log(|\mathcal{H}|)/\#_n(g)})$ in the finite setting (Shalev-Shwartz & Ben-David, 2014). The rates in Theorem 4 have two undesirable properties when compared to this theoretical rate: they have a worse exponent, and they depend on the minimum probability mass among all the groups.

Our next result shows that the theoretical rate suggested by per-group ERM can be achieved in the multi-group learning setting, modulo a logarithmic factor in $|\mathcal{G}|$.

Theorem 8 (Informal)**.** *There is an algorithm $\mathcal{A}$ such that the following holds for any finite hypothesis set $\mathcal{H}$ and finite set of groups $\mathcal{G}$. Given n i.i.d. training examples from $\mathcal{D}$, $\mathcal{A}$ produces a randomized predictor f such that, with high probability,*

$$L(f \mid g) \;\leq\; \inf_{h\in\mathcal{H}} L(h \mid g) + O\left(\left(\frac{\log(|\mathcal{H}||\mathcal{G}|)}{\#_n(g)}\right)^{1/2}\right)$$

for all $g \in \mathcal{G}$.

The improved rates of Theorem 8 come at the expense of increased complexity in both learning procedure and prediction algorithm.

In our final result, we show that in the *group-realizable* setting, where each group has a corresponding perfect predictor, this trade-off between optimal rates and simple algorithms is unnecessary.

Theorem 10 (Informal)**.** *There is an algorithm $\mathcal{A}$ such that the following holds for any finite hypothesis set $\mathcal{H}$ and finite set of groups $\mathcal{G}$ such that for all $g \in \mathcal{G}$ there exists an $h \in \mathcal{H}$ satisfying $L(h \mid g) = 0$. Given n i.i.d. training examples from $\mathcal{D}$, $\mathcal{A}$ produces a predictor f such that, with high probability,*

$$L(f \mid g) \;\leq\; O\left(\frac{\log(|\mathcal{H}||\mathcal{G}|)}{\#_n(g)}\right) \quad \text{for all } g \in \mathcal{G}.$$

1.2. Related work

There are a number of areas of active research that intersect with the present paper.

Distributionally robust optimization. The field of distributionally robust optimization, or DRO, focuses on the problem of optimizing some cost function such that the cost of the solution is robust to perturbations in the problem instance (Ben-Tal et al., 2009). In the context of machine learning and statistics, the idea is to use data from some training distribution to learn a predictor that will perform well when used on a worst-case choice of test distribution from some known class. In this field, the class of test distributions is typically a small perturbation of the training distribution (Bertsimas et al., 2018; Duchi et al., 2021; 2020).

A special case of DRO of particular relevance to the current work is group-wise DRO. Here, the class of test distributions is fixed to be a finite set of distributions, and the goal is to find a predictor whose worst-case conditional risk over any of these test distributions is minimized. Several solutions for this problem have been proposed, including mirror descent (Mohri et al., 2019), group-wise regularization (Sagawa et al., 2020a), group-wise sample reweighting (Sagawa et al., 2020b), and two-stage training procedures (Liu et al., 2021).

Group and individual fairness. As discussed above, prediction mistakes committed by machine-learned models can lead to widespread social harms, particularly if they are concentrated on disadvantaged social groups. To address this, a recent, but large, body of work has emerged to define fairness criteria of predictors and develop machine learning methods that meet these criteria. While many criteria for fair predictors abound, they can typically be broken into two

[1] Here and in the rest of the paper, big-O notation is only used to conceal constants. So, $a = O(b)$ should be read as 'There exists an absolute constant $C > 0$ such that $a \leq Cb$.'

categories. The first of these is group-wise fairness (Hardt et al., 2016; Agarwal et al., 2018; Donini et al., 2018) in which a classifier is trained to equalize some notion of harm, such as false-negative predictions, or benefit, such as true-positive predictions, across predefined groups. The second category of fairness notion is individual fairness (Dwork et al., 2012; Dwork & Ilvento, 2018), in which there is a distance function or notion of similarity among points, and the objective is to give similar predictions for similar points.

Of particular relevance to the present paper is the work of Kearns et al. (2018), which studied group fairness for a large (potentially infinite) number of (potentially overlapping) groups. The authors assumed boundedness of the VC-dimensions for both the hypothesis class and the class of groups, as well as access to an oracle for solving certain cost-sensitive classification problems. Under these assumptions, they provided an algorithm that solves for a convex combination of hypotheses from the hypothesis class that respects certain notions of fairness for all groups and is competitive with the best such fair convex combination.

Online fairness. There is a growing body of work at the intersection of online learning and fairness (Gillen et al., 2018; Noarov et al., 2021; Gupta et al., 2021). The most relevant to the present paper is the work of Blum & Lykouris (2020), which studies an online version of multi-group learning where the goal is to achieve low regret on each of the groups simultaneously. That work gives a reduction to sleeping experts, showing that for a particular choice of experts, the regret guarantees of the sleeping experts algorithm directly translates to a per-group regret guarantee. Inspired by this observation, in Section 4 we show that the offline multi-group learning problem can also be reduced to the sleeping experts problem. The online-to-batch conversion argument we use requires some care, however, since there are multiple objectives (one for each group) that need to be satisfied, in contrast to standard online-to-batch settings where only a single objective needs to be met.

Multicalibration, multiaccuracy, and outcome distinguishability. Also motivated by fairness considerations, a recent line of work is centered on calibrating predictions to be unbiased on a collection of groups or subpopulations.

Given a class of groups $\mathcal{G}$, *multiaccuracy* requires that the expectation of a predictor is close to expectation of the outcome y when conditioned on any $g \in \mathcal{G}$ (Hébert-Johnson et al., 2018; Diana et al., 2021). Kim et al. (2019) showed that for an appropriate choice of groups, multiaccuracy implies a type of multi-group learnability result. Unfortunately, the upper bound they show for the per-group error rate is only non-trivial when the best rate achievable at that group is small. In Appendix E, we show that this looseness is inherent to this type of multiaccuracy reduction.

Multicalibration is a more stringent notion than multiaccuracy that requires these expectations to be close when conditioned both on the group and the value of the prediction. Even more stringent is *outcome indistinguishability*, a family of criteria that requires the predictions of a function $f : \mathcal{X} \to [0, 1]$ to be indistinguishable against the true probabilities of positive versus negative outcomes with respect to classes of distinguishing algorithms with varying levels of access to the underlying distribution (Dwork et al., 2021).

Using a reduction to the outcome indistinguishability framework, Rothblum & Yona (2021a) provided an algorithm with a multi-group learning guarantee. Specifically, they showed that for a given finite hypothesis class $\mathcal{H}$ and finite set of groups $\mathcal{G}$, one can produce a predictor $f : \mathcal{X} \to \mathbb{R}$ such that $L(f \mid g) \leq \min_{h \in \mathcal{H}} L(h \mid g) + \epsilon$ for all $g \in \mathcal{G}$ with probability $1 - \delta$.[2] The sample complexity of their approach is $O\left(\frac{m_{\mathcal{H}}(\epsilon,\delta)^4}{\delta^4\gamma} \log \frac{|\mathcal{H}||\mathcal{G}|}{\epsilon}\right)$, where $m_{\mathcal{H}}(\epsilon, \delta)$ is the sample complexity of agnostically learning a classifier $h \in \mathcal{H}$ with excess error ϵ and failure probability δ (Rothblum & Yona, 2021b). Standard ERM arguments give us $m_{\mathcal{H}}(\epsilon, \delta) = O\left(\frac{1}{\epsilon^2} \log \frac{|\mathcal{H}|}{\delta}\right)$, leading to an overall sample complexity of $O\left(\frac{1}{\epsilon^8\delta^4\gamma} \operatorname{poly} \log \frac{|\mathcal{H}||\mathcal{G}|}{\epsilon\delta}\right)$.

In independent and concurrent work, Globus-Harris et al. (2022) also considered a multi-group learning setup under the framework of 'bias bug bounties.' In this setting, there is some deployed model and outsiders are incentivized to find groups on which the model does worse than Bayes optimal. If such a group is found, one can submit the group as well as a certificate of suboptimality to receive a bounty, and the model will be updated to improve performance on the affected group. Interestingly, the algorithm developed by Globus-Harris et al. to update their model (which they call LISTUPDATE) is equivalent to one of the algorithms presented in the present paper (PREPEND). Beyond the development of the concept of bias bug bounties, Globus-Harris et al. are primarily concerned with the computational complexity of finding such bounties, whereas the present work is focused on the statistical sample complexity of multi-group learning.

Hidden stratification. The *hidden stratification* problem refers to settings where there are meaningful subgroups or divisions of the data space, but they are unknown ahead of time. The fear, as illustrated by Oakden-Rayner et al. (2020), is that prediction errors can be concentrated on some important subgroup and lead to real-world harms. Sohoni et al. (2020) proposed addressing this problem by clustering

[2] In fact, their results apply to more general types of objectives that need not be *decomposable*; see (Rothblum & Yona, 2021a, Section 2) for details. In this paper, we focus only on objectives of the form $L(f \mid g)$, which are decomposable in their sense.

a dataset and solving a DRO problem as if the resulting clusters were the true known subgroups.

Transfer learning and covariate shift. Broadly speaking, *transfer learning* studies the problem of learning a predictor given many samples from a source distribution P and relatively few (or perhaps no) samples from a target distribution Q, where the predictor will ultimately be evaluated on Q. The results in this area depend on what is allowed to change between P and Q. Some works study the setting of *covariate shift*, where only the covariate distribution is allowed to change (Shimodaira, 2000; Zadrozny, 2004; Cortes et al., 2010; Kpotufe & Martinet, 2018; Hopkins et al., 2021). Others focus on *label shift*, where only the marginal label distribution changes, leaving the class conditional distributions unchanged (Azizzadenesheli et al., 2018). Finally, in the most general setting, P and Q may differ in both covariates and labels (Ben-David et al., 2010).

Of these transfer learning settings, the covariate shift framework most closely resembles the multi-group learning setting. However, the two key differences are that (1) the transfer learning setting is typically concerned with performance on a single target distribution and (2) the target distributions that arise in multi-group learning are restricted to conditional distributions of the source distribution. Importantly, in the multi-group learning setup, the target distributions never have support that is outside of the source distribution. Because of this restriction, we can achieve a much stronger performance guarantee compared to what is possible in the setting of general covariate shift.

One work that deviates from the single-target distribution framework is that of Hopkins et al. (2021), whose notion of covariate shift error is the maximum error over a set of target distributions, similar to the multi-group learning setup. However, in the covariate shift setting of Hopkins et al. (2021), there is only a single such benchmark hypothesis for all possible shifts, whereas in our multi-group setting, the benchmark hypothesis is allowed to depend on the group. Thus the guarantees in the setting of Hopkins et al. (2021) are not comparable to those in our setting.

Boosting. *Boosting* is a classical machine learning technique for converting weak learners, i.e., learners that output predictors that are only marginally more accurate than random guessing, into strong learners, i.e., learners that output predictors with very high accuracy (Schapire, 1990; Freund, 1995). Many of the algorithms for achieving multiaccuracy, multicalibration, and outcome indistinguishability can be viewed as boosting algorithms (Hébert-Johnson et al., 2018; Kim et al., 2019; Dwork et al., 2021). For instance, the algorithm of Kim et al. (2019) is based on the boosting algorithm of Trevisan et al. (2009). One of the algorithms proposed in this paper, PREPEND, is no exception here and may also be viewed as a boosting algorithm.

1.3. Paper outline

The remainder of the paper is organized as follows. In Section 2, we formalize the multi-group learning setting. In Section 3 we present a simple algorithm for the multi-group learning problem that achieves a suboptimal generalization bound. In Section 4, we give a reduction to the online sleeping experts problem. The resulting algorithm achieves the correct generalization rate, though this comes at the expense of a significantly more complicated learning algorithm. Finally, in Section 5, we consider a setting in which each group has a corresponding perfect classifier. Here we show that there is no need to trade off simplicity and optimality: a relatively simple algorithm achieves the optimal per-group performance guarantee.

All proofs are presented in the appendix.

2. Setting and notation

Let $\mathcal{X}$ denote an input space, $\mathcal{Y}$ denote a label space, and $\mathcal{Z}$ denote a prediction space. Let $\mathcal{D}$ denote a distribution over $\mathcal{X} \times \mathcal{Y}$. Throughout, $\mathcal{H} \subseteq \{h : \mathcal{X} \to \mathcal{Z}\}$ denotes a (benchmark) hypothesis class. A group g is a subset of the space $\mathcal{X}$. We overload notation by identifying a group $g \subseteq \mathcal{X}$ with the binary function $g : \mathcal{X} \to \{0, 1\}$ that indicates membership in g. We denote the set of groups of interest by $\mathcal{G}$, and let $P(g) := \mathbb{E}_{(x,y)\sim\mathcal{D}}[g(x)]$ for any group g. Let $\ell \colon \mathcal{Z} \times \mathcal{Y} \to [0, 1]$ be a bounded loss function, and for a predictor $f \colon \mathcal{X} \to \mathcal{Z}$, the conditional risk of f given g is

$$L(f \mid g) := \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(f(x), y) \mid x \in g].$$

For an i.i.d. sample $(x_1, y_1), \ldots, (x_n, y_n) \sim \mathcal{D}$, we define the following empirical quantities: let $\#_n(g) := \sum_{i=1}^n g(x_i)$, $P_n(g) := \#_n(g)/n$, and

$$L_n(f \mid g) := \frac{1}{\#_n(g)} \sum_{i=1}^n g(x_i)\ell(f(x_i), y_i).$$

The (unconditional) risk and empirical risk of f are $L(f) := \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(f(x), y)]$ and $L_n(f) := \frac{1}{n}\sum_{i=1}^n \ell(f(x_i), y_i)$.

