Omnipredictors for Constrained Optimization

Lunjia Hu [*1] **Inbal Rachel Livni Navon** [*1] **Omer Reingold** [*1] **Chutong Yang** [*1]

Abstract

The notion of omnipredictors (Gopalan, Kalai, Reingold, Sharan and Wieder ITCS 2022), suggested a new paradigm for loss minimization. Rather than learning a predictor based on a known loss function, omnipredictors can easily be post-processed to minimize any one of a rich family of loss functions compared with the loss of hypotheses in a class $\mathcal{C}$. It has been shown that such omnipredictors exist and are implied (for all convex and Lipschitz loss functions) by the notion of multicalibration from the algorithmic fairness literature. In this paper, we introduce omnipredictors for constrained optimization and study their complexity and implications. The notion that we introduce allows the learner to be unaware of the loss function that will be later assigned *as well as the constraints that will be later imposed*, as long as the subpopulations that are used to define these constraints are known. We show how to obtain omnipredictors for constrained optimization problems, relying on appropriate variants of multicalibration. We also investigate the implications of this notion when the constraints used are so-called group fairness notions.

1. Introduction

A predominant usage for outcome prediction is to inform the choice of a related action. Predicting the probability of a medical condition may help decide on a medical intervention or determine a life insurance premium rate. Predicting the probability of rain may help decide on the method of commuting to work or on a vacation destination or on wedding plans. For each possible action and outcome pair, there may be an associated loss – the cost of catching a cold while riding to work on a bike in the rain or perhaps the cost of changing a wedding venue at the last minute. A learning algorithm may try to come up with a hypothesis that determines an action to minimize an expected loss based on a particular loss function. The challenge in this prevalent paradigm of loss minimization is that different loss functions call for very different learning algorithms, which is problematic for a variety of reasons (e.g. multiple relevant loss functions or loss functions that are undetermined at the time of learning). The notion of omnipredictors that was introduced recently by Gopalan, Kalai, Reingold, Sharan and Wieder (Gopalan et al., 2022) provides a way to learn a single predictor that can be naturally post-processed (without access to data) to an action that minimizes any one of a very wide collection of loss functions. Gopalan et al. (2022) showed that omniprediction is implied by multicalibrated prediction, a notion introduced by Hébert-Johnson, Kim, Reingold and Rothblum in the algorithmic fairness literature (Hébert-Johnson et al., 2018).

While loss minimization is a natural goal, it may not be the only consideration in choosing an action. There may, for example, be capacity constraints (e.g. a limited number of vaccines) as well as fairness and diversity considerations. In this work, we introduce a notion of omniprediction that applies to the task of loss minimization *conditioned on a set of constraints*. For example, imagine we are deciding on which patients would receive a medical intervention when the budget for offering that intervention is limited (capacity constraint), or when we want this intervention to be assigned proportionally to the size of two subpopulations (statistical parity), or when we want the probability of receiving an intervention among patients who experience medical complications to be the same in two different subpopulations (equal opportunity). Our notion of omniprediction allows learning a single predictor that could be used to minimize a large collection of loss functions, even when arbitrary subsets of constraints are imposed from a rich family of constraints. We show how to formalize such a notion (exposing subtleties not existing in the original notion of omniprediction), how to obtain it using some variants of multicalibration, demonstrating that seeking an accurate depiction of the current world may be useful even when the final goal is a socially engineered action. Finally, we study the interaction between loss minimization and fairness constraints, showing that loss minimization has the potential to support fairness objectives.

[*]Equal contribution [1]Computer Science Department, Stanford University, Stanford, USA. Correspondence to: Lunjia Hu <lunjia@stanford.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

Unconstrained Omniprediction. We assume a distribution $\mathcal{D}$, over pairs (x, y), where $x \in X$ represents an individual, and y represents an outcome associated with x. For example, x is the attributes of a patient and y is whether that patient experienced a specific medical condition (in this paper, we will consider Boolean outcomes, i.e., $y \in \{0, 1\}$, but the notion could be generalized). We consider individual loss functions. A loss function ℓ is applied to an action a and an outcome y and signifies the loss $\ell(y, a)$ incurred when taking action a and observing outcome y (as we will discuss below, our results apply to a more general set of loss functions that can take into account membership of an individual in some predefined subpopulation).

The learning task of loss minimization is to learn a function c mapping individuals to actions such that the expected loss, $\mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(y, c(x))]$, is at least as small (up to some error term) as $\mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(y, c'(x))]$ for any function c' in a hypothesis class $\mathcal{C}$. Note that different loss functions may require different functions c and different learning algorithms to train them. The notion of omniprediction offers a way for a single algorithm to learn a predictor $p : X \to [0, 1]$ that allows optimizing any loss function in a rich family (e.g. all loss functions that are convex and κ-Lipschitz in the action). In this sense, p imitates the true probability predictor $p^* : X \to [0, 1]$ where $p^*(x) = \Pr_{\mathcal{D}}[y = 1 \mid x]$. Note that for every "nice" loss function, it is fairly easy to transform $p^*(x)$ to an action $a = \tau_\ell(p^*(x))$ that individually minimizes $\ell(y, a)$ (conditioned on x). Loosely, p is an $(\mathcal{L}, \mathcal{C})$-omnipredictor if for every $\ell \in \mathcal{L}$, applying τ_ℓ to p to get $c(x) = \tau_\ell(p(x))$ minimizes loss ℓ compared with the class $\mathcal{C}$. An omnipredictor resolves the aforementioned disadvantage of traditional loss minimization as it can be trained without knowledge of the specific loss function chosen and the loss function is only needed to decide on an action.

It has been shown in (Gopalan et al., 2022) that omniprediction is a somewhat surprising application of the notion of multicalibration, introduced by Hébert-Johnson et al. (2018) with the motivation of preventing unfair discrimination. Calibration roughly asks that every prediction value be accurate on average over the instances when the prediction value is given. Multicalibration asks a predictor to be calibrated not only over the entire population but also on many subpopulations (thus, a multicalibrated predictor cannot trade the accuracy of a relevant minority group for the benefit of the majority population). Ignoring some subtleties, a predictor p is $\mathcal{C}$-multicalibrated (up to error α) if for all $c \in \mathcal{C}$, $\sum_v \left|\mathbb{E}_{(x,y)\sim\mathcal{D}}[(y - v)c(x)\mathbf{1}(p(x) = v)]\right| \le \alpha$, where the summation is over v in the range of p (we assume the range is finite). It is shown in (Gopalan et al., 2022) that a $\mathcal{C}$-multicalibrated predictor is also an $(\mathcal{L}, \mathcal{C})$-omnipredictor for a wide class of loss functions (all convex and Lipschitz loss functions), and Gopalan et al. (2023) relax the multicalibration requirement to *calibrated multiaccuracy* when the loss functions have additional properties (e.g. when they are induced by generalized linear models). As we discuss in Appendix G, many previous algorithms can construct multiaccurate and multicalibrated predictors, and some of these algorithms have been implemented in real applications such as mortality risk prediction (Barda et al., 2020).

Constraints are Essential but Challenging. Omnipredictors constructed in previous work (Gopalan et al., 2022; 2023) allow us to efficiently solve various downstream loss minimization tasks. Each of these tasks aims to minimize the expectation of a loss function and beyond that the solutions to these tasks are not guaranteed to satisfy any nontrivial constraints. However, many loss minimization problems in practice naturally come with constraints that cannot be simply expressed as minimizing an expected loss $\mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(y, c(x))]$. For example, if an action $c(x)$ represents the amount of resources allocated to individual x, it is common to impose a budget constraint $\mathbb{E}[c(x)] \le B$ for an average budget B per individual. Other natural constraints come from the algorithmic fairness literature and are known as group fairness notions. Here, we assume that the entire set X of individuals is partitioned into t subpopulations (i.e., groups) $S_1, \dots, S_t$. Common examples of group fairness constraints include statistical parity ($\mathbb{E}[c(x)|x \in S_i]$ being approximately equal for every choice of $i = 1, \dots, t$), equal opportunity ($\mathbb{E}[c(x)|x \in S_i, y = 1]$ being approximately equal for every i), and equalized odds (for every $b = 0, 1$, the expectation $\mathbb{E}[c(x)|x \in S_i, y = b]$ being approximately equal for every i).

Constraints as basic as the budget constraint already impose challenges to the omniprediction results in previous work. This is because in previous work the final action $c(x) = \tau_\ell(p(x))$ is extremely local: it depends only on the loss function ℓ and the prediction $p(x)$ for that single individual x. Even if $p(x)$ equals the true conditional probability $\Pr_{\mathcal{D}}[y = 1|x]$, such local actions that completely ignore the marginal distribution over individuals and the predictions $p(x')$ for other individuals $x' \in X \setminus \{x\}$ cannot in general minimize the squared loss under even the simplest budget constraint (see Appendix A). While a loss function can be optimized for every individual separately, to determine whether an action $c(x)$ would violate the budget constraint, it is necessary to know the actions $c(x')$ assigned to other individuals $x' \in X \setminus \{x\}$. When constraints are present, omnipredictors are only possible when we allow more flexible ways of turning predictions into actions.

1.1. Our Contributions

We start by generalizing the powerful notion of omniprediction to more widely-applicable loss minimization tasks that have constraints.

Defining Omniprediction for Constrained Loss Minimization. We consider constrained loss minimization tasks in general forms, where every task has an objective function $f_0 : X \times A \times \{0,1\} \to \mathbb{R}$ and a collection of constraint functions $f_j : X \times A \times \{0,1\} \to \mathbb{R}$ indexed by $j \in J$. The goal of the task is to find an action function $c : X \to A$ that minimizes the objective $\mathbb{E}_{(x,y)\sim\mathcal{D}}[f_0(x, c(x), y)]$ while satisfying the constraints $\mathbb{E}_{(x,y)\sim\mathcal{D}}[f_j(x, c(x), y)] \le 0$ for every $j \in J$. Results in this paper extend to more general tasks where we use an arbitrary Lipschitz function to combine constraints as well as objectives (Appendix E).

Following previous work, for a class $\mathcal{T}$ of tasks and a class $\mathcal{C}$ of hypotheses $c : X \to A$, we say a predictor $p : X \to [0,1]$ is an omnipredictor if it allows us to "efficiently solve" any task $T \in \mathcal{T}$ compared to the hypotheses in $\mathcal{C}$. More specifically, in our constrained setting, an omnipredictor p allows us to "efficiently produce" a good action function $c : X \to A$ for any task $T \in \mathcal{T}$ such that c approximately satisfies all the constraints in T, and the objective achieved by c does not exceed (up to a small error) the objective of any $c' \in \mathcal{C}$ *that satisfy all the constraints of* T.

A key challenge in formalizing omniprediction for constrained loss minimization is to specify the procedure of "efficiently turning" a predictor $p : X \to [0,1]$ into an action function $c : X \to A$ for a specific task $T \in \mathcal{T}$. As discussed earlier, previous work only allows $c(x)$ to be $\tau(p(x))$ for a transformation function τ that only depends on T, and this local transformation is not sufficient in our constrained setting. We need more flexible transformations, and we also need to maintain the efficiency of such transformations. We solve this challenge by examining the semantics behind the transformation $\tau(p(x))$ in previous work: this transformation corresponds to solving the task T optimally while pretending that $p(x)$ is the true conditional probability $\Pr_{\mathcal{D}}[y = 1|x]$. We thus use transformations induced by solving the task on a *simulated distribution* defined by p in our definition of omniprediction (Definition 2.1). We show that this not only makes omniprediction possible for constrained problems, but also maintains the efficiency of the transformation. Moreover, as we discuss below, we can construct omnipredictors for important families of constrained loss minimization problems from group-wise variants of the multiaccuracy and/or multicalibration conditions. Note that conditions such as multiaccuracy and multicalibration are already needed in previous omniprediction results that do not handle constraints!

Constructing Omnipredictors for Group Objectives and Constraints. We develop omnipredictors for an important class of constrained loss minimization tasks, namely, tasks with *group objectives and constraints*. Here, as in many problems in the fairness literature, we assume that the set X of individuals is partitioned into t groups $S_1, \ldots, S_t$, and we let $g : X \to [t]$ denote the group partition function, i.e., $g(x) = i$ if and only if $x \in S_i$. We say an objective/constraint function $f : X \times A \times \{0,1\} \to \mathbb{R}$ is a *group constraint* if there exists $f' : [t] \times A \times \{0,1\} \to \mathbb{R}$ such that $f(x,a,y) = f'(g(x), a, y)$ for every $(x,a,y) \in X \times A \times \{0,1\}$. Tasks with group objectives and constraints are significantly more general than unconstrained tasks in previous work with a loss function $\ell(y,a)$ that does not depend on the individual x at all.

In Section 4, we show that omnipredictors for loss minimization problems with group objectives and constraints can be obtained from group-wise multiaccuracy and/or multicalibration conditions. Here, group-wise multiaccuracy and multicalibration require the predictor to satisfy multiaccuracy and multicalibration when conditioned on every group S_i (see Section 2.3 for formal definitions). Specifically, we show the following results from the simplest setting to more challenging ones:

1. We start by considering a simple but general class of objectives/constraints that are *convex and special* (Definition 4.2). Objectives in this class include the common ℓ_1 loss, the squared loss, loss induced by generalized linear models (up to scaling), and group combinations of these loss functions (e.g. each group chooses the ℓ_1 or the squared loss). Constraints in this class include budget constraints and group fairness constraints such as statistical parity, equal opportunity, and equalize odds. In Theorem 4.4, we show that omnipredictors for tasks with convex and special group objectives and constraints can be obtained from group multiaccuracy w.r.t. the hypothesis class $\mathcal{C}$ plus group calibration. This generalizes the results in (Gopalan et al., 2023) to our constrained and multi-group setting.