2.1. Multi-group agnostic learning

At a high level, the objective in the multi-group (agnostic) learning setting is to find a predictor f such that the conditional risk is not much larger than $\inf_{h\in\mathcal{H}} L(h \mid g)$ for all $g \in \mathcal{G}$. This setting was formalized by Rothblum & Yona (2021a); they require, for a given $\epsilon > 0$, that the excess conditional risks be uniformly small over all groups $g \in \mathcal{G}$:

$$L(f \mid g) \;\le\; \inf_{h\in\mathcal{H}} L(h \mid g) + \epsilon \quad \text{for all } g \in \mathcal{G}. \tag{1}$$

Note that the best hypothesis in the benchmark class $h_g \in \mathcal{H}$ for a particular group g may not be the same as that for a different group g'. Indeed, there may be no single $h \in \mathcal{H}$ that has low conditional risk for all groups in $\mathcal{G}$ simultaneously. Hence, a learner may typically need to choose a predictor f from outside of $\mathcal{H}$.

We will give non-uniform bounds on the excess conditional risks (discussed below), where ϵ is replaced by a quantity $\epsilon_n(g)$ depending on both the size n training set and the specific group $g \in \mathcal{G}$ in question:

$$L(f \mid g) \;\leq\; \inf_{h \in \mathcal{H}} L(h \mid g) + \epsilon_n(g) \quad \text{for all } g \in \mathcal{G}. \quad (2)$$

The quantity $\epsilon_n(g)$ will be a decreasing function of $\#_n(g)$, and it will be straightforward to determine a minimum sample size n (in terms of a prescribed $\epsilon > 0$) such that Eq. (2) implies Eq. (1).

2.2. Convergence of conditional risks

For a class of $\{0,1\}$-valued functions $\mathcal{F}$ defined over a domain $\mathcal{X}$, the k-th shattering coefficient, is given by

$$\Pi_k(\mathcal{F}) := \max_{x_1,\ldots,x_k \in \mathcal{X}} \left|\{(f(x_1),\ldots,f(x_k)) \,:\, f \in \mathcal{F}\}\right|.$$

For a class of real-valued functions $\mathcal{F}$ defined over a domain $\mathcal{X}$, the thresholded class is given by

$$\mathcal{F}_{\text{thresh}} \;:=\; \{x \mapsto \mathbb{1}[f(x) > \tau] \,:\, f \in \mathcal{F}, \tau \in \mathbb{R}\}.$$

Finally, for a hypothesis class $\mathcal{H}$ and a loss function $\ell\colon \mathcal{Z} \times \mathcal{Y} \to [0,1]$, the loss-composed class is

$$\ell \circ \mathcal{H} \;:=\; \{(x,y) \mapsto \ell(h(x),y) \,:\, h \in \mathcal{H}\}.$$

The following theorem shows that the empirical conditional risks converge uniformly to their population counterparts. This can be seen as a generalization of a result by Balsubramani et al. (2019), which demonstrated universal convergence of empirical conditional probabilities.

Theorem 1. *Let $\mathcal{H}$ be a hypothesis class, let $\mathcal{G}$ be a set of groups, and let $\ell\colon \mathcal{Z} \times \mathcal{Y} \to [0,1]$ be a loss function. With probability at least $1-\delta$,*

$$|L(h \mid g) - L_n(h \mid g)| \;\leq\; 9\sqrt{\frac{D}{\#_n(g)}} \quad \forall (h,g) \in \mathcal{H} \times \mathcal{G},$$

where $D = 2\log\left(\Pi_{2n}((\ell \circ \mathcal{H})_{\text{thresh}})\Pi_{2n}(\mathcal{G})\right) + \log(8/\delta)$.

In the standard agnostic binary classification setting, it is known that, in general, the best achievable error rate of a learning algorithm is on the order of the uniform convergence rate of the empirical risks of the entire hypothesis class (Shalev-Shwartz & Ben-David, 2014, Chapter 6). This can be seen as a statistical equivalence between learning and estimation. Theorem 1 raises the question of whether such an equivalence can also be established in the multi-group learning setting. In this work, we make partial progress towards establishing such an equivalence, providing a learning algorithm whose per-group error rate enjoys the same upper bound as the convergence rates in Theorem 1.

Algorithm 1 PREPEND

input Groups $\mathcal{G}$, hypothesis class $\mathcal{H}$, i.i.d. examples $(x_1,y_1),\ldots,(x_n,y_n)$ from $\mathcal{D}$, error bound $\epsilon_n\colon \mathcal{G} \to \mathbb{R}_+$.
output Decision list $f_T \in \texttt{DL}_T[\mathcal{G};\mathcal{H}]$.
  Compute $h_0 \in \operatorname{argmin}_{h \in \mathcal{H}} L_n(h)$
  Set $f_0 = [h_0] \in \texttt{DL}_0[\mathcal{G};\mathcal{H}]$.
  for $t = 0, 1, \ldots,$ **do**
    Compute

$$(g_{t+1}, h_{t+1}) \in \operatorname*{argmax}_{(g,h) \in \mathcal{G} \times \mathcal{H}} L_n(f_t \mid g) - L_n(h \mid g) - \epsilon_n(g).$$

    if $L_n(f_t \mid g_{t+1}) - L_n(h_{t+1} \mid g_{t+1}) \geq \epsilon_n(g_{t+1})$ **then**
      Prepend (g_{t+1}, h_{t+1}) to f_t to obtain

$$f_{t+1} := [g_{t+1}, h_{t+1}, g_t, h_t, \ldots, g_1, h_1, h_0].$$

    else
      return f_t.
    end if
  end for

3. A simple multi-group learning algorithm

In this section we will show that there is a particularly simple class of predictors for solving the multi-group learning problem: decision lists in which internal (decision) nodes are associated with functions from $\mathcal{G}$, and leaf (prediction) nodes are associated with functions from $\mathcal{H}$. We denote the set of such decision lists of length t by $\texttt{DL}_t[\mathcal{G};\mathcal{H}]$. The function computed by $f_t = [g_t, h_t, g_{t-1}, h_{t-1}, \ldots, g_1, h_1, h_0] \in \texttt{DL}_t[\mathcal{G};\mathcal{H}]$ is as follows: upon input $x \in \mathcal{X}$,

if $g_t(x) = 1$ **then** return $h_t(x)$ **else if** $g_{t-1}(x) = 1$ **then** return $h_{t-1}(x)$ **else if** $\cdots$ **else** return $h_0(x)$.

This computation can be recursively specified as

$$f_t(x) \;=\; \begin{cases} h_t(x) & \text{if } g_t(x) = 1 \\ f_{t-1}(x) & \text{if } g_t(x) = 0 \end{cases}$$

where $f_{t-1} = [g_{t-1}, h_{t-1}, \ldots, g_1, h_1, h_0] \in \texttt{DL}_{t-1}[\mathcal{G};\mathcal{H}]$. (We identify $\texttt{DL}_0[\mathcal{G};\mathcal{H}]$ with $\mathcal{H}$.)

We propose a simple algorithm, called PREPEND (Algorithm 1), for learning these decision lists. PREPEND proceeds in rounds, maintaining a current decision list

$f_t \in \texttt{DL}_t[\mathcal{G};\mathcal{H}]$. At each round, it searches for a group $g_{t+1} \in \mathcal{G}$ and a hypothesis $h_{t+1} \in \mathcal{H}$ that witnesses an empirical violation of Eq. (2). If such a violation is found, f_t is updated to f_{t+1} by prepending the pair (h_{t+1}, g_{t+1}) to the front of f_t. If no violation is found, then we claim that f_t is good enough, and terminate.

We first bound the number of iterations executed by Algorithm 1 before it terminates.

Lemma 2. *Suppose that every $g \in \mathcal{G}$ satisfies $P_n(g) \cdot \epsilon_n(g) \geq \epsilon_o$ and say that $\min_{h\in\mathcal{H}} L_n(h) \leq \alpha$. Then Algorithm 1 terminates after at most $t \leq \alpha/\epsilon_o$ rounds and outputs a predictor $f_t \in \texttt{DL}_t[\mathcal{G};\mathcal{H}]$ such that*

$$L_n(f_t \mid g) \leq \inf_{h\in\mathcal{H}} L_n(h \mid g) + \epsilon_n(g) \quad \text{for all } g \in \mathcal{G}.$$

3.1. Sample complexity

The key step in bounding the sample complexity of Algorithm 1 is in controlling the complexity of $\texttt{DL}_T[\mathcal{G};\mathcal{H}]$, whereupon Theorem 1 can be applied. To see how this is done, consider the case where $|\mathcal{G}|$ and $|\mathcal{H}|$ are finite. In this setting, there are T decision nodes, each of which can be chosen from $\mathcal{G}$, and there are $T+1$ prediction nodes chosen from $\mathcal{H}$. Thus, $|\texttt{DL}_T[\mathcal{G};\mathcal{H}]| \leq |\mathcal{G}|^T|\mathcal{H}|^{T+1}$.

To apply this observation, we first note that for any $f = [g_T, h_T, \ldots, g_1, h_1, h_0] \in \texttt{DL}_T[\mathcal{G};\mathcal{H}]$, if there are rounds $t < t'$ such that $g_t = g_{t'}$, then f is functionally equivalent to $f' \in \texttt{DL}_{T-1}[\mathcal{G};\mathcal{H}]$ where f' simply removes the occurrence of h_t, g_t in f. Thus, when the number of groups is finite, we can always pretend as if $\texttt{DL}_T[\mathcal{G};\mathcal{H}]$ is the set of decision lists of length *exactly* $|\mathcal{G}|$. The next result follows immediately.

Proposition 3. *Suppose $|\mathcal{G}|$ and $|\mathcal{H}|$ are finite. The following holds with probability at least $1-\delta$. If Algorithm 1 is run until convergence, it will terminate with a predictor f that satisfies*

$$L(f \mid g) \leq \min_{h\in\mathcal{H}} L(h \mid g) + \epsilon_n(g) + O\left(\sqrt{\frac{|\mathcal{G}|\log(|\mathcal{H}||\mathcal{G}|) + \log(1/\delta)}{\#_n(g)}}\right)$$

for all $g \in \mathcal{G}$.

Proposition 3 suggests that when $|\mathcal{G}|$ is small, a reasonable approach is to take

$$\epsilon_n(g) = O\left(\sqrt{\frac{|\mathcal{G}|\log(|\mathcal{H}||\mathcal{G}|) + \log(1/\delta)}{\#_n(g)}}\right),$$

in which case we will terminate with a predictor whose excess conditional error on any group g is $O(\epsilon_n(g))$. Thus, when the number of groups is small, estimation error and learning error are within a factor of $\sqrt{|\mathcal{G}|}$.

If the number of groups is very large, or infinite, Proposition 3 may be vacuous. However, if we have a lower bound on the empirical mass of any group (or perhaps restrict ourselves to such groups), then we can give a result that remains useful. To do so, we introduce the notation

$$\mathcal{G}_{n,\gamma} = \{g \in \mathcal{G} : \#_n(g) \geq \gamma n\}.$$

Given this notation, we have the following result.

Theorem 4. *Suppose that $\mathcal{H}$ and $\mathcal{G}$ are finite, and $\gamma > 0$ is given. There is a setting of $\epsilon_n(\cdot)$ such that the following holds. If Algorithm 1 is run with groups $\mathcal{G}_{n,\gamma}$, then with probability $1-\delta$, it terminates with a predictor f satisfying*

$$L(f \mid g) \leq \min_{h\in\mathcal{H}} L(h \mid g) + O\left(\sqrt[3]{\frac{K/\gamma}{\#_n(g)}} + \sqrt{\frac{\log(1/\delta)}{\#_n(g)}}\right)$$

for all $g \in \mathcal{G}_{n,\gamma}$, where $K := \log(|\mathcal{G}||\mathcal{H}|)$. In addition, with probability $1-\delta$, all $g \in \mathcal{G}$ satisfying $P(g) \geq \gamma + \sqrt{\frac{\log(|\mathcal{G}|/\delta)}{n}}$ will lie in $\mathcal{G}_{n,\gamma}$.

Theorem 4 implies the following sample complexity to achieve the error bound of the type in Eq. (1).

Corollary 5. *There is an absolute constant $c > 0$ such that the following holds. Suppose that $\mathcal{H}$ and $\mathcal{G}$ are finite, and $\gamma := \min_{g\in\mathcal{G}} P(g)$. There is a setting of $\epsilon_n(\cdot)$ such that the following holds. If*

$$n \geq \frac{c}{\epsilon^3\gamma^2}\log\left(\frac{|\mathcal{G}||\mathcal{H}|}{\delta}\right)$$

then with probability $1-\delta$, Algorithm 1 terminates with a predictor f satisfying

$$L(f \mid g) \leq \min_{h\in\mathcal{H}} L(h \mid g) + \epsilon \quad \text{for all } g \in \mathcal{G}.$$

PREPEND can also handle the setting where both the number of groups and the number of hypotheses are infinite, so long as the pseudo-dimension of $\ell \circ \mathcal{H}$ and the VC-dimension of $\mathcal{G}$ are bounded.