2. In Theorem 4.6, we show that for general convex and Lipschitz group objectives and constraints, we can construct omnipredictors from group multicalibration w.r.t. $\mathcal{C}$. This generalizes the results in (Gopalan et al., 2022) to our constrained and multi-group setting.

3. In Theorem 4.7, we show that for general (non-convex) group objectives and constraints, omnipredictors can be obtained from group calibration plus group *level-set* multiaccuracy w.r.t. $\mathcal{C}$, namely, being accurate in expectation over individuals $x \in S_i$ with $c(x) = a$ for every group i, hypothesis $c \in \mathcal{C}$, and action a.

We provide counterexamples in Appendix I to show that it is necessary to strengthen multiaccuracy/multicalibration to their group-wise and occasionally level-set variants in our constrained setting.

We prove all our omniprediction results in a unified and streamlined fashion using Lemma 3.1. Previously, Gopalan et al. (2023) also aim to build a unified framework for omnipredictors using the notion of outcome indistinguishability (Dwork et al., 2021). While the initial omniprediction result in (Gopalan et al., 2022) requires multicalibration (as an unconstrained special case of our Theorem 4.6), Gopalan et al. (2023) only require a weaker calibrated multiaccuracy condition (as an unconstrained special case of our Theorem 4.4) and they provide a simpler and more structured analysis than Gopalan et al. (2022). However, the result in Gopalan et al. (2023) requires the loss functions to satisfy additional properties (we call such loss functions *special objectives* in Definition 4.2), and it particularly focuses on loss functions induced by generalized linear models. That is, Gopalan et al. (2023) fall short of providing a simple analysis that fully reconstructs the result in Gopalan et al. (2022). By proving Theorems 4.4 and 4.6, we show that our streamlined analysis using Lemma 3.1 can not only reconstruct the results in Gopalan et al. (2022; 2023), but also generalize them to the constrained setting.

Loss Minimization Can Augment Fairness. When solving an optimization task T using an omnipredictor p, for fairness and interpretability reasons, it is natural to require the solution c to be *rank-preserving*. That is, we require $c(x) \geq c(x')$ when $p(x) \geq p(x')$. For example, this could mean that we grant higher loans to individuals predicted more likely to repay it. A violation of the rank-preserving property corresponds to granting excessive loans to people that are likely to default on it, which causes harm to these individuals as well as the ones that deserve the loans more. With group constraints, it makes more sense to only require ranks to be preserved within each group, i.e., for individuals x, x' satisfying $g(x) = g(x')$. This is a necessary relaxation, as group fairness constraints aim to increase opportunities for individuals from certain groups (e.g. to rectify historical discrimination), which would not always preserve ranks between individuals from different groups. However, some unreasonable objectives would incentivize the solution to be not rank-preserving even within a group. For example, an unreasonable objective could be $f(x, a, y) = 1 - |a - y|$ for all $x \in S_i$, and $f(x, a, y) = |a - y|$ for all other x, assuming the actions a are in $[0, 1]$ after scaling. This objective incentivizes giving loans to individuals in S_i that are likely to default on it, instead of those that are likely to repay it. A group fairness constraint, such as parity, can enforce giving a fair total amount of loan to the individuals in S_i but cannot promise that the loans are given to those predicted to be more likely to repay it. This limitation of group fairness notions has been repeatedly demonstrated (cf. (Dwork et al., 2012) for an early example), and often abuses of these notions lead to violations of the rank-preserving requirement as in the example. In Section 5, we formally study the conditions of the objective and constraints under which we can ensure that the solution c obtained from an omnipredictor p is rank-preserving within every group.

1.2. Related Work

Loss minimization under fairness or other constraints is a rich research area. For any given fairness definition, it is natural to ask how to learn under the corresponding constraints and how to minimize loss (or maximize utility). This has been studied for various group notions of fairness (cf (Zafar et al., 2017b)) but also for more refined notions such as metric fairness and multi-group metric fairness (Dwork et al., 2012; Rothblum & Yona, 2018; Kim et al., 2018). A common approach to combining loss minimization with fairness constraints is to add a fairness regularizer to the risk minimization (Donini et al., 2018; Kamishima et al., 2012; Zafar et al., 2017b). Non-convex constraints have been considered in (Cotter et al., 2019). Accordingly, they also formulate the problem as a non-convex optimization problem which may be hard to solve. There is also a line of empirical work on loss minimization with fairness constraints (Zemel et al., 2013; Zafar et al., 2017a; Goh et al., 2016). Finally, some recent related works focus on other learning setting under fairness constraint, like learning policies (Nabi et al., 2019), online learning (Bechavod & Roth, 2022), federated learning (Hu et al., 2022b), and ranking (Dwork et al., 2019).

A key difference between our work and most previous work on loss minimization is that we aim for learning a single predictor that can efficiently solve a variety of downstream constrained loss minimization tasks. Moreover, as we do not make any assumption on the true data distribution $\mathcal{D}$, we consider it infeasible to learn the distribution $\mathcal{D}$ entirely and we only require conditions such as multicalibration that can be much easier to achieve using existing algorithms in the literature. Some works, such as (Celis et al., 2019; Agarwal et al., 2018; Narasimhan, 2018; Sharifi-Malvajerdi et al., 2019), can deal with multiple loss minimization tasks but they require approximately learning the true distribution $\mathcal{D}$ within a small total variation distance or approximately learning the true labels.

In an influential paper, Hardt, Price and Srebro (Hardt et al., 2016) propose equalized odds and equal opportunity as group notions of fairness. They give methods of post-processing a predictor to enforce these constraints while minimizing loss. They show optimality compared with solutions that can be obtained from post-processing the predictor, whereas in this work we directly aim for optimality with respect to a rich pre-specified hypothesis class $\mathcal{C}$. We consider more general loss functions with real-valued actions compared to the loss functions in (Hardt et al., 2016) that only take binary values as input, and we also consider more general constraints beyond the group fairness constraints in (Hardt et al., 2016).

Rothblum & Yona (2021) use the notion of outcome indistinguishability (Dwork et al., 2021), closely related to multicalibration, to obtain loss minimization, not only on the entire population but also on many subpopulations. Their approach relies on a locality property of the loss function which they term *f-proper*. When this property is satisfied, for every fixed individual $x_0 \in X$, the optimal action $c(x_0)$ for that individual x_0 only depends on $\mathbb{E}[y|x = x_0]$ and not on $\mathbb{E}[y|x = x_1]$ for other individuals $x_1 \in X \setminus \{x_0\}$. In our constrained setting, this locality property fails to hold: to satisfy a group constraint, the action $c(x_0)$ must coordinate with the actions $c(x_1)$ for other individuals x_1 in or out of the group/subpopulation of x_0.

Independently of our work, Globus-Harris et al. (2022) also study the problem of solving downstream tasks by post-processing multicalibrated predictors. They focus on the 0-1 loss for classification tasks and thus their results do not imply the full power of omnipredictors that handle arbitrary loss functions from a rich family. They also focus on a few specific group fairness constraints, whereas we consider more general classes of constraints. By assuming multicalibration with respect to delicately-designed classes, their predictors can be efficiently post-processed to satisfy constraints on *intersecting* groups. Again independently of our work, Kim & Perdomo (2023) study omniprediction in an (unconstrained) performative setting, where the distribution of the outcome y of an individual x can change based on the action $c(x)$.

1.3. Limitations and Social Impacts

While our work is theoretical, we view it as giving a foundation and proof-of-concept for potential omnipredictors to be deployed in the real world with fairness considerations. Omnipredictors allow us to efficiently solve optimization problems and adapt to rich families of objectives and constraints, but carelessly choosing constraints and objectives for the omnipredictors may not always lead to good decisions. In some situations, different fairness constraints can lead to contradictory fairness guarantees, and choosing a wrong fairness constraint may lead to inappropriate actions. Our results in Section 5 are motivated by situations where fairness constraints alone are not enough for acceptable actions and a good objective is needed to augment the fairness constraints for arguably fairer actions. In terms of limitations of the results, our omnipredictors rely on a fixed group partition g and a fixed (unknown) distribution $\mathcal{D}$, and it remains an interesting question to construct omnipredictors that can adapt to changes in g and $\mathcal{D}$ as well. Addressing this limitation could help protect new and evolving subpopulations. It is an interesting question whether our techniques (in particular Lemma 3.1) can be applied to tasks with more general outcomes beyond binary outcomes $y \in \{0, 1\}$. Unconstrained versions of such tasks have been considered by Gopalan et al. (2022), and we leave it for future work to generalize their results to the constrained setting.

2. Problem Setup

Throughout the paper, we use X to denote a non-empty set of individuals, and use $\mathcal{D}$ to denote a distribution over $X \times \{0, 1\}$. We use A to denote a non-empty set of actions, and use $c : X \to A$ to denote an action function that assigns an action $c(x)$ to every individual $x \in X$ (e.g. hiring the individual or not). We occasionally consider a *randomized* action function $c : X \to \Delta_A$ that assigns every individual $x \in X$ a *distribution* $c(x) \in \Delta_A$ over actions in A. For generality we sometimes only make statements about randomized action functions, where one should view a deterministic action function $c : X \to A$ as the randomized action function $c' : X \to \Delta_A$ where $c'(x) \in \Delta_A$ is the degenerate distribution supported on $c(x)$ for every $x \in X$.

2.1. Constrained Loss Minimization Tasks

Given a loss function $f_0 : X \times A \times \{0, 1\} \to \mathbb{R}$ and a collection of constraints $f_j : X \times A \times \{0, 1\} \to \mathbb{R}$ indexed by $j \in J$, we define a constrained loss minimization task T to be the following optimization problem:

$$\begin{aligned} \underset{c:X\to A}{\text{minimize}} \quad & \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f_0(x, c(x), y) \\ \text{s.t.} \quad & \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f_j(x, c(x), y) \le 0 \quad \text{for every } j \in J. \end{aligned} \tag{1}$$

It is often challenging to solve a task T optimally, and we need to consider approximate and potentially randomized solutions. For $\beta \in \mathbb{R}$ and $\varepsilon \in \mathbb{R}_{\ge 0}$, we define $\mathsf{sol}_{\mathcal{D}}(T, \beta, \varepsilon)$ to be the set of randomized action functions $c : X \to \Delta_A$ satisfying

$$\begin{aligned} &\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_0(x, a, y) \le \beta, \quad \text{and} \\ &\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_j(x, a, y) \le \varepsilon \quad \text{for every } j \in J. \end{aligned}$$

For a class $\mathcal{C}$ of functions $c : X \to \Delta_A$, we define

$$\mathsf{opt}_{\mathcal{D}}(T, \mathcal{C}, \varepsilon) := \inf\{\beta \in \mathbb{R} : \mathcal{C} \cap \mathsf{sol}_{\mathcal{D}}(T, \beta, \varepsilon) \ne \emptyset\}.$$

Note that $\mathsf{opt}_{\mathcal{D}}(T, \mathcal{C}, \varepsilon)$ may take any value in $\mathbb{R} \cup \{\pm\infty\}$, where we define $\inf \emptyset = +\infty$. In Appendix E, we show how results in this paper extend to more general tasks where we combine constraints and objectives using arbitrary Lipschitz functions.

2.2. Omnipredictors for Constrained Loss Minimization

An omnipredictor, as introduced by Gopalan et al. (2022), allows us to solve a family of downstream loss minimization tasks without training a different model from scratch for every task in the family. Previous work focuses on omnipredictors for unconstrained loss minimization (Gopalan et al., 2022; 2023). We generalize this notion to constrained loss minimization as follows.

For a distribution $\mathcal{D}$ over $X \times \{0,1\}$ and a predictor $p : X \to [0,1]$, we define the *simulated distribution* $\mathcal{D}_p$ to be the distribution of $(x, y') \in X \times \{0,1\}$ where we first draw (x,y) from $\mathcal{D}$ and then draw y' from the Bernoulli distribution $\mathsf{Ber}(p(x))$ with mean $p(x)$. For a hypothesis class $\mathcal{C}$ consisting of functions $c : X \to \Delta_A$, suppose we want to solve a downstream constrained loss minimization task T on the true distribution $\mathcal{D}$ and we want a comparable or better solution than the best hypothesis $c \in \mathcal{C}$. An omnipredictor p should allow us to achieve this goal by finding an approximately optimal solution c' from another, ideally simpler, hypothesis class $\mathcal{C}'$ for the same task T but on the simulated distribution $\mathcal{D}_p$ defined by the omnipredictor p. Such an omnipredictor is particularly powerful when it works for tasks T from a rich family $\mathcal{T}$ and when solving any $T \in \mathcal{T}$ on the simulated distribution $\mathcal{D}_p$ over hypothesis class $\mathcal{C}'$ is significantly easier than directly solving T on the true distribution $\mathcal{D}$ over hypothesis class $\mathcal{C}$. This leads to the following formal definition of an omnipredictor:

Definition 2.1. Let $\mathcal{D}$ be a distribution over $X \times \{0,1\}$ and $\varepsilon \geq 0$ be a parameter. Let $\mathcal{T}$ be a collection of constrained loss minimization tasks and let $p : X \to [0,1]$ be a predictor. For classes $\mathcal{C}, \mathcal{C}'$ of functions $c : X \to \Delta_A$, we say p is a $(\mathcal{T}, \mathcal{C}, \mathcal{C}', \varepsilon)$-omnipredictor on $\mathcal{D}$ if the following holds for any $T \in \mathcal{T}$. Defining $\beta := \mathsf{opt}_{\mathcal{D}}(T, \mathcal{C}, 0) \in \mathbb{R}$ and $\beta' := \mathsf{opt}_{\mathcal{D}_p}(T, \mathcal{C}', \varepsilon/3) \in \mathbb{R}$, we have

$$\mathcal{C}' \cap \mathsf{sol}_{\mathcal{D}_p}(T, \beta' + \varepsilon/3, 2\varepsilon/3) \subseteq \mathsf{sol}_{\mathcal{D}}(T, \beta + \varepsilon, \varepsilon).$$

Suppose we have an omnipredictor p as in the definition above, and we want to solve an arbitrary constrained loss minimization task $T \in \mathcal{T}$ in comparison with the class $\mathcal{C}$, i.e., we want to find a solution in $\mathsf{sol}_{\mathcal{D}}(T, \beta + \varepsilon, \varepsilon)$. Instead of collecting data points from $\mathcal{D}$ and solve the task from scratch, we just need to find a solution in $\mathcal{C}' \cap \mathsf{sol}_{\mathcal{D}_p}(T, \beta' + \varepsilon/3, 2\varepsilon/3)$, i.e., a solution $c' \in \mathcal{C}'$ that approximately solves the task on the simulated distribution $\mathcal{D}_p$. This is usually much easier than solving the task on the original distribution $\mathcal{D}$ for the following two reasons:

Simplicity from $\mathcal{D}_p$. First, since we know p, we know the conditional distribution of y given x in $(x,y) \sim \mathcal{D}_p$, and thus the only unknown part about $\mathcal{D}_p$ is the marginal distribution of x, which can be learned from *unlabeled* data drawn from $\mathcal{D}$ (i.e., examples of x in $(x,y) \sim \mathcal{D}$ with y concealed). For all the omniprediction results in this paper, we assume that the downstream tasks have a group structure specified by a fixed group partition function $g : X \to [t]$ (see Section 4). To solve such tasks on the simulated distribution $\mathcal{D}_p$, all we need to know about the marginal distribution of x is the probability that x belongs to each of a few subsets defined independently of the actual downstream task. We can estimate these probabilities when we train the omnipredictor p, and no additional data (labeled or not) is needed at all when we use p to solve downstream tasks (see Appendix H).