Theorem 6. *Suppose that both the pseudo-dimension of $\ell \circ \mathcal{H}$ and the VC-dimension of $\mathcal{G}$ are bounded above by d, and $\gamma > 0$ is given. There is a setting of $\epsilon_n(\cdot)$ such that the following holds. If Algorithm 1 is run with groups $\mathcal{G}_{n,\gamma}$, then with probability $1-\delta$, it returns a predictor f satisfying*

$$L(f \mid g) \leq \min_{h\in\mathcal{H}} L(h \mid g) + O\left(\sqrt[3]{\frac{d\log n}{\gamma\#_n(g)}} + \sqrt{\frac{\log(1/\delta)}{\#_n(g)}}\right)$$

for all $g \in \mathcal{G}_{n,\gamma}$. Moreover, with probability $1-\delta$, all $g \in \mathcal{G}$ satisfying $P(g) \geq \gamma + \sqrt{\frac{d\log(2n)+\log(4/\delta)}{n}}$ lie in $\mathcal{G}_{n,\gamma}$.

3.2. Inadmissibility of evaluation functions

One interpretation of decision lists in the class $\texttt{DL}_t[\mathcal{G};\mathcal{H}]$ is that they correspond to orderings of the set $\mathcal{G}\times\mathcal{H}$. That is, for any $f\in\texttt{DL}_t[\mathcal{G};\mathcal{H}]$ there is a corresponding ordering $(g_1,h_1),(g_2,h_2),\ldots$ where for a given $x\in\mathcal{X}$, we first find

$$i(x) \;=\; \min\{i \,:\, g_i(x)=1\}$$

and classify according to $h_{i(x)}(x)$. Given this alternate perspective, one can ask whether it is possible to calculate such orderings directly. We will show a negative result here for the approach of evaluation functions.

An evaluation function $s:\mathcal{H}\times\mathcal{G}\times(X\times\mathcal{Y})^*\to\mathbb{R}$ takes as input $h\in\mathcal{H}$, $g\in\mathcal{G}$ and a sample $(x_1,y_1),\ldots,(x_n,y_n)$ and outputs a real number as a score. The ordering induced by an evaluation function is then simply the ordering of the corresponding scores, with ties being broken by some (possibly randomized) rule. We say that s is an order-1 evaluation function if $s(h,g,(x_1,y_1),\ldots,(x_n,y_n))$ is only a function of n, $P_n(g)$ and $L_n(h\mid g)$.

By Theorem 1, $P_n(g)$ and $L_n(h\mid g)$ converge to their expectations as n grows to infinity. Thus, in the limit, an order-1 evaluation function is a function of $P(g)$ and $L(h\mid g)$. Unfortunately, there exist two scenarios where these statistics are identical but no decision list solves the multi-group learning problem for both scenarios simultaneously.

Proposition 7. *There exist $\mathcal{H}=\{h_1,h_2\}$, $\mathcal{G}=\{g_1,g_2\}$, and distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ such that the following holds.*

- *$P_{\mathcal{D}_1}(g)=P_{\mathcal{D}_2}(g)$ and $L_{\mathcal{D}_1}(h\mid g)=L_{\mathcal{D}_2}(h\mid g)$ for all $h\in\mathcal{H}$, $g\in\mathcal{G}$.*
- *For any decision list $f\in\texttt{DL}_t[\mathcal{G};\mathcal{H}]$, there exists an $i\in\{1,2\}$ and $g\in\mathcal{G}$ such that*

$$L_{\mathcal{D}_i}(f\mid g) \;\geq\; \min_{h\in\mathcal{H}} L_{\mathcal{D}_i}(h\mid g)+\frac{1}{8}.$$

Here, the subscript $\mathcal{D}_i$ denotes taking probabilities with respect to $\mathcal{D}_i$.

4. A reduction to sleeping experts

In this section, we will show that the rate suggested by Theorem 1 is achievable via a reduction to the sleeping experts problem, similar to the result of Blum & Lykouris (2020). The sleeping experts problem is an online expert aggregation problem such that during every round, some experts are 'awake' and some are 'asleep' (Blum, 1997; Freund et al., 1997). The goal for a learner in this setting is to achieve low regret against every expert on those rounds in which it was awake.

To reduce the multi-group learning problem to a sleeping experts problem, we create an expert for every pair $(h,g)\in\mathcal{H}\times\mathcal{G}$. In each round t, we feed an example x_t and say that expert (h,g) is awake if and only if $g(x_t)=1$, in which case the expert prediction is $h(x_t)$ and the revealed loss is $\ell(h(x_t),y_t)$. Formally, the reduction looks as follows:

Algorithm 2 Reduction to sleeping experts

input Groups $\mathcal{G}$, hypothesis class $\mathcal{H}$, $2n$ i.i.d. examples $(x_1,y_1),\ldots,(x_n,y_n),(x'_1,y'_1),\ldots,(x'_n,y'_n)$ from $\mathcal{D}$.
output Randomized predictor Q.
Run MLC-HEDGE on $(x_1,y_1),\ldots,(x_n,y_n)$ using experts $\mathcal{H}\times\mathcal{G}$ with uniform initial probabilities, and per-expert learning rates $\eta_{h,g}=\min\Big\{\sqrt{\frac{\log(|\mathcal{H}||\mathcal{G}|)}{\sum_{i=i}^{n}g(x'_i)}},1\Big\}$. Let $p_1,\ldots,p_n$ be the internal hypotheses of MLC-HEDGE.
output $Q=$ uniform distribution over $p_1,\ldots,p_n$.

For $t=1,2,\ldots,n$:

- Input x_t is revealed.
- If $g(x_t)=1$, then expert (h,g) is awake and predicts $h(x_t)$. Otherwise, (h,g) is asleep.
- The learner produces a distribution p_t over the experts such that $\sum_{h,g}g(x_t)p_t(h,g)=1$.
- Label y_t is revealed, and the learner suffers loss $\hat{\ell}_t=\sum_{h,g}p_t(h,g)g(x_t)\ell(h(x_t),y_t)$.

There are several suitable algorithms in the literature for the sleeping experts problem. Most convenient for our purposes is MLC-HEDGE, originally proposed by Blum & Mansour (2007) and further analyzed by Gaillard et al. (2014). In Appendix C, we present MLC-HEDGE and state the learning guarantees proven by Gaillard et al. (2014).

To convert this online learner into a batch learning algorithm, we follow the strategy of Helmbold & Warmuth (1995). We do this by keeping track of the internal hypotheses of the online learner. That is, let $p_t(\cdot\,;x)$ be the distribution that the learner at time t would produce if fed the example x. Given these internal hypotheses, the final predictor is as follows: on input x, draw t uniformly at random from $\{1,\ldots,n\}$, draw (h,g) from $p_t(\cdot\,;x)$, and predict $h(x)$.

For a collection of internal hypotheses $p_1,\ldots,p_n$ and a distribution Q over such hypotheses, we use the notation

$$L(p_t\mid g) \;:=\; \mathbb{E}_{(x,y)}\left[\mathbb{E}_{(\tilde{h},\tilde{g})\sim p_t(\cdot\,;x)}\left[\ell\left(\tilde{h}(x),y\right)\right]\mid g\right]$$
$$L(Q\mid g) \;:=\; \mathbb{E}_{p_t\sim Q}\left[L(p_t\mid g)\right].$$

To run MLC-HEDGE, we need to specify an initial probability distribution over the experts and a set of expert-specific learning rates. Our initial distribution is uniform distribution over the experts. For the learning rates, we use half the

Algorithm 3 Consistent majority algorithm

input Groups $\mathcal{G}$, binary classifiers $\mathcal{H}$, n i.i.d. examples from a group-realizable distribution.
output Group-conditional majority vote predictor f.
For each $g \in \mathcal{G}$, let $\hat{h}_g \in \operatorname{argmin}_{h \in \mathcal{H}} L_n(h \mid g)$.
output Predictor $x \mapsto \operatorname{sign}\left(\sum_{g \in \mathcal{G}} g(x) \hat{h}_g(x)\right)$.

sample to estimate the empirical probability masses of each group and set the learning rates to optimize an upper bound on the per-expert regret (Gaillard et al., 2014, Theorem 16). Algorithm 2 presents the full method.

Theorem 8. *Let Q be the randomized predictor returned by Algorithm 2. Then*

$$L(Q \mid g) \;\leq\; \min_{h \in \mathcal{H}} L(h \mid g) + O\left(\sqrt{\frac{\log(|\mathcal{G}||\mathcal{H}|/\delta)}{\#_n(g)}}\right)$$

for all $g \in \mathcal{G}$ with probability at least $1 - \delta$.

Theorem 8 implies the following corollary.

Corollary 9. *There is an absolute constant $c > 0$ such that the following holds. Suppose $P(g) \geq \gamma > 0$ for all $g \in \mathcal{G}$. Let Q be the randomized predictor returned by Algorithm 2. If*

$$n \;\geq\; \frac{c}{\epsilon^2 \gamma} \log \frac{|\mathcal{G}||\mathcal{H}|}{\delta},$$

then with probability at least $1 - \delta$,

$$L(Q \mid g) \;\leq\; \min_{h \in \mathcal{H}} L(h \mid g) + \epsilon \quad \textit{for all } g \in \mathcal{G}.$$

The dependence on $\log |\mathcal{H}|$ in Theorem 8 and Corollary 9 can be replaced by the pseudo-dimension of $\ell \circ \mathcal{H}$ (up to a $\log n$ factor) using a slight modification to Algorithm 2. Simply use half of the training data to find, for each $g \in \mathcal{G}$, a hypothesis $\hat{h}_g \in \mathcal{H}$ satisfying

$$L(\hat{h}_g \mid g) \leq \inf_{h \in \mathcal{H}} L(h \mid g) + O\left(\sqrt{\frac{d \log n + \log(|\mathcal{G}|/\delta)}{\#_n(g)}}\right)$$

(say, using ERM); and then execute Algorithm 2 using $\{\hat{h}_g : g \in \mathcal{G}\}$ instead of $\mathcal{H}$, along with the other half of the training data. Removing the dependence on $\log |\mathcal{G}|$ is algorithmically less straightforward.

5. The group-realizable setting

We restrict our attention now to the case where $\mathcal{Y} = \mathcal{Z} = \{-1, +1\}$ and $\ell(z, y) = \mathbb{1}[z \neq y]$ is the binary zero-one loss. In the *group-realizable* setting, each group has an associated perfect classifier:

$$\min_{h \in \mathcal{H}} L(h \mid g) \;=\; 0 \quad \text{for all } g \in \mathcal{G}.$$

Note that *realizability* is the stronger assumption $\min_{h \in \mathcal{H}} L(h) = 0$. Realizability implies group-realizability, but not vice versa. In this setting, the arguments from Section 4 can be adapted to show that the randomized predictor produced by Algorithm 2 achieves error

$$L(Q \mid g) \;\leq\; O\left(\frac{\log(|\mathcal{G}||\mathcal{H}|)}{\#_n(g)}\right)$$

with high probability for all $g \in \mathcal{G}$. However, in the group-realizable setting, one can achieve this rate using a simpler approach.

Algorithm 3 shows the algorithm for the group-realizable setting. The idea is to use ERM to fit a classifier to every group. Since the distribution is group-realizable, all of these classifiers will be consistent on the data from their respective groups. Given a data point to predict on, we collect all the groups it lies in and predict using the majority vote of the associated groups.

Theorem 10. *Suppose Algorithm 3 is run on n i.i.d. examples from a group-realizable distribution and returns f. With probability at least $1 - \delta$,*

$$L(f \mid g) \;\leq\; O\left(\frac{\log\left(|\mathcal{G}||\mathcal{H}|/\delta\right)}{\#_n(g)}\right) \quad \textit{for all } g \in \mathcal{G}.$$

The function class used by Algorithm 3 is simple but seemingly quite powerful. One natural question is whether one could use this class to solve the multi-group learning problem in the general agnostic setting, for example via some sample splitting approach. The following result shows that this not possible in general.

Proposition 11. *There is a set of hypotheses $\mathcal{H} = \{h, h'\}$, a set of groups $\mathcal{G} = \{g_1, g_2, g_3\}$ and a distribution $\mathcal{D}$ such that the following holds. If $\mathcal{F}$ is the set of possible predictors produced by Algorithm 3, then for all $f \in \mathcal{F}$, there exists some $g \in \mathcal{G}$ such that*

$$L(f \mid g) > \min_{h \in \mathcal{H}} L(h \mid g) + \frac{1}{4}$$

where the loss is zero-one loss.

6. Discussion and open problems

In this work, we presented simple and near-optimal algorithms for multi-group learning. Here we point to some interesting directions for future work.

Computation. It is not clear if any of the algorithms considered in this work can be made efficient, as they seem to rely either on enumeration or on complicated optimization subroutines. Thus, it is an interesting open problem to devise multi-group learning algorithms that are efficient for some specific choices of $\mathcal{H}$ and $\mathcal{G}$ and some restrictions on the marginal distribution over $\mathcal{X}$.