Simplicity from $\mathcal{C}'$. Second, solving downstream tasks over the new class $\mathcal{C}'$ can be computationally much more efficient than solving them over the original class $\mathcal{C}$. In all of the omniprediction results in this paper, we choose $\mathcal{C}'$ to be very simple (as $\mathcal{C}_{p,g}$ and $\mathcal{C}_{p,g}^{\mathsf{rand}}$ in Definition 4.3) so that its complexity depends on the number of groups and the size of the range of p, which can be made to be very small ($O(1/\varepsilon)$), whereas $\mathcal{C}$ can be significantly more complex. Specifically, every function in $\mathcal{C}_{p,g}$ (resp. $\mathcal{C}_{p,g}^{\mathsf{rand}}$) assigns the same action (resp. same distribution over actions) to individuals x with the same $p(x)$ and $g(x)$. In Appendix H we give very efficient algorithms for solving constrained loss minimization tasks given omnipredictors.

In previous work on omniprediction without constraints, the optimal solution c on the simulated distribution $\mathcal{D}_p$ is trivial to find: it is given by choosing $c(x)$ so that $\mathbb{E}_{y \sim \mathsf{Ber}(p(x))} f_0(x, c(x), y)$ is minimized (Bayes optimal solution). That is, the optimal $c(x)$ depends only on x, f_0, and $p(x)$ (often f_0 does not depend on x and thus $c(x)$ only depends on f_0 and $p(x)$). Because of this locality property, previous definitions of omniprediction for *unconstrained* loss minimization simply explicitly uses the optimal solution on the simulated distribution $\mathcal{D}_p$ without defining a task on $\mathcal{D}_p$ or even without defining $\mathcal{D}_p$ at all. Our Definition 2.1 not only generalizes these previous definitions, but also deals with more challenging tasks with constraints where the locality property fails to hold.

2.3. Group Multiaccuracy and Multicalibration

A main contribution of this paper is showing that omnipredictors for a variety of constrained loss minimization problems can be obtained from group-wise multiaccuracy and/or multicalibration conditions. The notions of multiaccuracy and multicalibration are introduced by Hébert-Johnson et al. (2018) and Kim et al. (2019), and there are many algorithms for achieving these notions in previous work (see Appendix G). We define these notions here as special cases of the following generalized multicalibration notion. For the definitions below, we assume $\mathcal{D}$ is a distribution over $X \times \{0,1\}$ and $\varepsilon \geq 0$ is a parameter.

Definition 2.2 (Generalized multicalibration (GenMC) (see e.g. Kim et al., 2022, Definition 1.1 in Supplementary Information))**.** Let W be a class of functions $w : X \times [0,1] \to \mathbb{R}$. We say a predictor $p : X \to [0,1]$ satisfies (W, ε)-generalized multicalibration w.r.t. distribution $\mathcal{D}$ if

$$\left| \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} [(y - p(x)) w(x, p(x))] \right| \leq \varepsilon \text{ for every } w \in W.$$

For simplicity, we additionally require the range of p, $\mathsf{range}(p) := \{p(x) : x \in X\}$, to be a *finite* subset of $[0,1]$.[1] We use $\mathsf{GenMC}_{\mathcal{D}}(W, \varepsilon)$ to denote the set of predictors p satisfying the conditions above.

We define multiaccuracy and multicalibration below as special cases of GenMC in a general group-wise setting, by choosing an appropriate function class W in every definition. Here, we assume that the set X of individuals is partitioned into t groups (i.e., subpopulations). We use $g : X \to [t]$ to denote the group partition function that assigns every individual $x \in X$ a group index $g(x) \in [t] := \{1, \ldots, t\}$.

Definition 2.3 (Group Multiaccuracy (GrpMA))**.** For a class H of functions $h : X \to \mathbb{R}$ and group index $g : X \to [t]$, we define $\mathsf{GrpMA}_{\mathcal{D}}(H, g, \varepsilon)$ to be the set $\mathsf{GenMC}_{\mathcal{D}}(W, \varepsilon)$ where W consists of all functions $w : X \times [0,1] \to \mathbb{R}$ such that there exist $h \in H$ and $\tau : [t] \to [-1,1]$ satisfying $w(x,v) = h(x)\tau(g(x))$ for every $(x,v) \in X \times [0,1]$. We say a predictor p is (H, g, ε)-multiaccurate w.r.t. distribution $\mathcal{D}$ if $p \in \mathsf{GrpMA}_{\mathcal{D}}(H, g, \varepsilon)$. When the distribution $\mathcal{D}$ is clear from context, we often drop it and write $\mathsf{GrpMA}(H, g, \varepsilon)$ (similarly for other definitions below).

In Appendix B we give an equivalent definition of GrpMA in a form closer to similar definitions in the literature. We do this for other definitions below in this section as well.

Definition 2.4 (Group Multicalibration (GrpMC))**.** For a class H of functions $h : X \to \mathbb{R}$ and group index $g : X \to [t]$, we define $\mathsf{GrpMC}_{\mathcal{D}}(H, g, \varepsilon)$ to be the set $\mathsf{GenMC}_{\mathcal{D}}(W, \varepsilon)$ where W consists of all functions $w : X \times [0,1] \to \mathbb{R}$ such that there exist $h \in H$ and $\tau : [t] \times [0,1] \to [-1,1]$ satisfying $w(x,v) = h(x)\tau(g(x), v)$ for every $(x,v) \in X \times [0,1]$. We say a predictor p is (H, g, ε)-multicalibrated w.r.t. distribution $\mathcal{D}$ if $p \in \mathsf{GrpMC}_{\mathcal{D}}(H, g, \varepsilon)$.

The following definition of group calibration is a special case of group multicalibration where H only contains the constant function h that maps every $x \in X$ to 1:

Definition 2.5 (Group Calibration (GrpCal))**.** We define $\mathsf{GrpCal}_{\mathcal{D}}(g, \varepsilon)$ to be the set $\mathsf{GenMC}_{\mathcal{D}}(W, \varepsilon)$ where W consists of all functions $w : X \times [0,1] \to [-1,1]$ such that there exists $\tau : [t] \times [0,1] \to [-1,1]$ satisfying $w(x,v) = \tau(g(x), v)$ for every $(x,v) \in X \times [0,1]$. We say a predictor p is (g, ε)-calibrated w.r.t distribution $\mathcal{D}$ if $p \in \mathsf{GrpCal}_{\mathcal{D}}(g, \varepsilon)$.

The following definition is a variant of group multiaccuracy where the transformation τ also takes the function value $h(x)$ as input, and we view individuals x with the same $h(x)$ as belonging to the same level set of h.

Definition 2.6 (Group Level-Set Multiaccuracy (GrpLMA))**.** For an arbitrary finite set A and a class H of functions $h : X \to A$, we define $\mathsf{GrpLMA}_{\mathcal{D}}(H, g, \varepsilon)$ to be the set $\mathsf{GenMC}_{\mathcal{D}}(W, \varepsilon)$ where W consists of all functions $w : X \times [0,1] \to [-1,1]$ such that there exist $h \in H$ and $\tau : [t] \times A \to [-1,1]$ satisfying $w(x,v) = \tau(g(x), h(x))$ for every $(x,v) \in X \times [0,1]$. We say a predictor p is (H, g, ε)-level-set multiaccurate w.r.t distribution $\mathcal{D}$ if $p \in \mathsf{GrpLMA}_{\mathcal{D}}(H, g, \varepsilon)$.

When the group partition function g is a constant function g_0 that assigns every individual to the same group, we recover notions in the standard single-group setting: multiaccuracy ($\mathsf{MA}_{\mathcal{D}}(H, \varepsilon) := \mathsf{GrpMA}_{\mathcal{D}}(H, g_0, \varepsilon)$), multicalibration ($\mathsf{MC}_{\mathcal{D}}(H, \varepsilon) := \mathsf{GrpMC}_{\mathcal{D}}(H, g_0, \varepsilon)$), and calibration ($\mathsf{Cal}_{\mathcal{D}}(\varepsilon) := \mathsf{GrpCal}_{\mathcal{D}}(g_0, \varepsilon)$).

3. Our Approach

We describe our general approach for constructing and analyzing omnipredictors for constrained loss minimization tasks. Our approach is similar in spirit to the outcome indistinguishability perspective taken by (Gopalan et al., 2023), but our approach is more general: it takes constraints into account and can also be applied to reconstruct the results in previous papers on omnipredictors (Gopalan et al., 2022; 2023). In particular, we overcome the limitation of (Gopalan et al., 2023) that it falls short of fully explaining the initial omnipredictors results in (Gopalan et al., 2022). Our approach is based on the following key lemma:

Lemma 3.1. *Let $\mathcal{D}$ be a distribution over $X \times \{0,1\}$ and $\varepsilon \geq 0$ be a parameter. Let $\mathcal{T}$ be a collection of constrained loss minimization tasks and let $\mathcal{C}, \mathcal{C}'$ be classes of functions $c : X \to \Delta_A$. If a predictor p satisfies the following two properties for every $T \in \mathcal{T}$, then p is a $(\mathcal{T}, \mathcal{C}, \mathcal{C}', \varepsilon)$-omnipredictor on $\mathcal{D}$:*

1. *Let f_0 be the loss function of T and $(f_j)_{j \in J}$ be the constraints of T. For every $c \in \mathcal{C}$, there exists $c' \in \mathcal{C}'$ such that for every $j \in \{0\} \cup J$,*

$$\begin{aligned} &\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c'(x)} f_j(x,a,y) \\ &\leq \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_j(x,a,y) + \varepsilon/3. \end{aligned} \tag{2}$$

2. *For every $c \in \mathcal{C}'$ and every $j \in \{0\} \cup J$,*

$$\begin{aligned} &\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_j(x,a,y) \\ &\leq \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c(x)} f_j(x,a,y) + \varepsilon/3. \end{aligned} \tag{3}$$

Lemma 3.1 reduces the task of constructing an omnipredictor to satisfying the conditions in (2) and (3). We prove

[1] As we discuss in Appendix H, a simple discretization allows us to get a predictor p with $\mathsf{range}(p) \subseteq \{0, 1/m, 2/m, \ldots, 1\}$ for $m = O(1/\varepsilon)$ that satisfy all the group multiaccuracy and multicalibration requirements we need for our results. Also, all previous algorithms for achieving multicalibration naturally produce predictors with such discrete ranges.

Lemma 3.1 in Appendix C and show how to apply it to construct omnipredictors for a variety of constrained loss minimization tasks in Section 4. Lemma 3.1 allows us to give short and streamlined proofs for all our results in Section 4, and these results generalize previous results in (Gopalan et al., 2022; 2023) as special cases.

4. Omnipredictors from Group Multiaccuracy and Multicalibration

In this section, we apply Lemma 3.1 and show that we can obtain omnipredictors for loss minimization tasks with group objectives and constraints from group multiaccuracy and/or multicalibration conditions. Here, we assume that the individual set X is partitioned into t groups by a group partition function $g : X \to [t]$ assigning a group index $g(x) \in [i]$ to every individual $x \in X$.

Definition 4.1. For a group partition $g : X \to [t]$, we say an objective/constraint function $f : X \times A \times \{0,1\} \to \mathbb{R}$ is a group objective/constraint if there exists $f' : [t] \times A \times \{0,1\} \to \mathbb{R}$ such that $f(x,a,y) = f'(g(x),a,y)$ for every $(x,a,y) \in X \times A \times \{0,1\}$.

Proofs for the results in this section are deferred to Appendix D. These results show that algorithms in previous work for achieving multiaccuracy and multicalibration allow us to obtain omnipredictors even when constraints are imposed on the loss minimization tasks. We discuss these algorithms in more detail in Appendix G.

We start with a basic case where the objectives and constraints are convex and special, defined below. We use $\partial f(x,a)$ to denote $f(x,a,1) - f(x,a,0)$.

Definition 4.2. Let the action set $A \subseteq \mathbb{R}$ be an interval. We say an objective/constraint function $f : X \times A \times \{0,1\} \to \mathbb{R}$ is convex if $f(x,\cdot,y)$ is convex for every fixed $(x,y) \in X \times \{0,1\}$. We say f is special w.r.t a group partition $g : X \to [t]$ if there exist $\tau_1, \tau_2 : [t] \to [-1,1]$ such that $\partial f(x,a) = \tau_1(g(x)) + \tau_2(g(x))a$.

Examples of convex and special group objectives when $A = [0,1]$ include the ℓ_1 loss $f(x,a,y) = |a-y|/2$, the squared loss $f(x,a,y) = (a-y)^2/2$, and group-wise combinations of them (every group chooses either ℓ_1 or squared loss). As demonstrated in (Gopalan et al., 2023), loss functions induced from generalized linear models are also special after appropriate scaling. Examples of convex and special constraints include all *linear constraints*, i.e., constraint functions f for which there exist $\tau_1, \tau_2 : [t] \to [-1,1]$ and $\tau_3, \tau_4 : [t] \to \mathbb{R}$ such that

$$f(x,a,y) = \tau_1(i)y + \tau_2(i)ay + \tau_3(i) + \tau_4(i)a \quad (4)$$

for every $(x,a,y) \in X \times A \times \{0,1\}$ where $i := g(x)$. Linear constraints are general enough to express fairness constraints such as statistical parity, equal opportunity (equal true positive rates), and equalized odds (equal true positive rates and equal false positive rates) as follows. For every group $i \in [t]$, define $r_i := \Pr[g(x) = i]$, $r_i^+ := \Pr[g(x) = i | y = 1]$, and $r_i^- := \Pr[g(x) = i | y = 0]$. These fairness constraints can be expressed as[2]

$$\mathbb{E}[\mathbf{1}(g(x) = i)c(x)] = r_i \, \mathbb{E}[c(x)], \quad \text{(statistical parity)}$$

$$\mathbb{E}[\mathbf{1}(g(x) = i)c(x)y] = r_i^+ \, \mathbb{E}[c(x)y], \quad \text{(equal true positive rates)}$$

$$\mathbb{E}[\mathbf{1}(g(x) = i)c(x)(1-y)] = r_i^- \, \mathbb{E}[c(x)(1-y)]. \quad \text{(equal false positive rates)}$$

Each of the above fairness constraints can be written as $\mathbb{E}[f(x,c(x),y)] = 0$ for an appropriate f satisfying (4). For example, for statistical parity, we choose f as follows:

$$f(x,a,y) = \mathbf{1}(g(x) = i)a - r_i a. \quad \text{(statistical parity)}$$

Moreover, we can express *approximate* fairness constraints as a combination of linear constraints because $|\mathbb{E}[f(x,c(x),y)]| \le \alpha$ is equivalent to $\mathbb{E}[f(x,c(x),y) - \alpha] \le 0$ and $\mathbb{E}[-f(x,c(x),y) - \alpha] \le 0$.

For tasks with group objectives/constraints, we often choose the class $\mathcal{C}'$ in our definition of omnipredictors (Definition 2.1) to be $\mathcal{C}_{p,g}$ and $\mathcal{C}_{p,g}^{\mathsf{rand}}$ in the following definition:

Definition 4.3. For an action set A, a group partition function $g : X \to [t]$ and a predictor $p : X \to [0,1]$, we define $\mathcal{C}_{p,g}$ to be the class consisting of all functions $c : X \to A$ such that there exists $\tau : [t] \times [0,1] \to A$ satisfying $c(x) = \tau(g(x), p(x))$ for every $x \in X$. We define $\mathcal{C}_{p,g}^{\mathsf{rand}}$ to be the class consisting of all functions $c : X \to \Delta_A$ such that there exists $\tau : [t] \times [0,1] \to \Delta_A$ satisfying $c(x) = \tau(g(x), p(x))$ for every $x \in X$.

We now state our omniprediction theorem for convex and special constraints and objectives. In the theorems below, we use $\mathcal{D}$ to denote an underlying distribution over $X \times \{0,1\}$ and use $\mathcal{C}$ to denote a class of functions $c : X \to A$.

Theorem 4.4. *Let $A = [0,1]$ be an action set and let $g : X \to [t]$ be a group partition. Let $\mathcal{T}$ be a class of tasks that only have group constraints and group objectives that are all convex and special. Let p be a predictor in $\mathsf{GrpMA}_{\mathcal{D}}(\mathcal{C}, g, \varepsilon/6) \cap \mathsf{GrpCal}_{\mathcal{D}}(g, \varepsilon/6)$ and define $\mathcal{C}_{p,g}$ as in Definition 4.3. Then p is a $(\mathcal{T}, \mathcal{C}, \mathcal{C}_{p,g}, \varepsilon)$-omnipredictor on $\mathcal{D}$.*

We remark that the convexity assumption in the theorem above can be removed if we replace $\mathcal{C}_{p,g}$ with $\mathcal{C}_{p,g}^{\mathsf{rand}}$ (Theorem D.9), in which case we can handle any finite action

[2] Here for simplicity we assume that we know r_i, r_i^+, r_i^-. These quantities can be estimated from unlabeled data and a predictor satisfying group calibration. It is also natural to compute and store estimates for these quantities when we train an omnipredictor before seeing downstream tasks because these estimates are helpful for general tasks, not just for those with group fairness constraints (see Appendix H).

set $A \subseteq [0,1]$. Once we construct an omnipredictor using Theorem 4.4 (and other theorems in this section), we can efficiently transform it into nearly optimal actions for any task $T \in \mathcal{T}$ (see Appendix H). Theorem 4.4 generalizes the results in (Gopalan et al., 2023) that hold in the single-group unconstrained setting. Our following theorem deals with general convex and Lipschitz group objectives and constraints and it generalizes the results in (Gopalan et al., 2022).

Definition 4.5. We say an objective/constraint function $f : X \times A \times \{0,1\} \to \mathbb{R}$ is κ-Lipschitz if $f(x, \cdot, y)$ is κ-Lipschitz for every fixed $(x, y) \in X \times \{0,1\}$. We say f has B-bounded difference if $\partial f(x, a) \in [-B, B]$ for every $(x, a) \in X \times A$.

Theorem 4.6. *Let $A = [0,1]$ be an action set and let $g : X \to [t]$ be a group partition. Let $\mathcal{T}$ be a class of tasks that only have group objectives and group constraints that are all convex and 1-Lipschitz and have 1-bounded differences. Let p be a predictor in $\mathsf{GrpMC}_{\mathcal{D}}(\mathcal{C}, g, \varepsilon/15) \cap \mathsf{GrpCal}_{\mathcal{D}}(g, \varepsilon/15)$ and define $\mathcal{C}_{p,g}$ as in Definition 4.3. Then p is a $(\mathcal{T}, \mathcal{C}, \mathcal{C}_{p,g}, \varepsilon)$-omnipredictor on $\mathcal{D}$.*

Finally, we consider general group constraints. These constraints allow us to constrain the entire distribution of $c(x)$ (e.g. constraints on $\Pr[c(x) \in A']$ for $A' \subseteq A$) and the distribution of $c(x)$ within each group (e.g. constraints on $\Pr[c(x) \in A', g(x) = i]$).

Theorem 4.7. *Let A be a finite non-empty action set and let $g : X \to [t]$ be a group partition. Let $\mathcal{T}$ be a class of tasks with group constraints and group objectives that all have 1-bounded differences. Let p be a predictor in $\mathsf{GrpLMA}_{\mathcal{D}}(\mathcal{C}, g, \varepsilon/3) \cap \mathsf{GrpCal}_{\mathcal{D}}(g, \varepsilon/3)$ and define $\mathcal{C}_{p,g}^{\mathsf{rand}}$ as in Definition 4.3. Then p is a $(\mathcal{T}, \mathcal{C}, \mathcal{C}_{p,g}^{\mathsf{rand}}, \varepsilon)$-omnipredictor on $\mathcal{D}$.*

We give counterexamples in Appendix I showing that strengthening standard multiaccuracy and multicalibration to their group-wise and/or level-set variants in the theorems above is necessary.

5. Interaction between Group Fairness and Loss Minimization

In this section we explain how we can use our omnipredictors to get an additional property, which we call rank-preserving. The intuition is that if we assume the predictor $p : X \to [0,1]$ describes an approximation to the true probability $\Pr_{(x,y)\sim\mathcal{D}}[y = 1]$, then we want individuals x with higher $p(x)$ to get higher action values, for real-valued actions $a \in \mathbb{R}$. This requirement can be thought of as a fairness property, that individuals that are more likely to succeed (within the same group) should get higher actions.

Definition 5.1. Let $A \subseteq \mathbb{R}$ be set of real-valued actions. We say a transformation $\tau : [t] \times [0,1] \to A$ is *rank-preserving* if for all $i \in [t]$ and $v > v' \in [0,1]$ we have $\tau(i, v) \geq \tau(i, v')$. Let $p : X \to [0,1]$ be a predictor and $g : X \to [t]$ be a group index function. We denote by $\mathsf{rp}\text{-}\mathcal{C}_{p,g}$ the set of all functions $c \in \mathcal{C}_{p,g}$ such that there exists a rank-preserving transformation $\tau : [t] \times [0,1] \to A$ satisfying $c(x) = \tau(g(x), p(x))$ for every $x \in X$.

Our goal is to show that for a large class of optimization problems $\mathcal{T}$, the post-processing of the omnipredictor can output an optimal solution that is also rank-preserving. We achieve this by showing that for every problem $T \in \mathcal{T}$,

$$\mathsf{opt}_{\mathcal{D}_p}(T, \mathsf{rp}\text{-}\mathcal{C}_{p,g}, \varepsilon) \leq \mathsf{opt}_{\mathcal{D}_p}(T, \mathcal{C}_{p,g}, \varepsilon). \tag{5}$$

Inequality (5) implies that when we solve a downstream task using an omnipredictor, i.e., when we compute a solution in $\mathcal{C}_{p,g} \cap \mathsf{sol}_{\mathcal{D}_p}(T, \beta' + \varepsilon/3, 2\varepsilon/3)$ as in Definition 2.1, we can search only within the class $\mathsf{rp}\text{-}\mathcal{C}_{p,g}$ instead of the entire class $\mathcal{C}_{p,g}$. This would ensure that our final solution is rank-preserving. Searching over $\mathsf{rp}\text{-}\mathcal{C}_{p,g}$ can be implemented efficiently by adding linear constraints (for the rank-preserving requirement) to the linear/convex programming in Appendix H.

In Appendix F we prove Lemma F.4 showing that (5) holds for a class of optimization problems when the loss functions and constraints satisfy certain monotonicity conditions, and there is a single linear constraint per group $i \in [t]$. In Lemma F.6 we prove a randomized version of (5), when the constraints are independent of the outcome y. We remark that some monotonicity requirements from the loss functions and the constraints are necessary to promise a rank-preserving optimal solution.

Acknowledgments

We thank Cynthia Dwork and Guy Rothblum for discussions that motivated this project and Parikshit Gopalan and Michael Kim for useful discussions. We thank the authors of Globus-Harris et al. (2022) for discussing their work with us.

LH is supported by the Simons Foundation Collaboration on the Theory of Algorithmic Fairness, Omer Reingold's NSF Award IIS-1908774, and Moses Charikar's Simons Investigators award. IL is supported by the Simons Foundation Collaboration on the Theory of Algorithmic Fairness, the Sloan Foundation Grant 2020-13941, and the Zuckerman STEM Leadership Program. OR is supported by the Simons Foundation Collaboration on the Theory of Algorithmic Fairness and the Simons Foundation Investigators award 689988. CY is supported by the Simons Foundation Collaboration on the Theory of Algorithmic Fairness and Omer Reingold's NSF Award IIS-1908774.

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. PMLR, 2018.

Barda, N., Riesel, D., Akriv, A., Levy, J., Finkel, U., Yona, G., Greenfeld, D., Sheiba, S., Somer, J., Bachmat, E., Rothblum, G. N., Shalit, U., Netzer, D., Balicer, R., and Dagan, N. Developing a COVID-19 mortality risk prediction model when individual-level data are not available. *Nature Communications*, 11(1):4439, 2020. doi: 10.1038/s41467-020-18297-9. URL https://doi.org/10.1038/s41467-020-18297-9.

Bechavod, Y. and Roth, A. Individually fair learning with one-sided feedback. *arXiv preprint arXiv:2206.04475*, 2022.

Canonne, C. L. A short note on learning discrete distributions. *arXiv preprint arXiv:2002.11457*, 2020.

Celis, L. E., Huang, L., Keswani, V., and Vishnoi, N. K. Classification with fairness constraints: A meta-algorithm with provable guarantees. In *Proceedings of the conference on fairness, accountability, and transparency*, pp. 319–328, 2019.

Cotter, A., Jiang, H., Gupta, M. R., Wang, S., Narayan, T., You, S., and Sridharan, K. Optimization with non-differentiable constraints with applications to fairness, recall, churn, and other goals. *J. Mach. Learn. Res.*, 20 (172):1–59, 2019.

Donini, M., Oneto, L., Ben-David, S., Shawe-Taylor, J. S., and Pontil, M. Empirical risk minimization under fairness constraints. *Advances in Neural Information Processing Systems*, 31, 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. ACM, 2012.

Dwork, C., Kim, M. P., Reingold, O., Rothblum, G. N., and Yona, G. Learning from outcomes: Evidence-based rankings. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 106–125. IEEE, 2019.

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.

Feldman, V. Distribution-specific agnostic boosting. In Yao, A. C. (ed.), *Innovations in Computer Science - ICS 2010, Tsinghua University, Beijing, China, January 5-7, 2010. Proceedings*, pp. 241–250. Tsinghua University Press, 2010. URL http://conference.iiis.tsinghua.edu.cn/ICS2010/content/papers/20.html.