Representation. Through Algorithm 1, we have addressed the question of representational complexity for multi-group learning: we showed that decision lists of the form in $\mathtt{DL}_T[\mathcal{G}; \mathcal{H}]$ are sufficient. However, the full expressivity of this class is not necessary in all cases (such as in the group-realizable setting). An expanded investigation of these representational issues should address this gap.

Simplicity and optimality. Finally, it remains an interesting open problem is to design an algorithm that is simple like Algorithm 1 but that also enjoys the performance guarantees of Algorithm 2.

Acknowledgements

We thank Kamalika Chaudhuri for helpful initial discussions about hidden stratification. We acknowledge support from NSF grants CCF-1740833 and IIS-1563785, and a JP Morgan Faculty Award. Part of this work was completed while CT was at Columbia University.

References

Agarwal, A., Beygelzimer, A., Dudík, M., Langford, J., and Wallach, H. A reductions approach to fair classification. In *International Conference on Machine Learning*, pp. 60–69, 2018.

Azizzadenesheli, K., Liu, A., Yang, F., and Anandkumar, A. Regularized learning for domain adaptation under label shifts. In *International Conference on Learning Representations*, 2018.

Balsubramani, A., Dasgupta, S., Freund, Y., and Moran, S. An adaptive nearest neighbor rule for classification. In *Advances in Neural Information Processing Systems*, pp. 7579–7588, 2019.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

Ben-Tal, A., El Ghaoui, L., and Nemirovski, A. *Robust optimization*. Princeton university press, 2009.

Bertsimas, D., Gupta, V., and Kallus, N. Data-driven robust optimization. *Mathematical Programming*, 167(2):235–292, 2018.

Blum, A. Empirical support for winnow and weighted-majority algorithms: Results on a calendar scheduling domain. *Machine Learning*, 26(1):5–23, 1997.

Blum, A. and Lykouris, T. Advancing subgroup fairness via sleeping experts. In *Innovations in Theoretical Computer Science Conference (ITCS)*, 2020.

Blum, A. and Mansour, Y. From external to internal regret. *Journal of Machine Learning Research*, 8(6), 2007.

Boucheron, S., Bousquet, O., and Lugosi, G. Theory of classification: A survey of some recent advances. *ESAIM: probability and statistics*, 9:323–375, 2005.

Cesa-Bianchi, N., Conconi, A., and Gentile, C. On the generalization ability of on-line learning algorithms. *IEEE Transactions on Information Theory*, 50(9):2050–2057, 2004.

Cortes, C., Mansour, Y., and Mohri, M. Learning bounds for importance weighting. In *Advances in Neural Information Processing Systems*, pp. 442–450, 2010.

Cortes, C., Greenberg, S., and Mohri, M. Relative deviation learning bounds and generalization with unbounded loss functions. *Annals of Mathematics and Artificial Intelligence*, 85(1):45–70, 2019.

Diana, E., Gill, W., Kearns, M., Kenthapadi, K., Roth, A., and Sharifi-Malvajerdi, S. Multiaccurate proxies for downstream fairness. *arXiv preprint arXiv:2107.04423*, 2021.

Donini, M., Oneto, L., Ben-David, S., Shawe-Taylor, J., and Pontil, M. Empirical risk minimization under fairness constraints. In *Advances in Neural Information Processing Systems*, pp. 2796–2806, 2018.

Duchi, J., Hashimoto, T., and Namkoong, H. Distributionally robust losses for latent covariate mixtures, 2020.

Duchi, J. C., Glynn, P. W., and Namkoong, H. Statistics of robust optimization: A generalized empirical likelihood approach. *Mathematics of Operations Research*, 2021.

Dwork, C. and Ilvento, C. Individual fairness under composition. *Proceedings of Fairness, Accountability, Transparency in Machine Learning*, 2018.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pp. 214–226, 2012.

Dwork, C., Kim, M. P., Reingold, O., Rothblum, G. N., and Yona, G. Outcome indistinguishability. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 1095–1108, 2021.

Freedman, D. On tail probabilities for martingales. *The Annals of Probability*, 3(1):100–118, 1975.

Freund, Y. Boosting a weak learning algorithm by majority. *Information and computation*, 121(2):256–285, 1995.

Freund, Y., Schapire, R. E., Singer, Y., and Warmuth, M. K. Using and combining predictors that specialize. In *Proceedings of the twenty-ninth annual ACM Symposium on Theory of Computing*, pp. 334–343, 1997.

Gaillard, P., Stoltz, G., and Van Erven, T. A second-order bound with excess losses. In *Conference on Learning Theory*, pp. 176–196. PMLR, 2014.

Gillen, S., Jung, C., Kearns, M., and Roth, A. Online learning with an unknown fairness metric. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 2605–2614, 2018.

Globus-Harris, I., Kearns, M., and Roth, A. Beyond the frontier: Fairness without privacy loss. *arXiv preprint arXiv:2201.10408*, 2022.

Gupta, V., Jung, C., Noarov, G., Pai, M. M., and Roth, A. Online multivalid learning: Means, moments, and prediction intervals. *arXiv preprint arXiv:2101.01739*, 2021.

Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29:3315–3323, 2016.

Hébert-Johnson, U., Kim, M., Reingold, O., and Rothblum, G. Multicalibration: Calibration for the (computationally-identifiable) masses. In *International Conference on Machine Learning*, pp. 1939–1948, 2018.

Helmbold, D. P. and Warmuth, M. K. On weak learning. *Journal of Computer and System Sciences*, 50(3):551–573, 1995.

Hopkins, M., Kane, D., Lovett, S., and Mahajan, G. Realizable learning is all you need. *arXiv preprint arXiv:2111.04746*, 2021.

Kearns, M., Neel, S., Roth, A., and Wu, Z. S. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *International Conference on Machine Learning*, pp. 2564–2572, 2018.

Kim, M., Ghorbani, A., and Zou, J. Multiaccuracy: Black-box post-processing for fairness in classification. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 247–254, 2019.

Kpotufe, S. and Martinet, G. Marginal singularity, and the benefits of labels in covariate-shift. In *Conference On Learning Theory*, pp. 1882–1886, 2018.

Liu, E. Z., Haghgoo, B., Chen, A. S., Raghunathan, A., Koh, P. W., Sagawa, S., Liang, P., and Finn, C. Just train twice: Improving group robustness without training group information. In *International Conference on Machine Learning*, pp. 6781–6792. PMLR, 2021.

Mohri, M., Sivek, G., and Suresh, A. T. Agnostic federated learning. In *International Conference on Machine Learning*, pp. 4615–4625, 2019.

Noarov, G., Pai, M., and Roth, A. Online multiobjective minimax optimization and applications. *arXiv preprint arXiv:2108.03837*, 2021.

Oakden-Rayner, L., Dunnmon, J., Carneiro, G., and Ré, C. Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. In *Proceedings of the ACM conference on health, inference, and learning*, pp. 151–159, 2020.

Rothblum, G. and Yona, G. Multi-group agnostic pac learnability. In *International Conference on Machine Learning*, pp. 9107–9115, 2021a.

Rothblum, G. and Yona, G. Personal communication, 2021b.

Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations*, 2020a.

Sagawa, S., Raghunathan, A., Koh, P. W., and Liang, P. An investigation of why overparameterization exacerbates spurious correlations. In *International Conference on Machine Learning*, pp. 8346–8356. PMLR, 2020b.

Schapire, R. E. The strength of weak learnability. *Machine learning*, 5(2):197–227, 1990.

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

Shimodaira, H. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

Sohoni, N. S., Dunnmon, J. A., Angus, G., Gu, A., and Ré, C. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. *arXiv preprint arXiv:2011.12945*, 2020.

Trevisan, L., Tulsiani, M., and Vadhan, S. Regularity, boosting, and efficiently simulating every high-entropy distribution. In *24th Annual IEEE Conference on Computational Complexity*, 2009.

Zadrozny, B. Learning and evaluating classifiers under sample selection bias. In *International Conference on Machine Learning*, pp. 114, 2004.

A. Missing proofs from Section 2

A.1. Proof of Theorem 1

To simplify the proof, we will use the following notation. Let P be a probability distribution over $\mathcal{X}$, let $\mathcal{H}$ be a family of $[0,1]$-valued functions over $\mathcal{X}$, and let $\mathcal{G}$ be a family of $\{0,1\}$-valued functions over $\mathcal{X}$. Given a sample $x_1, \ldots, x_n$ drawn from P, we make the following definitions:

$$\begin{aligned} P(h \mid g) &:= \frac{P(hg)}{P(g)} := \frac{\mathbb{E}[h(x)g(x)]}{\mathbb{E}[g(x)]} \\ P_n(h \mid g) &:= \frac{P_n(hg)}{P_n(g)} := \frac{\sum_{i=1}^n h(x_i)g(x_i)}{\#_n(g)}. \end{aligned}$$

Given this notation, we will prove the following theorem, which directly implies Theorem 1.

Theorem 12. *Let P be a probability distribution over $\mathcal{X}$, let $\mathcal{H}$ be a family of $[0,1]$-valued functions over $\mathcal{X}$, and let $\mathcal{G}$ be a family of $\{0,1\}$-valued functions over $\mathcal{X}$. Then with probability at least $1-\delta$,*

$$\left|P(h \mid g) - P_n(h \mid g)\right| \;\leq\; \min\left\{9\sqrt{\frac{D}{\#_n(g)}}, 7\sqrt{\frac{DP_n(h \mid g)}{\#_n(g)}} + \frac{16D}{\#_n(g)}\right\}$$

for all $h \in \mathcal{H}, g \in \mathcal{G}$, where $D = 2\log \Pi_{2n}(\mathcal{H}_{\text{thresh}}) + 2\log \Pi_{2n}(\mathcal{G}) + \log(8/\delta)$.

To prove Theorem 12, we will provide relative deviation bounds on $[0,1]$-valued functions of the following form: with probability $1-\delta$,

$$|P(h) - P_n(h)| \;\leq\; \sqrt{\frac{P_n(h)\text{comp}(\mathcal{H})\log(1/\delta)}{n}} + \frac{\text{comp}(\mathcal{H})\log(1/\delta)}{n}$$

for all $h \in \mathcal{H}$, where $\text{comp}(\mathcal{H})$ is some complexity measure of $\mathcal{H}$.

To establish this relative deviation bound, we first reduce the problem from $[0,1]$-valued functions to $\{0,1\}$-valued functions.

Lemma 13. *If $\mathcal{H}$ is a class of $[0,1]$-valued function, then*

$$\begin{aligned} \Pr\left(\sup_{h\in\mathcal{H}} \frac{P(h) - P_n(h)}{\sqrt{P(h)}} > \epsilon\right) &\leq \Pr\left(\sup_{h'\in\mathcal{H}_{\text{thresh}}} \frac{P(h') - P_n(h')}{\sqrt{P(h')}} > \epsilon\right) \\ \Pr\left(\sup_{h\in\mathcal{H}} \frac{P_n(h) - P(h)}{\sqrt{P_n(h)}} > \epsilon\right) &\leq \Pr\left(\sup_{h'\in\mathcal{H}_{\text{thresh}}} \frac{P_n(h') - P(h')}{\sqrt{P_n(h')}} > \epsilon\right). \end{aligned}$$

Proof. The proof is inspired by the proof of Theorem 3 of Cortes et al. (2019). For $h \in \mathcal{H}, t \in [0,1]$, we use the notation

$$\begin{aligned} P(h > t) &= \mathbb{E}[\mathbb{1}[h(x) > t]] \\ P_n(h > t) &= \frac{1}{n}\sum_{i=1}^n \mathbb{1}[h(x_i) > t]. \end{aligned}$$

To prove the lemma, it will suffice to prove the following two implications

$$\begin{aligned} \forall h \in \mathcal{H}, t \in [0,1],\, P(h > t) - P_n(h > t) \leq \epsilon\sqrt{P(h > t)} &\longrightarrow \forall h \in \mathcal{H},\, P(h) - P_n(h) \leq \epsilon\sqrt{P(h)} \\ \forall h \in \mathcal{H}, t \in [0,1],\, P_n(h > t) - P(h > t) \leq \epsilon\sqrt{P_n(h > t)} &\longrightarrow \forall h \in \mathcal{H},\, P_n(h) - P(h) \leq \epsilon\sqrt{P_n(h)}. \end{aligned}$$

We will prove the first statement, as the second can be proven symmetrically. Assume that

$$P(h > t) - P_n(h > t) \leq \epsilon\sqrt{P(h > t)}$$

for all $h \in \mathcal{H}$ and $t \in [0,1]$. Since h is $[0,1]$-valued, we can write

$$\begin{aligned} P(h) &= \int_0^1 P(h > t)\ \mathrm{d}t \\ P_n(h) &= \int_0^1 P_n(h > t)\ \mathrm{d}t. \end{aligned}$$

Thus,

$$\begin{aligned} P(h) - P_n(h) &= \int_0^1 P(h > t) - P_n(h > t)\ \mathrm{d}t \\ &\leq \int_0^1 \epsilon\sqrt{P(h > t)}\ \mathrm{d}t \\ &\leq \epsilon\sqrt{\int_0^1 P(h > t)\ \mathrm{d}t} \\ &= \epsilon\sqrt{P(h)} \end{aligned}$$

where the first inequality is by assumption and the second is by Jensen's inequality. □

The following result, found for example in Boucheron et al. (2005), provides uniform convergence bounds on the relative deviations of $\{0,1\}$-valued functions.