Globus-Harris, I., Gupta, V., Jung, C., Kearns, M., Morgenstern, J., and Roth, A. Multicalibrated regression for downstream fairness. *arXiv preprint arXiv:2209.07312*, 2022.

Goh, G., Cotter, A., Gupta, M., and Friedlander, M. P. Satisfying real-world goals with dataset constraints. *Advances in Neural Information Processing Systems*, 29, 2016.

Gopalan, P., Kalai, A. T., Reingold, O., Sharan, V., and Wieder, U. Omnipredictors. In Braverman, M. (ed.), *13th Innovations in Theoretical Computer Science Conference, ITCS 2022, January 31 - February 3, 2022, Berkeley, CA, USA*, volume 215 of *LIPIcs*, pp. 79:1–79:21. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. doi: 10.4230/LIPIcs.ITCS.2022.79. URL https://doi.org/10.4230/LIPIcs.ITCS.2022.79.

Gopalan, P., Hu, L., Kim, M. P., Reingold, O., and Wieder, U. Loss Minimization Through the Lens Of Outcome Indistinguishability. In Tauman Kalai, Y. (ed.), *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*, volume 251 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 60:1–60:20, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-263-1. doi: 10.4230/LIPIcs.ITCS.2023.60. URL https://drops.dagstuhl.de/opus/volltexte/2023/17563.

Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29, 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. PMLR, 2018.

Hu, L. and Peale, C. Comparative Learning: A Sample Complexity Theory for Two Hypothesis Classes. In Tauman Kalai, Y. (ed.), *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*, volume 251 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 72:1–72:30, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-263-1. doi: 10.4230/LIPIcs.ITCS.2023.72. URL https://drops.dagstuhl.de/opus/volltexte/2023/17575.

Hu, L., Peale, C., and Reingold, O. Metric entropy duality and the sample complexity of outcome indistinguishability. In Dasgupta, S. and Haghtalab, N. (eds.), *Proceedings of The 33rd International Conference on Algorithmic Learning Theory*, volume 167 of *Proceedings of Machine Learning Research*, pp. 515–552. PMLR, 29 Mar–01 Apr 2022a. URL https://proceedings.mlr.press/v167/hu22a.html.

Hu, S., Wu, Z. S., and Smith, V. Provably fair federated learning via bounded group loss. *arXiv preprint arXiv:2203.10190*, 2022b.

Kalai, A. T., Mansour, Y., and Verbin, E. On agnostic boosting and parity learning. In *STOC'08*, pp. 629–638. ACM, New York, 2008. doi: 10.1145/1374376.1374466. URL https://doi.org/10.1145/1374376.1374466.

Kamishima, T., Akaho, S., Asoh, H., and Sakuma, J. Fairness-aware classifier with prejudice remover regularizer. In *Joint European conference on machine learning and knowledge discovery in databases*, pp. 35–50. Springer, 2012.

Kearns, M. J. and Schapire, R. E. Efficient distribution-free learning of probabilistic concepts (extended abstract). In *31st Annual Symposium on Foundations of Computer Science, St. Louis, Missouri, USA, October 22-24, 1990, Volume I*, pp. 382–391. IEEE Computer Society, 1990. doi: 10.1109/FSCS.1990.89557. URL https://doi.org/10.1109/FSCS.1990.89557.

Kim, M., Reingold, O., and Rothblum, G. Fairness through computationally-bounded awareness. *Advances in Neural Information Processing Systems*, 31, 2018.

Kim, M. P. and Perdomo, J. C. Making Decisions Under Outcome Performativity. In Tauman Kalai, Y. (ed.), *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*, volume 251 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 79:1–79:15, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-263-1. doi: 10.4230/LIPIcs.ITCS.2023.79. URL https://drops.dagstuhl.de/opus/volltexte/2023/17582.

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

Kim, M. P., Kern, C., Goldwasser, S., Kreuter, F., and Reingold, O. Universal adaptability: Target-independent inference that competes with propensity scoring. *Proceedings of the National Academy of Sciences*, 119(4):e2108097119, 2022. doi: 10.1073/pnas.2108097119. URL https://www.pnas.org/doi/abs/10.1073/pnas.2108097119.

Nabi, R., Malinsky, D., and Shpitser, I. Learning optimal fair policies. In *International Conference on Machine Learning*, pp. 4674–4682. PMLR, 2019.

Narasimhan, H. Learning with complex loss functions and constraints. In *International Conference on Artificial Intelligence and Statistics*, pp. 1646–1654. PMLR, 2018.

Rothblum, G. N. and Yona, G. Probably approximately metric-fair learning. Unpublished Manuscript, 2018.

Rothblum, G. N. and Yona, G. Multi-group agnostic PAC learnability. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9107–9115. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/rothblum21a.html.

Sharifi-Malvajerdi, S., Kearns, M., and Roth, A. Average individual fairness: Algorithms, generalization and experiments. *Advances in Neural Information Processing Systems*, 32, 2019.

Zafar, M. B., Valera, I., Rodriguez, M. G., and Gummadi, K. P. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th international conference on world wide web*, pp. 1171–1180, 2017a.

Zafar, M. B., Valera, I., Rogriguez, M. G., and Gummadi, K. P. Fairness constraints: Mechanisms for fair classification. In *Artificial intelligence and statistics*, pp. 962–970. PMLR, 2017b.

Zemel, R., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. Learning fair representations. In *International conference on machine learning*, pp. 325–333. PMLR, 2013.

A. Constraints Require Global Transformations

We give a simple example where the optimal action $c(x)$ for a constrained loss minimization problem depends not only on the single prediction $p(x)$ but also on the predictions $p(x')$ for other individuals $x' \neq x$, assuming that the predictor p agrees with the ground truth: $p(x) = \mathbb{E}[y|x]$. Let $X = \{x_1, x_2\}$ be a set of two individuals, and let $\mathcal{D}$ be a distribution of $(x, y) \in X \times \{0, 1\}$ such that the marginal distribution of x is the uniform distribution over X, and for a predictor $p : X \to [0, 1]$ we have $\mathbb{E}_{\mathcal{D}}[y|x = x_i] = p(x_i)$ for every $i = 1, 2$. Consider the problem of minimizing the expected squared loss $\mathbb{E}_{\mathcal{D}}[(y - c(x))^2]$ under a budget constraint $\mathbb{E}_{\mathcal{D}}[c(x)] \leq 1/2$ over action functions $c : X \to \mathbb{R}$. Defining $a_1 := c(x_1), a_2 := c(x_2), p_1 := p(x_1), p_2 := p(x_2)$, we can write the problem as minimizing

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}[(y - c(x))^2] &= \frac{1}{2}(p_1(1 - a_1)^2 + (1 - p_1)(0 - a_1)^2) + \frac{1}{2}(p_2(1 - a_2)^2 + (1 - p_2)(0 - a_2)^2) \\ &= \frac{1}{2}p_1(1 - p_1) + \frac{1}{2}p_2(1 - p_2) + \frac{1}{2}(a_1 - p_1)^2 + \frac{1}{2}(a_2 - p_2)^2 \end{aligned} \tag{6}$$

under the constraint $a_1 + a_2 \leq 1$ over the variables a_1, a_2. Note that the first two terms in the objective (6) are independent of the actions a_1, a_2, so minimizing (6) is equivalent to minimizing the Euclidean distance from point (a_1, a_2) to point (p_1, p_2). The optimal solution (a_1, a_2) is given by projecting the two dimensional point (p_1, p_2) onto the feasible region (grey area in Figure 1). It is clear that a_1 depends on both p_1 and p_2, and similarly a_2 depends on both p_1 and p_2. It is straightforward to generalize this example to more than two individuals where the optimal action for any individual depends on the predictions for all the other individuals.

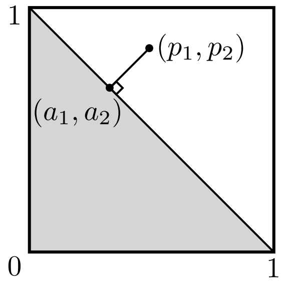

Figure 1. A simple constrained problem whose optimal solution has global dependence.

B. Proof of Equivalence in Multiaccuracy and Multicalibration Definitions

Below we state equivalent definitions of the notions in Section 2.3.

Equivalently to Definition 2.3, $\mathsf{GrpMA}_{\mathcal{D}}(H, g, \varepsilon)$ is the set of predictors $p : X \to [0, 1]$ satisfying the following for every $h \in H$:

$$\sum_{i\in[t]} \left| \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))h(x)\mathbf{1}(g(x) = i)] \right| \leq \varepsilon.$$

Equivalently to Definition 2.4, $\mathsf{GrpMC}_{\mathcal{D}}(H, g, \varepsilon)$ is the set of predictors $p : X \to [0, 1]$ satisfying the following for every $h \in H$:

$$\sum_{i\in[t]} \sum_{v\in\mathsf{range}(p)} \left| \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))h(x)\mathbf{1}(g(x) = i, p(x) = v)] \right| \leq \varepsilon.$$

where the sum is over $i \in [t]$ and $v \in \mathsf{range}(p)$.

Equivalently to Definition 2.5, $\mathsf{GrpCal}_{\mathcal{D}}(g, \varepsilon)$ is the set of predictors $p : X \to [0, 1]$ satisfying:

$$\sum_{i\in[t]} \sum_{v\in\mathsf{range}(p)} \left| \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))\mathbf{1}(g(x) = i, p(x) = v)] \right| \leq \varepsilon.$$

Equivalently to Definition 2.6, $\mathsf{GrpLMA}(H, g, \varepsilon)$ is the set of predictors $p : X \to [0, 1]$ satisfying the following for every $h \in H$:

$$\sum_{i\in[t]} \sum_{a\in A} \left| \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [(y - p(x))\mathbf{1}(g(x) = i, h(x) = a)] \right| \le \varepsilon.$$

We prove the equivalence relationship for GrpMA. Similar proofs can be applied to other definitions.

Claim B.1. In Definition 2.3, a predictor p belongs to $\mathsf{GrpMA}(\mathcal{C}, g, \varepsilon)$ if and only if

$$\sum_{i\in[t]} | \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [(y - p(x))h(x)\mathbf{1}(g(x) = i)]| \le \varepsilon \quad \text{for every } h \in H. \tag{7}$$

Proof. We first show that $p \in \mathsf{GrpMA}(\mathcal{C}, g, \varepsilon)$ implies (7). For a fixed $h \in H$, we choose $\tau : [t] \to [-1, 1]$ such that

$$\tau(i) = \mathsf{sign} \left(\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [(y - p(x))h(x)\mathbf{1}(g(x) = i)] \right), \tag{8}$$

where $\mathsf{sign}(v) = 1$ if $v \ge 0$, and $\mathsf{sign}(v) = -1$ if $v < 0$. By our assumption $p \in \mathsf{GrpMA}(\mathcal{C}, g, \varepsilon)$,

$$\begin{aligned} \varepsilon &\ge \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [(y - p(x))h(x)\tau(g(x))] \\ &= \sum_{i\in[t]} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [(y - p(x))h(x)\mathbf{1}(g(x) = i)\tau(i)] \\ &= \sum_{i\in[t]} | \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [(y - p(x))h(x)\mathbf{1}(g(x) = i)]|. && \text{(by (8))} \end{aligned}$$

This proves (7). Now we prove that (7) implies $p \in \mathsf{GrpMA}(\mathcal{C}, g, \varepsilon)$. For any $h \in H$ and $\tau : [t] \to [-1, 1]$,

$$\begin{aligned} &\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [(y - p(x))h(x)\tau(g(x))] \\ &= \sum_{i\in[t]} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [(y - p(x))h(x)\mathbf{1}(g(x) = i)\tau(i)] \\ &\le \sum_{i\in[t]} | \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [(y - p(x))h(x)\mathbf{1}(g(x) = i)]| && \text{(by } \tau(i) \in [-1, 1]) \\ &\le \varepsilon. && \text{(by (7))} \end{aligned}$$

This proves $p \in \mathsf{GrpMA}(\mathcal{C}, g, \varepsilon)$. □

Remark B.2. The proof above can be adapted to show that if we restrict τ to only output values in $\{-1, 1\}$ instead of $[-1, 1]$, we also get an equivalent definition of GrpMA, and this holds for other definitions in Section 2.3 as well.

C. Proof of Lemma 3.1

We restate and prove Lemma 3.1 below.

Lemma C.1. *Let $\mathcal{D}$ be a distribution over $X \times \{0, 1\}$ and $\varepsilon \ge 0$ be a parameter. Let $\mathcal{T}$ be a collection of constrained loss minimization tasks and let $\mathcal{C}, \mathcal{C}'$ be classes of functions $c : X \to \Delta_A$. If a predictor p satisfies the following two properties for every $T \in \mathcal{T}$, then p is a $(\mathcal{T}, \mathcal{C}, \mathcal{C}', \varepsilon)$-omnipredictor on $\mathcal{D}$:*

1. *Let f_0 be the loss function of T and $(f_j)_{j\in J}$ be the constraints of T. For every $c \in \mathcal{C}$, there exists $c' \in \mathcal{C}'$ such that for every $j \in \{0\} \cup J$,*

$$\begin{aligned} &\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c'(x)} f_j(x, a, y) \\ &\le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_j(x, a, y) + \varepsilon/3. \end{aligned} \tag{2}$$

2. *For every $c \in \mathcal{C}'$ and every $j \in \{0\} \cup J$,*

$$\begin{aligned}&\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_j(x,a,y) \\ \leq &\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c(x)} f_j(x,a,y) + \varepsilon/3.\end{aligned} \tag{3}$$

Proof. Fix an arbitrary task $T \in \mathcal{T}$. Define $\beta := \mathsf{opt}_{\mathcal{D}}(T, \mathcal{C}, 0)$ and $\beta' := \mathsf{opt}_{\mathcal{D}_p}(T, \mathcal{C}', \varepsilon/3)$ as in Definition 2.1. By the definition of β, for any $\beta_1 > \beta$, there exists $c \in \mathcal{C} \cap \mathsf{sol}_{\mathcal{D}}(T, \beta_1, 0)$. By (2), there exists $c' \in \mathcal{C}' \cap \mathsf{sol}_{\mathcal{D}_p}(T, \beta_1 + \varepsilon/3, \varepsilon/3)$. This implies that $\beta' \leq \beta_1 + \varepsilon/3$, and thus $\beta' \leq \beta + \varepsilon/3$. Now we have $\beta' + \varepsilon/3 \leq \beta + 2\varepsilon/3$, and thus

$$\mathcal{C}' \cap \mathsf{sol}_{\mathcal{D}_p}(T, \beta' + \varepsilon/3, 2\varepsilon/3) \subseteq \mathcal{C}' \cap \mathsf{sol}_{\mathcal{D}_p}(T, \beta + 2\varepsilon/3, 2\varepsilon/3). \tag{9}$$

Inequality (3) implies that for any $\beta_2 \in \mathbb{R}$ and $\varepsilon' \in \mathbb{R}_{\geq 0}$, $\mathcal{C}' \cap \mathsf{sol}_{\mathcal{D}_p}(T, \beta_2, \varepsilon') \subseteq \mathsf{sol}_{\mathcal{D}}(T, \beta_2 + \varepsilon/3, \varepsilon' + \varepsilon/3)$, and thus

$$\mathcal{C}' \cap \mathsf{sol}_{\mathcal{D}_p}(T, \beta + 2\varepsilon/3, 2\varepsilon/3) \subseteq \mathsf{sol}_{\mathcal{D}}(T, \beta + \varepsilon, \varepsilon). \tag{10}$$

Combining (9) and (10) completes the proof. □

D. Proofs for Section 4

D.1. Proof of Theorem 4.4

Theorem 4.4. *Let $A = [0, 1]$ be an action set and let $g : X \to [t]$ be a group partition. Let $\mathcal{T}$ be a class of tasks that only have group constraints and group objectives that are all convex and special. Let p be a predictor in $\mathsf{GrpMA}_{\mathcal{D}}(\mathcal{C}, g, \varepsilon/6) \cap \mathsf{GrpCal}_{\mathcal{D}}(g, \varepsilon/6)$ and define $\mathcal{C}_{p,g}$ as in Definition 4.3. Then p is a $(\mathcal{T}, \mathcal{C}, \mathcal{C}_{p,g}, \varepsilon)$-omnipredictor on $\mathcal{D}$.*

We first prove three helper lemmas/claims below and then prove Theorem 4.4.

Claim D.1. For any predictor $p : X \to [0, 1]$, any function $f : X \times A \times \{0, 1\} \to \mathbb{R}$ and any $c : X \to A$, we have

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c(x), y) - \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c(x), y) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [(y - p(x))\partial f(x, c(x))], \tag{11}$$

where $\partial f(x, a) := f(x, a, 1) - f(x, a, 0)$ for every $(x, a) \in X \times A$.

Proof. The claim is proved by plugging the following equation into the left-hand side of (11).

$$f(x, c(x), y) = f(x, c(x), 0) + y\, \partial f(x, c(x)).$$

We get

$$\begin{aligned}&\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c(x), y) - \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c(x), y) \\ = &\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [f(x, c(x), 0) + y\, \partial f(x, c(x))] - \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} [f(x, c(x), 0) + y\, \partial f(x, c(x))].\end{aligned}$$

The distributions $\mathcal{D}, \mathcal{D}_p$ are identical on the x part, therefore $f(x, c(x), 0)$ cancels out. The distribution $\mathcal{D}_p$ is defined such that $y = 1$ with probability $p(x)$, which finishes the proof. □

Lemma D.2. *In the setting of Theorem 4.4, for every $c \in \mathcal{C}$, there exists $c' \in \mathcal{C}_{p,g}$ such that for every convex and special group objective/constraint $f : X \times A \times \{0, 1\} \to \mathbb{R}$, it holds that*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c'(x), y) \leq \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c(x), y) + \varepsilon/3.$$

Proof. By Claim D.1,

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c(x), y) - \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c(x), y) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [(y - p(x))\partial f(x, c(x))]. \tag{12}$$

Since f is a special objective/constraint, there exist $\tau_1, \tau_2 : [t] \to [-1,1]$ such that $\partial f(x, c(x)) = \tau_1(g(x)) + \tau_2(g(x))c(x)$. By our assumption that $p \in \mathsf{GrpCal}(g, \varepsilon/6)$, we have

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))\tau_1(g(x))] \geq -\varepsilon/6.$$

By our assumption that $p \in \mathsf{GrpMA}(\mathcal{C}, g, \varepsilon/6)$, we have

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))\tau_2(g(x))c(x)] \geq -\varepsilon/6.$$

Combining them, we have

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))\partial f(x, c(x))] = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))(\tau_1(g(x)) + \tau_2(g(x))c(x))] \geq -\varepsilon/3. \tag{13}$$

Finally, define τ such that $\tau(i, v) = \mathbb{E}[c(x)|g(x) = i, p(x) = v]$ and define $c'(x) = \tau(g(x), p(x))$. It is clear that $c' \in \mathcal{C}_{p,g}$. Moreover, by the convexity of f, we have

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c'(x), y) \leq \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c(x), y).$$

Combining this with (12) and (13) completes the proof. □

Lemma D.3. *In the setting of Theorem 4.4, for every $c \in \mathcal{C}_{p,g}$, for every convex and special group objective/constraint $f : X \times A \times \{0,1\} \to \mathbb{R}$, it holds that*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c(x), y) \leq \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c(x), y) + \varepsilon/3.$$

Proof. By Claim D.1,

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c(x), y) - \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c(x), y) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))\partial f(x, c(x))]. \tag{14}$$

Since f is convex and special, there exists $\tau : [t] \times A \to [-2, 2]$ such that $\partial f(x, a) = \tau(g(x), a)$. Since $c \in \mathcal{C}_{p,g}$, there exists $\tau' : X \times [0,1] \to A$ such that $c(x) = \tau(g(x), p(x))$. Therefore,

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))\partial f(x, c(x))] = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))\tau(g(x), \tau'(g(x), p(x)))] \leq \varepsilon/3, \tag{15}$$

where the last inequality holds by our assumption that $p \in \mathsf{GrpCal}(g, \varepsilon/6)$. Combining (14) and (15) completes the proof. □

Proof of Theorem 4.4. The proof is completed by applying Lemma 3.1 to the setting of Theorem 4.4 and observing that (2) and (3) in Lemma 3.1 can be established by Lemma D.2 and Lemma D.3, respectively. □

D.2. Proof of Theorem 4.6

Theorem 4.6. *Let $A = [0,1]$ be an action set and let $g : X \to [t]$ be a group partition. Let $\mathcal{T}$ be a class of tasks that only have group objectives and group constraints that are all convex and 1-Lipschitz and have 1-bounded differences. Let p be a predictor in $\mathsf{GrpMC}_{\mathcal{D}}(\mathcal{C}, g, \varepsilon/15) \cap \mathsf{GrpCal}_{\mathcal{D}}(g, \varepsilon/15)$ and define $\mathcal{C}_{p,g}$ as in Definition 4.3. Then p is a $(\mathcal{T}, \mathcal{C}, \mathcal{C}_{p,g}, \varepsilon)$-omnipredictor on $\mathcal{D}$.*

We first prove three helper lemmas below and then prove Theorem 4.6.

Lemma D.4 ((Gopalan et al., 2022))**.** *Let $c : X \to \mathbb{R}$ be a function. Let $g : X \to [t]$ be a group partition function. Let $f : X \times \mathbb{R} \times \{0,1\} \to \mathbb{R}$ be a convex 1-Lipschitz group objective/constraint (Definitions 4.1, 4.2 and 4.5). Define $\tau, \tau' : [t] \to \mathbb{R}$ such that $\tau(i) = \mathbb{E}[y|g(x) = i]$ and $\tau'(i) = \mathbb{E}[c(x)|g(x) = i]$ for every $i \in [t]$. Assume that $\sum_{i\in[t]} |\mathbb{E}_{(x,y)\sim\mathcal{D}}[(y - \tau(i))c(x)\mathbf{1}(g(x) = i)]| \leq \varepsilon$. We have*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[f(x, \tau'(g(x)), y)] \leq \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[f(x, c(x), y)] + 2\varepsilon.$$

Lemma D.4 is essentially Theorem 19 in (Gopalan et al., 2022). The only difference is that in (Gopalan et al., 2022), the function f is not allowed to depend on x, whereas in Lemma D.4, we allow f to depend on the group index $g(x)$ of x. The proof in (Gopalan et al., 2022) can be used here without any essential change.

Lemma D.5. *In the setting of Theorem 4.6, for every $c \in \mathcal{C}$, there exists $c' \in \mathcal{C}_{p,g}$ such that for every convex 1-Lipschitz group objective/constraint $f : X \times A \times \{0,1\} \to \mathbb{R}$ with 1-bounded difference, it holds that*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f_0(x, c'(x), y) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f_0(x, c(x), y) + \varepsilon/3. \tag{16}$$

Proof. We fix an arbitrary $c \in \mathcal{C}$ and define $\tau, \tau' : [t] \times [0,1] \to [0,1]$ such that $\tau(i,v) = \mathbb{E}[y|g(x) = i, p(x) = v]$ and $\tau'(i,v) = \mathbb{E}[c(x)|g(x) = i, p(x) = v]$ for every $(i,v) \in [t] \times [0,1]$.

By our assumption that $p \in \mathsf{GrpCal}(g, \varepsilon/15)$,

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} |p(x) - \tau(g(x), p(x))| \le \varepsilon/15.$$

By our assumption that $p \in \mathsf{GrpMC}(\mathcal{C}, g, \varepsilon/15)$,

$$\sum_{i\in[t]} \sum_{v\in\mathsf{range}(p)} |\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))c(x)\mathbf{1}(g(x) = i, p(x) = v)]| \le \varepsilon/15.$$

Combining the inequalities above,

$$\sum_{i\in[t]} \sum_{v\in\mathsf{range}(p)} |\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - \tau(g(x), p(x)))c(x)\mathbf{1}(g(x) = i, p(x) = v)]| \le 2\varepsilon/15.$$

Define $c' : X \to A$ such that $c'(x) = \tau'(g(x), p(x))$ for every $x \in X$. Clearly, $c' \in \mathcal{C}_{p,g}$. Taking the groups in Lemma D.4 to be $\{x \in X : g(x) = i, p(x) = v\}$ here for $(i,v) \in [t] \times \mathsf{range}(p)$, we have

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c'(x), y) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c(x), y) + 4\varepsilon/15. \tag{17}$$

By Claim D.1,

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c'(x), y) - \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c'(x), y) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))\partial f(x, c'(x))]. \tag{18}$$

Since we assume that f is a group objective/constraint and it has 1-bounded difference, there exists $\tau'' : [t] \times A \times \to [-1,1]$ such that $\partial f(x,a) = \tau''(g(x), a)$. By our definition $c'(x) = \tau'(g(x), p(x))$,

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))\partial f(x, c'(x))] = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))\tau''(g(x), \tau'(g(x), p(x)))].$$

By our assumption that $p \in \mathsf{GrpCal}(g, \varepsilon/15)$,

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - p(x))\tau''(g(x), \tau'(g(x), p(x)))] \ge -\varepsilon/15. \tag{19}$$

Combining (17), (18), and (19) proves (16). □

Lemma D.6. *In the setting of Theorem 4.6, for every $c \in \mathcal{C}_{p,g}$, for every convex 1-Lipschitz group objective/constraint $f : X \times A \times \{0,1\} \to \mathbb{R}$ with 1-bounded difference, it holds that*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c(x), y) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c(x), y) + \varepsilon/3.$$

Proof. The proof is similar to the proof of Lemma D.3 and we omit the details. In the proof of Lemma D.3, we use the assumption that $p \in \mathsf{GrpCal}(g, \varepsilon/6)$ and the fact that there exists $\tau : [t] \times A \to [-2,2]$ such that $\partial f(x,a) = \tau(g(x), a)$. For our f with 1-bounded difference, we can similarly take $\tau : [t] \times A \to [-1,1]$ and use our assumption that $p \in \mathsf{GrpCal}(g, \varepsilon/15) \subseteq \mathsf{GrpCal}(g, \varepsilon/3)$. □

Proof of Theorem 4.6. The proof is completed by applying Lemma 3.1 to the setting of Theorem 4.6 and observing that (2) and (3) in Lemma 3.1 can be established by Lemma D.5 and Lemma D.6, respectively. □

D.3. Proof of Theorem 4.7

Theorem 4.7. *Let A be a finite non-empty action set and let $g : X \to [t]$ be a group partition. Let $\mathcal{T}$ be a class of tasks with group constraints and group objectives that all have 1-bounded differences. Let p be a predictor in $\mathsf{GrpLMA}_{\mathcal{D}}(\mathcal{C}, g, \varepsilon/3) \cap \mathsf{GrpCal}_{\mathcal{D}}(g, \varepsilon/3)$ and define $\mathcal{C}^{\mathsf{rand}}_{p,g}$ as in Definition 4.3. Then p is a $(\mathcal{T}, \mathcal{C}, \mathcal{C}^{\mathsf{rand}}_{p,g}, \varepsilon)$-omnipredictor on $\mathcal{D}$.*

We first prove two helper lemmas below and then prove Theorem 4.7.