Lemma 14. *If $\mathcal{F}$ is a class of $\{0,1\}$-valued function, then*

$$\begin{aligned} \Pr\left(\sup_{f\in\mathcal{F}} \frac{P_n(f) - P(f)}{\sqrt{P_n(f)}} > \epsilon\right) &\leq 4\Pi_{2n}(\mathcal{F})\exp\left(-\frac{\epsilon^2 n}{4}\right) \\ \Pr\left(\sup_{f\in\mathcal{F}} \frac{P(f) - P_n(f)}{\sqrt{P(f)}} > \epsilon\right) &\leq 4\Pi_{2n}(\mathcal{F})\exp\left(-\frac{\epsilon^2 n}{4}\right) \end{aligned}$$

where $\Pi_k(\mathcal{F})$ is the k-th shattering coefficient of $\mathcal{F}$.

Combining these two results, we have the following.

Lemma 15. *If $\mathcal{H}$ is a class of $[0,1]$-valued function, then with probability $1-\delta$,*

$$\begin{aligned} P(h) - P_n(h) &\leq 2\sqrt{P_n(h)\frac{\log\Pi_{2n}(\mathcal{H}_{\text{thresh}}) + \log(8/\delta)}{n}} + 4\frac{\log\Pi_{2n}(\mathcal{H}_{\text{thresh}}) + \log(8/\delta)}{n} \\ P(h) - P_n(h) &\geq -2\sqrt{P_n(h)\frac{\log\Pi_{2n}(\mathcal{H}_{\text{thresh}}) + \log(8/\delta)}{n}} \end{aligned}$$

for all $h \in \mathcal{H}$.

Proof. Combining Lemmas 13 and 14, we immediately have that with probability $1-\delta$

$$\begin{aligned} \frac{P_n(h) - P(h)}{\sqrt{P_n(h)}} &\leq 2\sqrt{\frac{\log\Pi_{2n}(\mathcal{H}_{\text{thresh}}) + \log(8/\delta)}{n}} \\ \frac{P(h) - P_n(h)}{\sqrt{P(h)}} &\leq 2\sqrt{\frac{\log\Pi_{2n}(\mathcal{H}_{\text{thresh}}) + \log(8/\delta)}{n}} \end{aligned}$$

for all $h \in \mathcal{H}$. Let us condition on this occurring.

Using the standard trick that for $A, B, C \geq 0$, the inequality $A \leq B\sqrt{A} + C$ entails the inequality $A \leq B^2 + B\sqrt{C} + C$, we can observe from the second inequality above that

$$P(h) \leq P_n(h) + 2\sqrt{P_n(h)\frac{\log\Pi_{2n}(\mathcal{H}_{\text{thresh}}) + \log(8/\delta)}{n}} + 4\frac{\log\Pi_{2n}(\mathcal{H}_{\text{thresh}}) + \log(8/\delta)}{n}.$$

Combined with the first inequality, we have the lemma statement. □

Now we turn to the proof of Theorem 12.

Proof of Theorem 12. The proof is similar to the proof of Theorem 5 of Balsubramani et al. (2019). Let $\mathcal{F} = \mathcal{G} \cup \{hg : h \in \mathcal{H}, g \in \mathcal{G}\}$. Note that for $g \in \mathcal{G}$, $h \in \mathcal{H}$ and $t \in \mathbb{R}$, we have

$$\mathbb{1}[h(x)g(x) > t] = g(x)\,\mathbb{1}[h(x) > t].$$

Thus, if we let $\mathcal{C} = \{hg : h \in \mathcal{H}, g \in \mathcal{G}\}$, we observe that

$$\begin{aligned}\log \Pi_n(\mathcal{F}_{\text{thresh}}) &\le \log \Pi_n(\mathcal{G}) + \log \Pi_n(\mathcal{C}_{\text{thresh}}) \\ &\le \log \Pi_n(\mathcal{G}) + \log \Pi_n(\mathcal{H}_{\text{thresh}})\Pi_n(\mathcal{G}) \\ &\le 2\log \Pi_n(\mathcal{H}_{\text{thresh}})\Pi_n(\mathcal{G}).\end{aligned}$$

Combining this with Lemma 15, we have with probability $1-\delta$

$$P_n(f) - 2\sqrt{P_n(f)\frac{D}{n}} \;\le\; P(f) \;\le\; P_n(f) + 2\sqrt{P_n(f)\frac{D}{n}} + 4\frac{D}{n}$$

for all $f \in \mathcal{F}$, where we used the definition $D = 2\log\left(\Pi_{2n}(\mathcal{H}_{\text{thresh}})\right) + 2\log\left(\Pi_{2n}(\mathcal{G})\right) + \log(8/\delta)$. Let us condition on this event holding.

Now fix some $h \in \mathcal{H}$ and $g \in \mathcal{G}$. We can work out

$$\begin{aligned}P(h \mid g) &= \frac{P(hg)}{P(g)} \\ &\le \frac{P_n(hg) + 2\sqrt{P_n(hg)\frac{D}{n}} + 4\frac{D}{n}}{P_n(g) - 2\sqrt{P_n(g)\frac{D}{n}}} \\ &= \frac{P_n(hg)}{P_n(g)} \cdot \frac{1 + 2\sqrt{\frac{D}{nP_n(hg)}} + 4\frac{D}{nP_n(hg)}}{1 - 2\sqrt{\frac{D}{nP_n(g)}}} \\ &= \frac{\#_n(hg)}{\#_n(g)} \cdot \frac{1 + 2\sqrt{\frac{D}{\#_n(hg)}} + 4\frac{D}{\#_n(hg)}}{1 - 2\sqrt{\frac{D}{\#_n(g)}}}.\end{aligned}$$

Here, we have used the notation $\#_n(f) = nP_n(f)$. Observe that if $\#_n(g) \le 16D$, the theorem statement is trivial. Thus, we may assume that $\#_n(g) > 16D$, whereupon the inequality $\frac{1}{1-x} \le 1 + 2x$ for all $x < 1/2$ implies

$$\frac{1}{1 - 2\sqrt{\frac{D}{\#_n(g)}}} \;\le\; 1 + 4\sqrt{\frac{D}{\#_n(g)}}.$$

Thus, we have

$$\begin{aligned}P(h \mid g) &\le \frac{\#_n(hg)}{\#_n(g)} \cdot \left(1 + 2\sqrt{\frac{D}{\#_n(hg)}} + 4\frac{D}{\#_n(hg)}\right)\left(1 + 4\sqrt{\frac{D}{\#_n(g)}}\right) \\ &= \frac{\#_n(hg)}{\#_n(g)} \cdot \left(1 + 2\sqrt{\frac{D}{\#_n(hg)}} + 4\frac{D}{\#_n(hg)} + 4\sqrt{\frac{D}{\#_n(g)}} + 4\frac{D}{\sqrt{\#_n(hg)\#_n(g)}} + 16\frac{D^{3/2}}{\#_n(hg)\sqrt{\#_n(g)}}\right) \\ &= \frac{\#_n(hg)}{\#_n(g)} + 2\frac{\sqrt{D\#_n(hg)}}{\#_n(g)} + 4\frac{D}{\#_n(g)} + 4\frac{\#_n(hg)}{\#_n(g)}\sqrt{\frac{D}{\#_n(g)}} + 4\frac{D\sqrt{\#_n(hg)}}{\#_n(g)^{3/2}} + 16\frac{D^{3/2}}{\#_n(g)^{3/2}} \\ &\le P_n(h \mid g) + \min\left\{9\sqrt{\frac{D}{\#_n(g)}}, 7\sqrt{\frac{DP_n(h \mid g)}{\#_n(g)}} + \frac{8D}{\#_n(g)}\right\}\end{aligned}$$

where we have made use of the inequalities $\#_n(hg) \leq \#_n(g)$ and $\#_n(g) > 16D$.

In the other direction, we have two cases: $\#_n(hg) < 4D$ and $\#_n(hg) \geq 4D$. Let us assume first that $\#_n(hg) < 4D$. Then observe that

$$\begin{aligned} P_n(h \mid g) - 9\sqrt{\frac{D}{\#_n(g)}} &= \frac{\#_n(hg)}{\#_n(g)} - 9\sqrt{\frac{D}{\#_n(g)}} \\ &< \frac{4D}{\#_n(g)} - 9\sqrt{\frac{D}{\#_n(g)}} \\ &= \sqrt{\frac{D}{\#_n(g)}}\left(4\sqrt{\frac{D}{\#_n(g)}} - 9\right) \;\leq\; 0 \;\leq\; P(h \mid g) \end{aligned}$$

where we have used the fact that $\#_n(g) > 16D$. Similarly, we also have

$$\begin{aligned} P_n(h \mid g) - 7\sqrt{\frac{DP_n(h \mid g)}{\#_n(g)}} - \frac{8D}{\#_n(g)} &= \frac{\#_n(hg)}{\#_n(g)} - 7\sqrt{\frac{D\#_n(hg)}{\#_n(g)^2}} - \frac{8D}{\#_n(g)} \\ &< \frac{4D}{\#_n(g)} - 7\sqrt{\frac{D\#_n(hg)}{\#_n(g)^2}} - \frac{8D}{\#_n(g)} \;\leq\; 0 \;\leq\; P(h \mid g) \end{aligned}$$

Thus, we may assume that $\#_n(hg) \geq 4D$, so that we have

$$\begin{aligned} P(h \mid g) &= \frac{P(hg)}{P(g)} \\ &\geq \frac{P_n(hg) - 2\sqrt{P_n(hg)\frac{D}{n}}}{P_n(g) + 2\sqrt{P_n(g)\frac{D}{n}} + 4\frac{D}{n}} \\ &= \frac{\#_n(hg)}{\#_n(g)} \cdot \frac{1 - 2\sqrt{\frac{D}{\#_n(hg)}}}{1 + 2\sqrt{\frac{D}{\#_n(g)}} + 4\frac{D}{\#_n(g)}}. \end{aligned}$$

Using the inequality $\frac{1}{1+x} \geq 1 - x$ for all $x \geq 0$, we have

$$\begin{aligned} P(h \mid g) &\geq \frac{\#_n(hg)}{\#_n(g)}\left(1 - 2\sqrt{\frac{D}{\#_n(hg)}}\right)\left(1 - 2\sqrt{\frac{D}{\#_n(g)}} - 4\frac{D}{\#_n(g)}\right) \\ &\geq \frac{\#_n(hg)}{\#_n(g)}\left(1 - 2\sqrt{\frac{D}{\#_n(hg)}} - 2\sqrt{\frac{D}{\#_n(g)}} - 4\frac{D}{\#_n(g)}\right) \\ &= \frac{\#_n(hg)}{\#_n(g)} - 2\frac{\sqrt{D\#_n(hg)}}{\#_n(g)} - 2\frac{\#_n(hg)\sqrt{D}}{\#_n(g)^{3/2}} - 4\frac{D\#_n(hg)}{\#_n(g)^2} \\ &\geq P_n(h \mid g) - \min\left\{5\sqrt{\frac{D}{\#_n(g)}}, 4\sqrt{\frac{DP_n(h \mid g)}{\#_n(g)}} + \frac{4D}{\#_n(g)}\right\} \end{aligned}$$

where we again made use of the inequalities $\#_n(hg) \leq \#_n(g)$ and $\#_n(g) > 16D$. □

B. Missing proofs from Section 3

B.1. Proof of Lemma 2

We first note that if the algorithm terminates at round t, then we trivially must have

$$L_n(f_t \mid g) \leq \inf_{h \in \mathcal{H}} L_n(h \mid g) + \epsilon_n(g) \quad \text{for all } g \in \mathcal{G}.$$

If the algorithm does not terminate at round t, then we must have found a pair h_{t+1}, g_{t+1} such that

$$L(f_t \mid g_{t+1}) - L(h_{t+1} \mid g_{t+1}) \geq \epsilon_n(g_{t+1}),$$

and we must have prepended the pair (h_{t+1}, g_{t+1}) onto f_t to create f_{t+1}. Observe that this prepending action implies that f_{t+1} will agree with h_{t+1} on g_{t+1} and agree with f_t everywhere else. Thus, we have

$$\begin{aligned} L_n(f_t) - L_n(f_{t+1}) &= \mathbb{E}\left[\ell(f_t(x), y) - \ell(f_{t+1}(x), y)\right] \\ &= P_n(g_{t+1})\mathbb{E}\left[\ell(f_t(x), y) - \ell(h_{t+1}(x), y) \mid g_{t+1}(x) = 1\right] \\ &\geq P_n(g_{t+1})\epsilon_n(g_{t+1}) \;\geq\; \epsilon_o. \end{aligned}$$

Thus, $L_n(f_t)$ decreases by ϵ_o at every update, and we have $L_n(f_0) \leq \alpha$. The theorem follows by combining these two observations.