Lemma D.7. *In the setting of Theorem 4.7, for every $c \in \mathcal{C}$, there exists $c' \in \mathcal{C}^{\mathsf{rand}}_{p,g}$ such that for every group objective/constraint $f : X \times A \times \{0, 1\} \to \mathbb{R}$ with 1-bounded difference, it holds that*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c'(x)} f(x, a, y) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c(x), y) + \varepsilon/3.$$

Proof. By Claim D.1,

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c(x), y) - \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c(x), y) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} [(y - p(x))\partial f(x, c(x))]. \tag{20}$$

Since we assume that f is a group objective/constraint and it has 1-bounded difference, there exists $\tau : [t] \times A \to [-1, 1]$ such that $\partial f(x, a) = \tau(g(x), a)$. By our assumption that $p \in \mathsf{GrpLMA}(\mathcal{C}, g, \varepsilon/3)$,

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} [(y - p(x))\partial f(x, c(x))] = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} [(y - p(x))\tau(g(x), c(x)))] \ge -\varepsilon/3. \tag{21}$$

Combining (20) and (21), we have

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c(x), y) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, c(x), y) + \varepsilon/3. \tag{22}$$

Now we define $\tau' : [t] \times [0, 1] \to \Delta_A$ such that $\tau'(i, v)$ is the conditional distribution of $c(x)$ given $g(x) = i$ and $p(x) = v$. We define $c' : X \to \Delta_A$ such that $c'(x) = \tau'(g(x), c(x))$. Clearly, $c' \in \mathcal{C}^{\mathsf{rand}}_{p,g}$. Since f is a group objective/constraint, there exists $\tau'' : [t] \times A \times \{-1, 1\} \to \mathbb{R}$ such that $f(x, a, y) = \tau''(g(x), a, y)$. Now we have

$$\begin{aligned}
\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, c(x), y) &= \mathbb{E}[\mathbb{E}[f(x, c(x), y)|g(x), p(x)]] \\
&= \mathbb{E}[\mathbb{E}[\tau''(g(x), c(x), y)|g(x), p(x)]] \\
&= \mathop{\mathbb{E}}_{x}\left[\mathop{\mathbb{E}}_{a\sim\tau'(g(x),p(x)),y\sim\mathsf{Ber}(p(x))} [\tau''(g(x), a, y)]\right] \\
&= \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c'(x)} [f(x, a, y)].
\end{aligned} \tag{23}$$

Combining (22) and (23) completes the proof. □

Lemma D.8. *In the setting of Theorem 4.7, for every $c \in \mathcal{C}^{\mathsf{rand}}_{p,g}$, for every group objective/constraint $f : X \times A \times \{0, 1\} \to \mathbb{R}$ with 1-bounded difference, it holds that*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f(x, a, y) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c(x)} f(x, a, y) + \varepsilon/3. \tag{24}$$

Proof. By our assumption $c \in \mathcal{C}^{\mathsf{rand}}_{p,g}$, there exists $\tau : [t] \times [0, 1] \to \Delta_A$ such that $c(x) = \tau(g(x), p(x))$ for every $x \in X$. Consider the joint distribution of (x, a, y) where $(x, y) \sim \mathcal{D}$ and $a \sim c(x)$. This distribution can be equivalently defined as follows. We first construct a function $\tau' : [t] \times [0, 1] \to A$ at random, where $\tau'(i, v) \in A$ is drawn independently from the distribution $\tau(i, v) \in \Delta_A$ for every $(i, v) \in [t] \times [0, 1]$. We then draw $(x, y) \sim \mathcal{D}$ and choose $c(x) = \tau'(g(x), p(x))$. This equivalent construction also works when we replace $\mathcal{D}$ with $\mathcal{D}_p$. Therefore, to prove (24), it suffices to prove that for every $\tau' : [t] \times [0, 1] \to A$,

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x, \tau'(g(x), p(x)), y) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x, \tau'(g(x), p(x)), y) + \varepsilon/3. \tag{25}$$

By Claim D.1,

$$\begin{aligned}&\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x,\tau'(g(x),p(x)),y) - \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x,\tau'(g(x),p(x)),y)\\ ={}& \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y-p(x))\partial f(x,\tau'(g(x),p(x)))]. \end{aligned}\tag{26}$$

Since we assume that f is a group objective/constraint and it has 1-bounded difference, there exists $\tau'' : [t]\times A \to [-1,1]$ such that $\partial f(x,a) = \tau''(g(x),a)$. Therefore,

$$\begin{aligned}&\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y-p(x))\partial f(x,\tau'(g(x),p(x)))]\\ ={}& \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y-p(x))\tau''(g(x),\tau'(g(x),p(x)))]\\ \le{}& \varepsilon/3,\end{aligned}\tag{27}$$

where the last inequality follows from our assumption $p \in \mathsf{GrpCal}(g,\varepsilon/3)$. Combining (26) and (27) proves (25). □

Proof of Theorem 4.7. The proof is completed by applying Lemma 3.1 to the setting of Theorem 4.7 and observing that (2) and (3) in Lemma 3.1 can be established by Lemma D.7 and Lemma D.8, respectively. □

D.4. Variant of Theorem 4.4

Theorem D.9. *Let $\mathcal{D}$ be a distribution over $X\times\{0,1\}$. Let $A \subseteq [0,1]$ be a finite action set. Let $\mathcal{T}$ be a class of tasks that only have group constraints and group objectives that are all special. Let $\mathcal{C}$ be a class of functions $c : X \to A$. Let p be a predictor in $\mathsf{GrpMA}_{\mathcal{D}}(\mathcal{C},g,\varepsilon/6)\cap \mathsf{GrpCal}_{\mathcal{D}}(g,\varepsilon/6)$ and define $\mathcal{C}^{\mathsf{rand}}_{p,g}$ as in Definition 4.3. Then p is a $(\mathcal{T},\mathcal{C},\mathcal{C}^{\mathsf{rand}}_{p,g},\varepsilon)$-omnipredictor on $\mathcal{D}$.*

We first prove two helper lemmas below and then prove Theorem D.9.

Lemma D.10. *In the setting of Theorem D.9, for every $c\in\mathcal{C}$, there exists $c'\in\mathcal{C}^{\mathsf{rand}}_{p,g}$ such that for every special group objective/constraint $f : X\times A\times\{0,1\}\to\mathbb{R}$, it holds that*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p}\ \mathop{\mathbb{E}}_{a\sim c'(x)} f(x,a,y) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x,c(x),y)+\varepsilon/3.$$

Proof. Using the same argument as in the proof of Lemma D.2, we can show that

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f(x,c(x),y) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f(x,c(x),y)+\varepsilon/3.$$

This is the same as (22) as in the proof of Lemma D.7, and the rest of the proof follows the same argument as in the proof of Lemma D.7. □

Lemma D.11. *In the setting of Theorem D.9, for every $c\in\mathcal{C}^{\mathsf{rand}}_{p,g}$, for every special group objective/constraint $f : X\times A\times\{0,1\}\to\mathbb{R}$, it holds that*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\ \mathop{\mathbb{E}}_{a\sim c(x)} f(x,a,y) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p}\ \mathop{\mathbb{E}}_{a\sim c(x)} f(x,a,y)+\varepsilon/3.$$

Proof. The proof follows from the same argument as the proof of Lemma D.8. In Lemma D.8, we use the assumption that $p\in\mathsf{GrpCal}(g,\varepsilon/3)$ and that f has 1-bounded difference. Here we have the assumption that $p\in\mathsf{GrpCal}(g,\varepsilon/6)$, and since we assume f is special and $A\subseteq[0,1]$, we know that f has 2-bounded difference. □

Proof of Theorem D.9. The proof is completed by applying Lemma 3.1 to the setting of Theorem D.9 and observing that (2) and (3) in Lemma 3.1 can be established by Lemma D.10 and Lemma D.11, respectively. □

E. Lipschitz Combination of Constraints

We show that all our omniprediction results in Section 4 can be extended to more general constrained loss minimization tasks where we combine the constraints using a Lipschitz function. Specifically, we consider more general tasks where each task T not only has an objective $f_0 : X \times A \times \{0,1\} \to \mathbb{R}$ and constraints $f_j : X \times A \times \{0,1\}$ for $j \in [m]$, but also has a combining function $\Gamma : \mathbb{R}^m \to \mathbb{R}$. The task T corresponds to the following optimization problem:

$$\begin{aligned} \underset{c:X\to A}{\text{minimize}} \quad & \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f_0(x, c(x), y) \\ \text{s.t.} \quad & \Gamma\left(\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f_1(x, c(x), y), \dots, \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f_m(x, c(x), y) \right) \le 0. \end{aligned} \tag{28}$$

The task in (1) can be viewed as a special case of (28) where Γ is the max function: $\Gamma(r_1, \dots, r_m) = \max(r_1, \dots, r_m)$. For a task T in the form of (28), for $\beta \in \mathbb{R}$ and $\varepsilon \in \mathbb{R}_{\ge 0}$, we can again define $\mathsf{sol}_{\mathcal{D}}(T, \beta, \varepsilon)$ to be the set of randomized action functions $c : X \to \Delta_A$ satisfying

$$\begin{aligned} & \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_0(x, a, y) \le \beta, \quad \text{and} \\ & \Gamma\left(\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_1(x, a, y), \dots, \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_m(x, a, y) \right) \le \varepsilon. \end{aligned}$$

Correspondingly, for a class $\mathcal{C}$ consisting of functions $c : X \to \Delta_A$, we define

$$\mathsf{opt}_{\mathcal{D}}(T, \mathcal{C}, \varepsilon) := \inf\{\beta \in \mathbb{R} : \mathcal{C} \cap \mathsf{sol}_{\mathcal{D}}(T, \beta, \varepsilon) \ne \emptyset\}.$$

We can then similarly define omnipredictors for these tasks in the same way as in Definition 2.1.

Here we focus on obtaining omnipredictors for tasks with Lipschitz combining functions Γ. We say Γ is κ-Lipschitz (in the ℓ_∞ norm) if $|\Gamma(r_1, \dots, r_m) - \Gamma(r_1', \dots, r_m')| \le \kappa \max_{i\in[m]} |r_i - r_i'|$. For tasks with 1-Lipschitz combining functions, we have the following analogue of Lemma 3.1:

Lemma E.1. *Let $\mathcal{T}$ be a class of constrained loss minimization tasks each having a 1-Lipschitz combining function. Let $\mathcal{C}$ and $\mathcal{C}'$ be classes of action functions $f : X \to \Delta_A$ as in Definition 2.1. If a predictor p satisfies the following two properties for every $T \in \mathcal{T}$, then p is a $(\mathcal{T}, \mathcal{C}, \mathcal{C}', \varepsilon)$-omnipredictor:*

1. *Let f_0 be the loss function of T and $(f_j)_{j\in J}$ be the constraints of T. For every $c \in \mathcal{C}$, there exists $c' \in \mathcal{C}'$ such that*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c'(x)} f_0(x, a, y) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_0(x, a, y) + \varepsilon/3, \quad \textit{and} \tag{29}$$

$$\left| \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c'(x)} f_j(x, a, y) - \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_j(x, a, y) \right| \le \varepsilon/3 \quad \textit{for every } j \in J. \tag{30}$$

2. *For every $c \in \mathcal{C}'$,*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_0(x, a, y) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c(x)} f_0(x, a, y) + \varepsilon/3, \quad \textit{and} \tag{31}$$

$$\left| \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_j(x, a, y) - \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c(x)} f_j(x, a, y) \right| \le \varepsilon/3 \quad \textit{for every } j \in J. \tag{32}$$

Lemma E.1 can be proved similarly to Lemma 3.1 using the observation that (30) implies the following by the 1-Lipschitz assumption on Γ and an analogous observation for (32):

$$\begin{aligned} & \Gamma\left(\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c'(x)} f_1(x, a, y), \dots, \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c'(x)} f_m(x, a, y) \right) \\ \le {} & \Gamma\left(\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_1(x, a, y), \dots, \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)} f_m(x, a, y) \right) + \varepsilon/3. \end{aligned}$$

We thus omit the proof of Lemma E.1. The only difference between Lemma E.1 and Lemma 3.1 in the requirements needed for p to be an omnipredictor is the additional absolute values in (30) and (32). As all our proofs in Section 4 are through Lemma 3.1, they can be adapted to tasks with constraints combined by a Lipschitz function using Lemma E.1. The absolute values in (30) and (32) only require us to make sure that for every constraint function f, both f and $-f$ satisfy the assumptions needed for our theorems in Section 4 (e.g. we need to replace "convex" by "affine"). Note that all linear constraints f defined in (4) satisfy that both f and $-f$ are convex and special. Ideas in this section can be applied to tasks where the objective function is also a Lipschitz combination:

$$\begin{aligned} \underset{c:X\to A}{\text{minimize}} \quad & \Gamma'\left(\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f_1'(x,c(x),y),\ldots,\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f_{m'}'(x,c(x),y)\right) \\ \text{s.t.} \quad & \Gamma\left(\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f_1(x,c(x),y),\ldots,\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} f_m(x,c(x),y)\right) \le 0. \end{aligned}$$

F. Rank-preserving Transformations of Omnipredictors

In this section we identify cases where we can ensure that our solutions to downstream tasks given an omnipredictor p is rank-preserving (Definition 5.1). Specifically, we identify conditions under which (5) is satisfied.

Our results in this section focus on actions $a \in [0,1]$ and assume that the objective function is rank-preserving:

Definition F.1. Let $g : X \to [t]$ be a group partition function. We say an objective function $f_0 : X\times[0,1]\times\{0,1\} \to [0,1]$ is rank-preserving (within groups), if there exists a function $f : [t]\times[0,1]\times\{0,1\}$ such that for all $x \in X, a \in [0,1], y \in \{0,1\}$, we have $f_0(x,a,y) = f(g(x),a,y)$ and for every $i \in [t]$ and $a > a' \in [0,1]$,

$$\begin{aligned} f(i,a,1) &\le f(i,a',1) \\ f(i,a,0) &\ge f(i,a',0). \end{aligned}$$

Rank preserving a desired property which holds when the loss function represents the distance between the taken action and the outcome. In particular, the ℓ_1 loss and squared loss satisfy it, as well as every loss function of form $f(x,a,y) = \mathrm{dist}(a,y)$, when dist is a distance function.