B.2. Proof of Proposition 3

The discussion before the statement of Proposition 3 implies that $|\mathtt{DL}_T[\mathcal{G}; \mathcal{H}]| \leq |\mathcal{H}|^{|\mathcal{G}|+1}|\mathcal{G}|^{|\mathcal{G}|}$. Applying Theorem 1, and utilizing the fact that for any finite class $\mathcal{H}$, we have $\Pi_n(\mathcal{H}) \leq |\mathcal{H}|$, we have with probability at least $1 - \delta$

$$\begin{aligned} |L(f \mid g) - L_n(f \mid g)| &\leq 9\sqrt{\frac{2\log|\mathtt{DL}_T[\mathcal{G}; \mathcal{H}]| + 2\log|\mathcal{G}| + \log(8/\delta)}{\#_n(g)}} \\ &\leq 9\sqrt{\frac{2(|\mathcal{G}|+1)\log(|\mathcal{H}||\mathcal{G}|) + \log(8/\delta)}{\#_n(g)}} \end{aligned}$$

for all $f \in \mathtt{DL}_T[\mathcal{G}; \mathcal{H}]$ and $g \in \mathcal{G}$. Combined with Lemma 2, we have the proposition statement.

B.3. Proof of Theorem 4

We will actually show the slightly stronger bound of

$$L(f \mid g) \;\leq\; \min_{h \in \mathcal{H}} L(h \mid g) + 22\left(\frac{\alpha\log(|\mathcal{G}||\mathcal{H}|)}{\gamma\#_n(g)}\right)^{1/3} + 18\sqrt{\frac{\log(8/\delta)}{\#_n(g)}}$$

where $\alpha = \min_{h \in \mathcal{H}} L_n(h)$. The theorem follows from the fact that $\alpha \leq 1$.

We will take $\epsilon_n(g)$ to be the function

$$\epsilon_n(g) \;=\; 36^{2/3}\left(\alpha\log(|\mathcal{G}||\mathcal{H}|)\right)^{1/3}\left(\frac{n}{\gamma}\right)^{1/6}\left(\frac{1}{\#_n(g)}\right)^{1/2}.$$

Then the number of rounds PREPEND takes is bounded as

$$\begin{aligned} T &\leq \frac{\alpha}{\min_{g \in \mathcal{G}_{n,\gamma}} P_n(g)\epsilon_n(g)} \\ &= \frac{\alpha}{36^{2/3}\left(\alpha\log(|\mathcal{G}||\mathcal{H}|)\right)^{1/3}(n/\gamma)^{1/6}\min_{g \in \mathcal{G}_{n,\gamma}} P_n(g)(\#_n(g))^{-1/2}} \\ &= \alpha^{2/3}36^{-2/3}\left(\log(|\mathcal{G}||\mathcal{H}|)\right)^{-1/3}\left(\frac{n}{\gamma}\right)^{-1/6} \cdot \left(\frac{n}{\gamma}\right)^{1/2} \\ &= \alpha^{2/3}36^{-2/3}\left(c\log(|\mathcal{G}||\mathcal{H}|)\right)^{-1/3}\left(\frac{n}{\gamma}\right)^{1/3}. \end{aligned}$$

Observe that since $|\mathcal{G}|$ and $|\mathcal{H}|$ are finite, we have $|\mathtt{DL}_T[\mathcal{G}; \mathcal{H}]| \leq |\mathcal{H}|^{T+1}|\mathcal{G}|^T$. Combining Theorem 1 and Lemma 2, we

have with probability $1-\delta$

$$\begin{aligned}
L(f \mid g) \leq\ & L_n(f \mid g) + 9\sqrt{\frac{2(T+1)\log(|\mathcal{H}||\mathcal{G}|) + \log(8/\delta)}{\#_n(g)}} \\
\leq\ & \min_{h\in\mathcal{H}} L_n(h \mid g) + \epsilon_n(g) + 9\sqrt{\frac{2(T+1)\log(|\mathcal{H}||\mathcal{G}|) + \log(8/\delta)}{\#_n(g)}} \\
\leq\ & \min_{h\in\mathcal{H}} L(h \mid g) + \epsilon_n(g) + 18\sqrt{\frac{2(T+1)\log(|\mathcal{H}||\mathcal{G}|) + \log(8/\delta)}{\#_n(g)}} \\
\leq\ & \min_{h\in\mathcal{H}} L(h \mid g) + \epsilon_n(g) + 36\sqrt{\frac{T\log(|\mathcal{H}||\mathcal{G}|)}{\#_n(g)}} + 18\sqrt{\frac{\log(8/\delta)}{\#_n(g)}}.
\end{aligned}$$

Now by our bound on T, we have

$$\begin{aligned}
36\sqrt{\frac{T\log(|\mathcal{H}||\mathcal{G}|)}{\#_n(g)}} \ \leq\ & 36\left(\frac{c\log(|\mathcal{H}||\mathcal{G}|)}{\#_n(g)}\cdot\alpha^{2/3}36^{-2/3}\left(\log(|\mathcal{G}||\mathcal{H}|)\right)^{-1/3}\left(\frac{n}{\gamma}\right)^{1/3}\right)^{1/2} \\
=\ & 36^{2/3}\left(\alpha\log(|\mathcal{G}||\mathcal{H}|)\right)^{1/3}\left(\frac{n}{\gamma}\right)^{1/6}\left(\frac{1}{\#_n(g)}\right)^{1/2} \ =\ \epsilon_n(g).
\end{aligned}$$

Thus, we have

$$L(f \mid g) \ \leq\ \min_{h\in\mathcal{H}} L_n(h \mid g) + 2\epsilon_n(g) + 18\sqrt{\frac{\log(8/\delta)}{\#_n(g)}}.$$

Finally, observe that $n/\gamma \leq \#_n(g)/\gamma^2$ for all $g \in \mathcal{G}_{n,\gamma}$. Thus, we have

$$\begin{aligned}
\epsilon_n(g) \ =\ & 36^{2/3}\left(\alpha\log(|\mathcal{G}||\mathcal{H}|)\right)^{1/3}\left(\frac{n}{\gamma}\right)^{1/6}\left(\frac{1}{\#_n(g)}\right)^{1/2} \\
\leq\ & 36^{2/3}\left(\alpha\log(|\mathcal{G}||\mathcal{H}|)\right)^{1/3}\left(\frac{\#_n(g)}{\gamma^2}\right)^{1/6}\left(\frac{1}{\#_n(g)}\right)^{1/2} \\
\leq\ & 36^{2/3}\left(\frac{\alpha\log(|\mathcal{G}||\mathcal{H}|)}{\gamma\#_n(g)}\right)^{1/3}. \quad \square
\end{aligned}$$

B.4. Proof of Corollary 5

By Lemma 14, we have with probability $1-\delta/2$ that

$$P_n(g) \ \geq\ P(g) - 2\sqrt{\frac{P(g)}{n}\log\left(16|\mathcal{G}|/\delta\right)}$$

for all $g \in \mathcal{G}$. For the choice of n in the statement, this implies $P_n(g) \geq \gamma/2$. By Theorem 4, we have with probability $1-\delta/2$,

$$L(f \mid g) \ \leq\ \min_{h\in\mathcal{H}} L(h \mid g) + 22\left(\frac{2\log(16|\mathcal{G}||\mathcal{H}|/\delta)}{\gamma\#_n(g)}\right)^{1/3}$$

for all $g \in \mathcal{G}_{n,\gamma/2}$. By a union bound, both of these hold with probability at least $1-\delta$, in which case we have $\mathcal{G}_{n,\gamma/2} = \mathcal{G}$. Plugging in our bound on $\#_n(g) = nP_n(g)$, we have

$$\begin{aligned}
L(f \mid g) \ \leq\ & \min_{h\in\mathcal{H}} L(h \mid g) + 22\left(\frac{2\log(16|\mathcal{G}||\mathcal{H}|/\delta)}{\gamma\#_n(g)}\right)^{1/3} \\
\leq\ & \min_{h\in\mathcal{H}} L(h \mid g) + 22\left(\frac{4\log(16|\mathcal{G}||\mathcal{H}|/\delta)}{\gamma^2 n}\right)^{1/3} \\
\leq\ & \min_{h\in\mathcal{H}} L(h \mid g) + \epsilon
\end{aligned}$$

where the last line follows from the choice of n in the corollary statement. $\square$

B.5. Proof of Theorem 6

Observe first that every prediction by $f \in \mathtt{DL}_T[\mathcal{G}; \mathcal{H}]$ is actually done by some $h \in \mathcal{H}$. Thus,

$$\begin{aligned}\Pi_n((\ell \circ \mathtt{DL}_T[\mathcal{G}; \mathcal{H}])_{\text{thresh}}) &\leq \Pi_n((\ell \circ \mathcal{H})_{\text{thresh}})^{T+1} \Pi_n(\mathcal{G})^T \\ &\leq \binom{n}{\leq d}^{T+1} \binom{n}{\leq d}^T \\ &\leq (2n)^{d(2T+1)}\end{aligned}$$

where the second line follows from the definition of pseudo-dimension and the Sauer-Shelah-Perles Lemma, and third line follows from the inequality $\binom{a}{\leq b} \leq (2a)^b$. Thus, with probability $1 - \delta$ we have

$$|L(f \mid g) - L_n(f \mid g)| \leq 18\sqrt{\frac{d(T+1)\log(2n) + \log(8/\delta)}{\#_n(g)}} \tag{3}$$

for all $f \in \mathtt{DL}_T[\mathcal{G}; \mathcal{H}]$ and $g \in \mathcal{G}$. Let us condition on this occurring.

We will take $\epsilon_n(g)$ to be the function

$$\epsilon_n(g) = \left(2 \cdot 36^2 d \log(2n)\right)^{1/3} \left(\frac{n}{\gamma}\right)^{1/6} \left(\frac{1}{\#_n(g)}\right)^{1/2}.$$

Then the number of rounds the rewrite algorithm takes is bounded as

$$\begin{aligned}T &\leq \frac{L_0}{\min_{g \in \mathcal{G}_{n,\gamma}} P_n(g)\epsilon_n(g)} \\ &= \frac{1}{\left(2 \cdot 36^2 d \log(2n)\right)^{1/3} (n/\gamma)^{1/6} \min_{g \in \mathcal{G}_{n,\gamma}} P_n(g)(\#_n(g))^{-1/2}} \\ &\leq \left(2 \cdot 36^2 d \log(2n)\right)^{-1/3} \left(\frac{n}{\gamma}\right)^{-1/6} \cdot \left(\frac{n}{\gamma}\right)^{1/2} \\ &= \left(2 \cdot 36^2 d \log(2n)\right)^{-1/3} \left(\frac{n}{\gamma}\right)^{1/3}.\end{aligned}$$

Here, we have used the fact that the loss is bounded above by 1. Eq. (3) implies that, for the function f returned by Algorithm 1, we have for all $g \in \mathcal{G}_{n,\gamma}$,

$$\begin{aligned}L(f \mid g) &\leq L_n(f \mid g) + 18\sqrt{\frac{d(T+1)\log(2n) + \log(8/\delta)}{\#_n(g)}} \\ &\leq \min_{h \in \mathcal{H}} L_n(h \mid g) + \epsilon_n(g) + 18\sqrt{\frac{d(T+1)\log(2n) + \log(8/\delta)}{\#_n(g)}} \\ &\leq \min_{h \in \mathcal{H}} L(h \mid g) + \epsilon_n(g) + 36\sqrt{\frac{d(T+1)\log(2n) + \log(8/\delta)}{\#_n(g)}} \\ &\leq \min_{h \in \mathcal{H}} L(h \mid g) + \epsilon_n(g) + 36\sqrt{\frac{2Td\log(2n)}{\#_n(g)}} + 36\sqrt{\frac{\log(8/\delta)}{\#_n(g)}}.\end{aligned}$$

Now by our bound on T, we have

$$\begin{aligned}36\sqrt{\frac{2Td\log(2n)}{\#_n(g)}} &\leq 36\sqrt{\frac{2d\log(2n)}{\#_n(g)} \left(2 \cdot 36^2 d \log(2n)\right)^{-1/3} \left(\frac{n}{\gamma}\right)^{1/3}} \\ &= \left(2 \cdot 36^2 d \log(2n)\right)^{1/3} \left(\frac{n}{\gamma}\right)^{1/6} \left(\frac{1}{\#_n(g)}\right)^{1/2} \\ &= \epsilon_n(g).\end{aligned}$$

Thus, we have

$$L(f \mid g) \;\leq\; \min_{h \in \mathcal{H}} L_n(h \mid g) + 2\epsilon_n(g) + 36\sqrt{\frac{\log(8/\delta)}{\#_n(g)}}.$$

Finally, observe that $n/\gamma \leq \#_n(g)/\gamma^2$ for all $g \in \mathcal{G}_{n,\gamma}$. Thus, we have

$$\begin{aligned}
\epsilon_n(g) \;&=\; \left(2 \cdot 36^2 d \log(2n)\right)^{1/3} \left(\frac{n}{\gamma}\right)^{1/6} \left(\frac{1}{\#_n(g)}\right)^{1/2} \\
&\leq\; \left(2 \cdot 36^2 d \log(2n)\right)^{1/3} \left(\frac{\#_n(g)}{\gamma^2}\right)^{1/6} \left(\frac{1}{\#_n(g)}\right)^{1/2} \\
&\leq\; 14 \left(\frac{d \log(2n)}{\gamma \#_n(g)}\right)^{1/3}. \quad \square
\end{aligned}$$

B.6. Proof of Proposition 7

Let $\mathcal{X} = \{x_0, x_1, x_2\}$ and $g_i = \{x_0, x_i\}$ for $x_i = 1, 2$. Let the marginal distribution over $\mathcal{X}$ be uniform. We will consider the 3 class classification setting, where $h_1(x) = 1$ and $h_2(x) = 2$ for all $x \in \mathcal{X}$. We will consider two scenarios for the conditional distribution of y given x.

In scenario 1, we have

$$P(y \mid x_0) = \begin{cases} 1/4 & \text{if } y = 1, \\ 0 & \text{if } y = 2, \\ 3/4 & \text{if } y = 3, \end{cases}$$

$$P(y \mid x_1) = \begin{cases} 3/4 & \text{if } y = 1, \\ 1/4 & \text{if } y = 2, \\ 0 & \text{if } y = 3, \end{cases}$$

$$P(y \mid x_2) = \begin{cases} 1 & \text{if } y = 2, \\ 0 & \text{if } y \in \{1, 3\}. \end{cases}$$

Abusing notation, we have under scenario 1:

$$\begin{aligned}
L(h_1 \mid x_0) &= 3/4 & L(h_1 \mid x_1) &= 1/4 & L(h_1 \mid x_2) &= 1 \\
L(h_2 \mid x_0) &= 1 & L(h_2 \mid x_1) &= 3/4 & L(h_2 \mid x_2) &= 0 \\
L(h_1 \mid g_1) &= 1/2 & L(h_1 \mid g_2) &= 7/8 & & \\
L(h_2 \mid g_1) &= 7/8 & L(h_2 \mid g_2) &= 1/2. & &
\end{aligned}$$

Observe that under scenario 1, for any decision list $f \in \texttt{DL}[\mathcal{G}; \mathcal{H}]$ that does not order (h_1, g_1) at the beginning there exists $g \in \mathcal{G}$ such that

$$L(f \mid g) \;\geq\; \min_{h \in \mathcal{H}} L(h \mid g) + 1/8.$$

Algorithm 4 MLC-HEDGE in the multi-group setting

input Groups $\mathcal{G}$, hypothesis class $\mathcal{H}$, learning rates $\eta_{h,g} \in [0,1]$.
output Internal hypotheses $p_1(\cdot;\cdot), \dots p_n(\cdot;\cdot)$.
Initialize weights $w_{h,g}^{(0)} = \frac{1}{|\mathcal{H}||\mathcal{G}|}$.
for $t = 1, 2, \dots, n$ **do**
Define

$$p_t((h,g);x) \;:=\; \frac{g(x)(1-e^{-\eta_{h,g}})w_{h,g}^{(t-1)}}{\sum_{h',g'} g'(x)(1-e^{-\eta_{h',g'}})w_{h',g'}^{(t-1)}}.$$

Receive point (x_t, y_t) and incur loss

$$\hat{\ell}_t \;=\; \sum_{h,g} g(x_t)\ell(h(x_t), y_t)p_t((h,g);x_t).$$

Update weight vectors

$$w_{h,g}^{(t)} \;=\; w_{h,g}^{(t-1)} \exp\left(\eta_{h,g} g(x_t)\left(\hat{\ell}_t e^{-\eta_{h,g}} - \ell(h(x_t), y_t)\right)\right).$$

end for
output $p_1, \dots, p_n$.

In scenario 2, we have

$$P(y \mid x_0) = \begin{cases} 0 & \text{if } y = 1, \\ 1/4 & \text{if } y = 2, \\ 3/4 & \text{if } y = 3, \end{cases}$$

$$P(y \mid x_1) = \begin{cases} 1 & \text{if } y = 1, \\ 0 & \text{if } y \in \{2,3\}, \end{cases}$$

$$P(y \mid x_2) = \begin{cases} 1/4 & \text{if } y = 1, \\ 3/4 & \text{if } y = 2, \\ 0 & \text{if } y = 3. \end{cases}$$

Under scenario 2:

$$\begin{aligned} L(h_1 \mid x_0) &= 1 & L(h_1 \mid x_1) &= 0 & L(h_1 \mid x_2) &= 3/4 \\ L(h_2 \mid x_0) &= 3/4 & L(h_2 \mid x_1) &= 1 & L(h_2 \mid x_2) &= 1/4 \\ L(h_1 \mid g_1) &= 1/2 & L(h_1 \mid g_2) &= 7/8 & & \\ L(h_2 \mid g_1) &= 7/8 & L(h_2 \mid g_2) &= 1/2. & & \end{aligned}$$

Conversely, we have that under scenario 2, for any decision list $f \in \texttt{DL}[\mathcal{G}; \mathcal{H}]$ that does not order (h_2, g_2) at the beginning there exists $g \in \mathcal{G}$ such that

$$L(f \mid g) \;\geq\; \min_{h \in \mathcal{H}} L(h \mid g) + 1/8.$$

C. Missing proofs from Section 4

C.1. MLC-HEDGE algorithm and guarantees

Algorithm 4 displays MLC-HEDGE, as presented by Gaillard et al. (2014), in the multi-group learning setting. Theorem 16 of Gaillard et al. (2014) translates as follows.

Theorem 16. *Let $\eta_{h,g} \in [0,1]$ be the learning rate assigned to expert (h,g), and suppose that the initial probabilities are uniform over the experts. For each expert (h,g), the cumulative loss of* MLC-HEDGE *satisfies*

$$\sum_{t=1}^{n} g(x_t)(\hat{\ell}_t - \ell(h(x_t), y_t)) \;\leq\; \left(e - 1 + \frac{1}{\eta_{h,g}}\right) \log(|\mathcal{H}||\mathcal{G}|) + (e-1)\eta_{h,g} \sum_{t=1}^{n} g(x_t)\ell(h(x_t), y_t).$$

C.2. An online-to-batch guarantee

For a collection of internal hypotheses $p_1, \ldots, p_n$ and a distribution Q over such hypotheses, we use the notational conventions

$$\begin{aligned} L(p_t \mid g) \;&:=\; \mathbb{E}_{(x,y)}\left[\mathbb{E}_{(\tilde{h},\tilde{g})\sim p_t(\cdot\,;\,x)}\left[\ell(\tilde{h}(x), y)\right] \mid g\right] \\ L(Q \mid g) \;&:=\; \mathbb{E}_{p_t \sim Q}\left[L(p_t \mid g)\right]. \end{aligned}$$

The following lemma shows that the average population losses of these internal hypotheses can be bounded in terms of their average empirical performance.

Lemma 17. *Suppose the loss function is bounded in the range $[0,1]$. Let $p_1, \ldots, p_n$ be a sequence of hypotheses produced by an online learning algorithm on an i.i.d. sequence $(x_1, y_1), \ldots, (x_n, y_n)$ with associated losses $\hat{\ell}_1, \ldots, \hat{\ell}_n$. Then with probability at least $1 - \delta$, we have for all $g \in \mathcal{G}$ simultaneously*

$$\frac{1}{n}\sum_{t=1}^{n} L(p_t \mid g) \;\leq\; \frac{1}{n}\sum_{t=1}^{n} \frac{g(x_t)}{P(g)}\hat{\ell}_t + \sqrt{\frac{1}{nP(g)}\log\frac{|\mathcal{G}|}{\delta}} + \frac{2}{3nP(g)}\log\frac{|\mathcal{G}|}{\delta}.$$

A key ingredient in the proof of Lemma 17 is Freedman's inequality (Freedman, 1975).

Theorem 18 (Freedman's inequality)**.** *Let $V_1, \ldots, V_T$ be a martingale difference sequence with respect to filtration $\mathcal{F}_t$ such that there exist constants $a, b \geq 0$ satisfying*

- *$|V_t| \leq a$ for all $t = 1, \ldots, T$ with probability 1 and*
- *$\sum_{t=1}^{T} \mathbb{E}[V_t^2 \mid \mathcal{F}_{t-1}] \leq b^2$.*

Then with probability at least $1 - \delta$, we have

$$\sum_{t=1}^{T} V_t \leq \frac{2}{3}a\log\frac{1}{\delta} + b\sqrt{2\ln\frac{1}{\delta}}.$$

Our proof of Lemma 17 is similar to the online-to-batch reduction of Cesa-Bianchi et al. (2004). Namely, fix $g \in \mathcal{G}$ and define the random variable

$$V_t \;=\; \frac{1}{n}L(p_t \mid g) - \frac{1}{nP(g)}g(x_t)\hat{\ell}_t \;=\; \frac{1}{n}L(p_t \mid g) - \frac{1}{nP(g)}g(x_t)\sum_{h,g'} g'(x_t)p_t(h,g';x_t)\ell(h(x_t), y_t).$$

Notice that $V_1, \ldots, V_n$ form a martingale difference sequence. Moreover, $V_t \in \left[-\frac{1}{nP(g)}, \frac{1}{nP(g)}\right]$. Letting $\mathcal{F}_t$ denote the

sigma-field of all outcomes up to time t, we can calculate

$$\begin{aligned}
&\mathbb{E}\left[\left(g(x_t)\sum_{h,g'} g'(x_t)p_t(h,g';x_t)\ell(h(x_t),y_t)\right)^2 \mid \mathcal{F}_{t-1}\right] \\
&\qquad = P(g)\mathbb{E}\left[\mathbb{E}\left[\left(\sum_{h,g'} g'(x_t)p_t(h,g';x_t)\ell(h(x_t),y_t)\right)^2 \mid g\right] \mid \mathcal{F}_{t-1}\right] \\
&\qquad \le P(g)\mathbb{E}\left[\mathbb{E}\left[\sum_{h,g'} g'(x_t)p_t(h,g';x_t)\ell(h(x_t),y_t)^2 \mid g\right] \mid \mathcal{F}_{t-1}\right] \\
&\qquad \le P(g)\mathbb{E}\left[L(p_t \mid g) \mid \mathcal{F}_{t-1}\right] = P(g)L(p_t \mid g)
\end{aligned}$$

where the first inequality is Jensen's inequality and the second follows from the fact that the losses lie in $[0,1]$. Thus,

$$\mathbb{E}[V_t^2 \mid \mathcal{F}_{t-1}] \le \frac{1}{n^2 P(g)} L(p_t \mid g) - \frac{1}{n^2} L(p_t \mid g)^2 \le \frac{1}{n^2 P(g)} L(p_t \mid g).$$

Freedman's inequality then implies that with probability at least $1-\delta/|\mathcal{G}|$,

$$\begin{aligned}
\frac{1}{n}\sum_{t=1}^{n} L(p_t \mid g) &\le \frac{1}{nP(g)}\sum_{t=1}^{T} g(x_t)\hat{\ell}_t + \frac{1}{n}\sqrt{\frac{1}{P(g)}\sum_{t=1}^{n} L(p_t \mid g)\log\frac{|\mathcal{G}|}{\delta}} + \frac{2}{3nP(g)}\log\frac{|\mathcal{G}|}{\delta} \\
&\le \frac{1}{nP(g)}\sum_{t=1}^{T} g(x_t)\hat{\ell}_t + \sqrt{\frac{1}{nP(g)}\log\frac{|\mathcal{G}|}{\delta}} + \frac{2}{3nP(g)}\log\frac{|\mathcal{G}|}{\delta},
\end{aligned}$$

where we have again used the fact that the losses lie in $[0,1]$. Taking a union bound over $\mathcal{G}$ finishes the proof. □

C.3. Proof of Theorem 8

We will show that with probability at least $1-\delta$, the predictor Q returned by Algorithm 2 satisfies

$$L(Q \mid g) \le \min_{h\in\mathcal{H}} L(h \mid g) + 60\sqrt{\frac{D}{\#_n(g)}} + \frac{16D}{\#_n(g)} \qquad \forall g \in \mathcal{G},$$

where $D = 2\log(|\mathcal{H}||\mathcal{G}|) + \log\frac{64}{\delta}$

Let $m = \lfloor n/2 \rfloor$, and let $(x_1,y_1),\ldots,(x_m,y_m)$, $(x'_1,y'_1),\ldots,(x'_m,y'_m)$ be the data split utilized by Algorithm 2. For these two splits of our data, we will use the notation

$$\begin{aligned}
S_m(g) &= \sum_{i=1}^{m} g(x_i) \\
S'_m(g) &= \sum_{i=1}^{m} g(x'_i) \\
L_m(h \mid g) &= \frac{1}{S_m(g)}\sum_{i=1}^{m} g(x_i)\ell(h(x_i),y_i).
\end{aligned}$$

From Theorem 1 and Lemma 15, we have that with probability at least $1-\delta/2$,

$$\begin{aligned}
L_m(h \mid g) &\leq L(h \mid g) + 9\sqrt{\frac{D}{S_m(g)}} \\
-2\sqrt{S_m(g)D} &\leq mP(g) - S_m(g) \leq 2\sqrt{S_m(g)D} + 4D \\
-2\sqrt{S'_m(g)D} &\leq mP(g) - S'_m(g) \leq 2\sqrt{S'_m(g)D} + 4D \\
-2\sqrt{\#_n(g)D} &\leq nP(g) - \#_n(g) \leq 2\sqrt{\#_n(g)D} + 4D
\end{aligned}$$

for all $h \in \mathcal{H}$ and $g \in \mathcal{G}$. Moreover, combining Theorem 16 and Lemma 17, we have that with probability at least $1-\delta/2$,

$$\begin{aligned}
L(Q \mid g) &= \frac{1}{m}\sum_{t=1}^{m} L(p_t \mid g) \\
&\leq \frac{1}{m}\sum_{t=1}^{m} \frac{g(x_t)}{P(g)}\hat{\ell}_t + \sqrt{\frac{1}{mP(g)}\log\frac{2|\mathcal{G}|}{\delta}} + \frac{2}{3mP(g)}\log\frac{2|\mathcal{G}|}{\delta} \\
&\leq \frac{1}{mP(g)}\left[\sum_{t=1}^{m} g(x_t)\ell(h_g(x_t), y_t) + \left(e - 1 + \frac{1}{\eta_{h_g,g}}\right)\log(|\mathcal{H}||\mathcal{G}|) + (e-1)\eta_{h_g,g}\sum_{t=1}^{m} g(x_t)\ell(h(x_t), y_t)\right] \\
&\quad + \sqrt{\frac{1}{mP(g)}\log\frac{2|\mathcal{G}|}{\delta}} + \frac{2}{3mP(g)}\log\frac{2|\mathcal{G}|}{\delta} \\
&\leq \frac{1}{mP(g)}\left[\sum_{t=1}^{m} g(x_t)\ell(h(x_t), y_t) + \left(2 + \frac{1}{\eta_{h_g,g}}\right)\log(|\mathcal{H}||\mathcal{G}|) + 2\eta_{h,g}S_m(g)\right] \\
&\quad + \sqrt{\frac{1}{mP(g)}\log\frac{2|\mathcal{G}|}{\delta}} + \frac{2}{3mP(g)}\log\frac{2|\mathcal{G}|}{\delta}
\end{aligned}$$

for all $g \in \mathcal{G}$, where $h_g := \operatorname{argmin}_{h\in\mathcal{H}} L(h \mid g)$ and the last line has used the fact that the losses are restricted to $[0,1]$. By a union bound, with probability at least $1-\delta$ all of the above occurs. Let us condition on this happening.

Pick some $g \in \mathcal{G}$. Observe that the theorem trivially holds if $\#_n(g) < 352D+4$. Thus, we may assume $\#_n(g) \geq 352D+4$. In this setting, we can then see that

$$\begin{aligned}
S_m(g) &\geq mP(g) - 2\sqrt{S_m(g)D} + 4D \\
&\geq \left(\frac{n}{2} - 1\right)P(g) - 2\sqrt{S_m(g)D} - 4D \\
&\geq \frac{1}{2}\left(\#_n(g) - 2\sqrt{\#_n(g)D}\right) - 2\sqrt{S_m(g)D} - 4D - 1 \\
&\geq \frac{1}{2}\#_n(g) - 3\sqrt{\#_n(g)D} - 4D - 1 \\
&\geq \frac{1}{4}\#_n(g).
\end{aligned}$$

Here the second-to-last line follows from the fact that $S_m(g) \leq \#_n(g)$ and the last line follows from our lower bound on $\#_n(g)$. By a similar chain of reasoning, we also have $S'_m(g) \geq \frac{1}{4}\#_n(g)$. Given this, we can bound

$$\begin{aligned}
\frac{1}{mP(g)}\sum_{t=1}^{m} g(x_t)\ell(h(x_t), y_t) &\leq \frac{S_m(g)}{mP(g)}\left(L(h \mid g) + 9\sqrt{\frac{D}{S_m(g)}}\right) \\
&\leq \frac{mP(g) + 2\sqrt{S_m(g)D}}{mP(g)}L(h \mid g) + \frac{9\sqrt{DS_m(g)}}{mP(g)} \\
&\leq L(h \mid g) + \frac{11\sqrt{DS_m(g)}}{S_m(g) - 2\sqrt{DS_m(g)}} \leq L(h \mid g) + 22\sqrt{\frac{D}{S_m(g)}}
\end{aligned}$$

where the last inequality follows from the fact that $S_m(g) \geq \frac{1}{4}\#_n(g) > 16D$. Similarly, we also have

$$\frac{\sqrt{S'_m(g)}}{mP(g)} \;\leq\; 2\sqrt{\frac{1}{S'_m(g)}}$$

and

$$\frac{S_m(g)}{mP(g)} \;\leq\; 2.$$

Putting it all together, we have

$$\begin{aligned}
L(Q \mid g) \;&\leq\; L(h \mid g) + 22\sqrt{\frac{D}{S_m(g)}} + \frac{2}{mP(g)}\left(\log(|\mathcal{G}||\mathcal{H}|) + \frac{1}{3}\log\frac{2|\mathcal{G}|}{\delta}\right) \\
&\qquad + \frac{\sqrt{S'_m(g)\log(|\mathcal{G}||\mathcal{H}|)}}{mP(g)} + \frac{2S_m(g)}{mP(g)}\sqrt{\frac{\log(|\mathcal{G}||\mathcal{H}|)}{S'_m(g)}} + \sqrt{\frac{1}{mP(g)}\log\frac{2|\mathcal{G}|}{\delta}} \\
&\leq\; L(h \mid g) + 22\sqrt{\frac{D}{S_m(g)}} + \frac{4D}{S_m(g)} + 2\sqrt{\frac{D}{S'_m(g)}} + 4\sqrt{\frac{D}{S'_m(g)}} + 2\sqrt{\frac{D}{S_m(g)}} \\
&\leq\; L(h \mid g) + 60\sqrt{\frac{D}{\#_n(g)}} + \frac{16D}{\#_n(g)}. \quad \square
\end{aligned}$$

D. Missing proofs from Section 5

D.1. Proof of Theorem 10

Let $\mathcal{F}$ denote the set of possible predictors produced by Algorithm 3. Since the only degrees of freedom enjoyed by Algorithm 3 are which classifier to assign to which group, we see that $|\mathcal{F}| \leq |\mathcal{G}||\mathcal{H}|$. Moreover, since each of the classifiers $\hat{h}_g$ are consistent with the data in their respective groups, we can conclude that f is consistent with all of the data. In particular, $L_n(f \mid g) = 0$ for all $g \in \mathcal{G}$. Theorem 12 then implies that with probability at least $1 - \delta$,

$$L(f \mid g) \;\leq\; \frac{16}{\#_n(g)}\left(2\log\left(|\mathcal{F}||\mathcal{H}|\right) + \log(8/\delta)\right) \;\leq\; \frac{16}{\#_n(g)}\left(2\log\left(|\mathcal{G}|^2|\mathcal{H}|\right) + \log(8/\delta)\right)$$

for all $g \in \mathcal{G}$.

D.2. Proof of Proposition 11

Let $\mathcal{X} = \{x_0, x_1, x_2, x_3\}$ with the following probabilities

$$\begin{aligned}
P((x_0, +1)) \;&=\; 3/24, \\
P((x_1, -1)) \;&=\; 7/24, \\
P((x_2, -1)) \;&=\; 7/24, \\
P((x_3, +1)) \;&=\; 7/24.
\end{aligned}$$

We also take $g_i = \{x_0, x_i\}$ for $i = 1, 2, 3$, and let $\mathcal{H} = \{h, h'\}$ where $h(x) = +1$ and $h'(x) = -1$ for all $x \in \mathcal{X}$.

Now any function $f \in \mathcal{F}$ corresponds to an assignment $\sigma : \mathcal{G} \to \mathcal{H}$ of groups to hypotheses. There are two cases to consider here.

Case 1: $\sigma(g_1) = \sigma(g_2) = h'$**.** In this case, we have that $f(x_0) = -1$. Thus,

$$L(f \mid g_3) \;\geq\; P(x_0 \mid g_3) \;=\; \frac{3}{10} \;=\; L(h \mid g) + \frac{3}{10}.$$

Case 2: $h \in \{\sigma(g_1), \sigma(g_2)\}$**.** Suppose without loss of generality that $\sigma(g_1) = h$. Then we have $f(x_1) = +1$. Thus,

$$L(f \mid g_1) \;\geq\; P(x_1 \mid g_1) \;=\; \frac{7}{10} \;=\; L(h' \mid g) + \frac{4}{10}.$$

E. Insufficiency of multiaccuracy

In the binary prediction setting where $\mathcal{Y} = \{0, 1\}$, Kim et al. (2019) use the definition that a function $f : \mathcal{X} \to [0, 1]$ is α-multiaccurate with respect to $\mathcal{C} \subset [-1, +1]^{\mathcal{X}}$ if

$$\mathbb{E}_{x,y}\left[c(x)(f(x) - y)\right] \;\leq\; \alpha$$

for all $c \in \mathcal{C}$. To simplify the discussion, let us assume that the label y is a deterministic function of the corresponding x, i.e. there exists $\eta \in \{0, 1\}^{\mathcal{X}}$ such that $\mathbb{E}[y \mid x] = \eta(x)$. In this setting, Kim et al. showed that multiaccuracy can be translated into a notion of multi-group learning as follows.

Proposition 19 (Proposition 1 from Kim et al. (2019))**.** *Let $\mathcal{C}$ and $S \subseteq \mathcal{X}$ be given, and suppose that $f : \mathcal{X} \to [0, 1]$ is α-multiaccurate with respect to $\mathcal{C}$. Further define $\hat{\eta}(x) = 1 - 2\eta(x)$. If there exists a $c \in \mathcal{C}$ such that*

$$\mathbb{E}_{x,y}[|c(x) - \hat{\eta}(x)\,\mathbb{1}[x \in S]|] \leq \tau,$$

then

$$\Pr_{x,y}\left(\operatorname{sign}(f(x) - 1/2) \neq \operatorname{sign}(y - 1/2) \mid x \in S\right) \;\leq\; \frac{2}{P(S)}\left(\alpha + \tau\right).$$

In the multi-group learning setup where $\mathcal{H} \subset [-1, +1]^{\mathcal{X}}$ and $\mathcal{G} \subset \{0, 1\}^{\mathcal{X}}$, we can take $\mathcal{C} = \{x \mapsto h(x)g(x) \mid h \in \mathcal{H}, g \in \mathcal{G}\}$. Proposition 19 tells us that α-multiaccuracy with respect to $\mathcal{C}$ implies

$$\Pr_{x,y}\left(\operatorname{sign}(f(x) - 1/2) \neq \operatorname{sign}(y - 1/2) \mid x \in g\right) \;\leq\; 4 \inf_{h \in \mathcal{H}} L(h \mid g) + \frac{2\alpha}{P(g)} \quad \text{for all } g \in \mathcal{G}, \tag{4}$$

where the loss in $L(\cdot \mid \cdot)$ is zero-one loss.

When each group has a corresponding low error classifier in $\mathcal{H}$, Eq. (4) tells us that multiaccuracy leads to reasonably good predictions across all groups. However, the bound in Eq. (4) devolves to no better than random guessing whenever $\inf_{h \in \mathcal{H}} L(h \mid g) \geq 1/8$. One may ask if this is due to some slack in the proof of Proposition 19 or if it is some intrinsic looseness associated with multiaccuracy. The following result shows that multiaccuracy reduction must result in at least some constant in front of $\inf_{h \in \mathcal{H}} L(h \mid g)$.

Proposition 20. *Suppose $|\mathcal{X}| \geq 3$ and let $\epsilon > 0$. There exist $\eta, f, g \in \{0, 1\}^{\mathcal{X}}$, $h \in \{-1, +1\}^{\mathcal{X}}$, and a marginal distribution over $\mathcal{X}$ such that*

- *f is 0-multiaccurate with respect to $\mathcal{C} = \{h \cdot g\}$,*
- *$\Pr_x\left(h(x) \neq \eta(x) \mid x \in g\right) = \epsilon$, but*
- *$\Pr_x\left(f(x) \neq \eta(x) \mid x \in g\right) \;=\; 2\epsilon$.*

Proof. Suppose $\mathcal{X} = \{x_0, x_1, x_2\}$ with $P(x_0) = 1 - 2\epsilon$ and $P(x_1) = P(x_2) = \epsilon$. Let $\eta(x) = g(x) = 1$ for all $x \in \mathcal{X}$ and define

$$f(x) \;=\; \begin{cases} 0 & \text{if } x \in \{x_1, x_2\} \\ 1 & \text{if } x = x_0 \end{cases} \qquad \text{and} \qquad h(x) \;=\; \begin{cases} 1 & \text{if } x \in \{x_0, x_1\} \\ -1 & \text{if } x = x_2 \end{cases}.$$

Then we can establish the following facts.

1. $\Pr_x\left(h(x) \neq \eta(x) \mid x \in g\right) = P(x_2) = \epsilon$.
2. $\Pr_x\left(f(x) \neq \eta(x) \mid x \in g\right) = 1 - P(x_0) = 2\epsilon$.

3. For $c = h \cdot g$, we have

$$\mathbb{E}[c(x)(f(x) - \eta(x))] \;=\; \mathbb{E}[h(x)f(x)] - \mathbb{E}[h(x)] \;=\; 1 - 2\epsilon - (1 - 2\epsilon) \;=\; 0.$$

Thus, f is 0-multiaccurate with respect to $\mathcal{C} = \{c\}$.

Combining all of the above gives the proposition. □

Thus, to get multi-group learning bounds of the form in Eq. (1) or Eq. (2), we must go beyond this type of simple application of multiaccuracy.