We also assume that the predictor p which we use to solve downstream tasks is monotone:

Definition F.2. Let $\mathcal{D}$ be a distribution over $X \times \{0,1\}$ and let $g : X \to [t]$ be a group partition function. We say a predictor $p : X \to [0,1]$ is *monotone* w.r.t. $\mathcal{D}$ and g if for every $i \in [t]$ and for every $v, v' \in [0,1]$ satisfying $v > v'$, we have $\mathbb{E}_{(x,y)\sim\mathcal{D}}[y|p(x) = v, g(x) = i] \ge \mathbb{E}_{(x,y)\sim\mathcal{D}}[y|p(x) = v', g(x) = i]$.

This monotonicity requirement is satisfied if $\mathcal{D} = \mathcal{D}_p$. In Appendix F.1 we describe how to modify p using samples from $\mathcal{D}$ to satisfy this requirement even when $\mathcal{D}$ is different from $\mathcal{D}_p$.

We start with the simpler case, where we assume that the constraint combines functions that are fixed for each group:

Definition F.3. Let $\mathcal{D}$ be a distribution over $X \times \{0,1\}$. Let $g : X \to [t]$ be a group partition function. Let A be an action set. Let $\sigma_1,\ldots,\sigma_t : A \times \{0,1\} \to \mathbb{R}$ be functions. We say a constrained loss minimization task T as in (28) is compatible with $\sigma_1,\ldots,\sigma_t$ if T only has a single constraint of the form

$$\Gamma(\xi_1,\ldots,\xi_t) \le 0 \tag{33}$$

for some function $\Gamma : \mathbb{R}^t \to \mathbb{R}$ where $\xi_i := \mathbb{E}_{(x,y)\sim\mathcal{D}}[\mathbf{1}(g(x) = i)\sigma_i(c(x),y)]$.

In the lemma below we establish (5) when each σ_i is restricted to the form $\sigma_i(a,y) = \alpha_1 y + \alpha_2 a + \alpha_3 ay$ for some $\alpha_1,\alpha_2,\alpha_3 \in \mathbb{R}$ satisfying $\alpha_2(\alpha_2+\alpha_3) \ge 0$. The flexibility in the combining function Γ allows the constraint (33) to express group fairness constraints such as statistical parity and equal opportunity even when each σ_i is restricted in this special form.

Lemma F.4. *Let $\mathcal{D}$ be a distribution over $X \times \{0,1\}$, and $g : X \to [t]$ be a group partition function. Let $A = [0,1]$ be the action set, and let $\sigma_1,\ldots,\sigma_t : A \times \{0,1\} \to \mathbb{R}$ be functions where each σ_i can be written as $\sigma_i(a,y) = \alpha_1 y + \alpha_2 a + \alpha_3 ay$ for some $\alpha_1,\alpha_2,\alpha_3 \in \mathbb{R}$ satisfying $\alpha_2(\alpha_2+\alpha_3) \ge 0$. Let T be a task compatible with $\sigma_1,\ldots,\sigma_k$ and assume its objective function $f_0 : X \times A \times \{0,1\} \to \mathbb{R}$ is rank-preserving and convex. For a predictor $p : X \to [0,1]$, assuming p is monotone w.r.t. $\mathcal{D}$ and g (which is always satisfied when $\mathcal{D} = \mathcal{D}_p$), we have*

$$\mathsf{opt}_{\mathcal{D}}(T, \mathsf{rp}\text{-}\mathcal{C}_{p,g}, \varepsilon) = \mathsf{opt}_{\mathcal{D}}(T, \mathcal{C}_{p,g}, \varepsilon).$$

Furthermore, given any deterministic $c \in \mathcal{C}_{p,g}$, such that $c(x) = \tau(g(x), p(x))$ for a transformation $\tau : [t] \times V \to A$, there exists an algorithm running in time polynomial in $t, |V|, \varepsilon$ and outputting a transformation $\tilde{\tau} : [t] \times V \to A$ that is rank-preserving, and $c'(x) = \tilde{\tau}(g(x), p(x))$ has the same objective value as c up to a factor of ε with high probability.

We remark that the requirement $\alpha_2(\alpha_2 + \alpha_3) \geq 0$ cannot be removed. Without this requirement, it could be the case that some functions in $\mathsf{rp}\text{-}\mathcal{C}_{p,g}$ satisfy the constraint but none of them is rank-preserving. This highlights the importance of picking appropriate loss functions and constraints if we want to achieve fair outcome.

Proof. We prove the claim by an iterative process, taking τ that is not rank-preserving on some inputs and correcting it.

Suppose τ is not rank-preserving, and there exists $i \in [t], v, v' \in V$ such that $(\tau(i,v) - \tau(i,v'))(v - v') < 0$. We show a local correction from τ to τ' such that τ' is rank-preserving on v, v'. The final transformation is created by fixing all such violations. We denote

$$\theta = \Pr_{(x,y)\sim\mathcal{D}}[p(x) = v | g(x) = i, p(x) \in \{v, v'\}] \tag{34}$$

$$q_v = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[y | g(x) = i, p(x) = v] \tag{35}$$

$$q_{v'} = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[y | g(x) = i, p(x) = v'] \tag{36}$$

Constraint value. Let us assume

$$\theta|\alpha_2 + \alpha_3 q_v| \leq (1-\theta)|\alpha_2 + \alpha_3 q_{v'}|. \tag{37}$$

This is without loss of generality because otherwise we can switch the roles of v and v', in which case θ and $1-\theta$ also switch. Our goal is to update the values of $\tau(i,v)$ and $\tau(i,v')$ so that they no longer violate the rank-preserving property while keeping ξ_i defined below unchanged so that the constraint of T is still satisfied:

$$\xi_i := \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\mathbf{1}(g(x) = i)\sigma_i(\tau(g(x), p(x)), y)].$$

By our assumption, we can write σ_i as $\sigma_i(a, y) = \alpha_1 y + \alpha_2 a + \alpha_3 a y$ for $\alpha_1, \alpha_2, \alpha_3 \in \mathbb{R}$ satisfying $\alpha_2(\alpha_2 + \alpha_3) \geq 0$. We only change the values of $\tau(i,v)$ and $\tau(i,v')$, so to keep ξ_i unchanged, it suffices to keep the following conditional expectation unchanged:

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\sigma_i(\tau(g(x), p(x)), y) | g(x) = i, p(x) \in \{v, v'\}] \tag{38}$$

$$= \alpha_1(\theta q_v + (1-\theta)q_{v'}) + \alpha_2(\theta\tau(i,v) + (1-\theta)\tau(i,v')) + \alpha_3(\theta q_v \tau(i,v) + (1-\theta)q_{v'}\tau(i,v')). \tag{39}$$

Simply switching between $\tau(i,v)$ and $\tau(i,v')$ does not work, as the constraint can be violated. Instead, we set $\tau'(i,v) = \tau(i,v')$, and then we want to set $\tau'(i,v')$ to some value keeping the expectation in Equation (38) exactly the same. We denote $\tau'(i,v') = z$ and look for z such that

$$\begin{aligned} &\alpha_1(\theta q_v + (1-\theta)q_{v'}) + \alpha_2(\theta\tau(i,v') + (1-\theta)z) + \alpha_3(\theta q_v \tau(i,v') + (1-\theta)q_{v'} z) \\ &= \alpha_1(\theta q_v + (1-\theta)q_{v'}) + \alpha_2(\theta\tau(i,v) + (1-\theta)\tau(i,v')) + \alpha_3(\theta q_v \tau(i,v) + (1-\theta)q_{v'}\tau(i,v')) \end{aligned}$$

That is,

$$z = \frac{\alpha_2\theta + \alpha_3\theta q_v}{\alpha_2(1-\theta) + \alpha_3(1-\theta)q_{v'}}\tau(i,v) + \frac{\alpha_2(1-2\theta) + \alpha_3((1-\theta)q_{v'} - \theta q_v)}{\alpha_2(1-\theta) + \alpha_3(1-\theta)q_{v'}}\tau(i,v'). \tag{40}$$

We can only set $\tau'(i,v')$ to a value $z \in [0,1]$, so we need to check that the above expression is in this range. We can write $z = \gamma\tau(i,v) + (1-\gamma)\tau(i,v')$ for

$$\gamma = \frac{\alpha_2\theta + \alpha_3\theta q_v}{\alpha_2(1-\theta) + \alpha_3(1-\theta)q_{v'}} = \frac{\theta}{1-\theta} \cdot \frac{\alpha_2 + \alpha_3 q_v}{\alpha_2 + \alpha_3 q_{v'}}. \tag{41}$$

We prove $\gamma \in [0,1]$ as follows, which implies $z \in [0,1]$ (as the range of τ is also $[0,1]$). To prove $\gamma \geq 0$, we know that $\theta \in [0,1]$, so we need to show that the second expression is also positive. From the lemma requirement, we have that $\alpha_2(\alpha_2 + \alpha_3) \geq 0$, together with $q_v, q_v' \in [0,1]$ we get that $(\alpha_2 + q_v\alpha_3)/(\alpha_2 + q_{v'}\alpha_3) \geq 0$ and so $\gamma \geq 0$. The fact $\gamma \leq 1$ follows from (37).

Objective. We are left with showing that the objective of T (i.e., expected value of f_0) is not increased by the correction. For simplicity of notations we denote $\ell_0(v) = f_0(x, \tau(i, v), 0)$ and $\ell_1(v) = f_0(x, \tau(i, v), 1)$ for some x with $g(x) = i$. The values of $\ell_0(v)$ and $\ell_1(v)$ are independent of the actual choice of such x because f_0 is a group constraint by Definition F.1.

Since the objective value is an expectation and therefore additive, it is enough to analyze the value for x such that $g(x) = i, p(x) \in \{v, v'\}$. The original expected objective value for these x's is

$$\begin{aligned}
&\mathop{\mathbb{E}}_{(x,y)\sim D}[f_0(x, \tau(g(x), p(x)), y) | g(x) = i, p(x) \in \{v, v'\}] \\
&= \ell_1(v)\theta q_v + \ell_0(v)\theta(1 - q_v) \\
&\quad + \ell_1(v')(1 - \theta)q_{v'} + \ell_0(v')(1 - \theta)(1 - q_{v'}).
\end{aligned}$$

The same expectation with τ' is given by

$$\begin{aligned}
&\mathop{\mathbb{E}}_{(x,y)\sim D}[f_0(x, \tau'(g(x), p(x)), y) | g(x) = i, p(x) \in \{v, v'\}] \\
&= \ell_1(v')\theta q_v + \ell_0(v')\theta(1 - q_v) \\
&\quad + f_0(i, z, 1)(1 - \theta)q_{v'} + f_0(i, z, 0)(1 - \theta)(1 - q_{v'}) \\
&\le \ell_1(v')\theta q_v + \ell_0(v')\theta(1 - q_v) \\
&\quad + (\gamma\ell_1(v) + (1 - \gamma)\ell_1(v'))(1 - \theta)q_{v'} \\
&\quad + (\gamma\ell_0(v) + (1 - \gamma)\ell_0(v'))(1 - \theta)(1 - q_{v'}).
\end{aligned}$$

The last inequality holds because of the convexity of f_0. Taking the difference, we get

$$\begin{aligned}
&\mathop{\mathbb{E}}_{(x,y)\sim D}[f_0(x, \tau(g(x), p(x), y) - f_0(x, \tau'(g(x), p(x), y) | g(x) = i, p(x) \in \{v, v'\}] \\
&\ge (\ell_1(v) - \ell_1(v'))\theta q_v + (\ell_0(v) - \ell_0(v'))\theta(1 - q_v) \\
&\quad + \gamma(\ell_1(v') - \ell_1(v))(1 - \theta)q_{v'} + \gamma(\ell_0(v') - \ell_0(v))(1 - \theta)(1 - q_{v'}) \\
&= (\ell_1(v) - \ell_1(v'))(\theta q_v - \gamma(1 - \theta)q_{v'}) \\
&\quad + (\ell_0(v') - \ell_0(v))(\gamma(1 - \theta)(1 - q_{v'}) - \theta(1 - q_v)). \qquad (42)
\end{aligned}$$

We show that the expression above is nonnegative. We focus on the case where $v > v'$, and a similar argument applies to the other case $v < v'$. By our assumption that $\tau(i, v), \tau(i, v')$ violate the rank-preserving property, we know that $\tau(i, v') \ge \tau(i, v)$. By our assumption that f_0 is rank-preserving,

$$\begin{aligned}
\ell_1(v) - \ell_1(v') &= f_0(i, \tau(i, v), 1) - f_0(i, \tau(i, v'), 1) \ge 0, \\
\ell_0(v') - \ell_0(v) &= f_0(i, \tau(i, v'), 0) - f_0(i, \tau(i, v), 0) \ge 0. \qquad (43)
\end{aligned}$$

By (41),

$$\gamma = \frac{\theta}{1 - \theta} \cdot \frac{(\alpha_2 + \alpha_3)q_v + \alpha_2(1 - q_v)}{(\alpha_2 + \alpha_3)q_{v'} + \alpha_2(1 - q_{v'})}.$$

By our monotonicity assumption on p, we have $q_v \ge q_{v'}$. Using our assumption that $\alpha_2(\alpha_2 + \alpha_3) \ge 0$ in the equation above, we get

$$\frac{\theta}{1 - \theta}\frac{1 - q_v}{1 - q_{v'}} \le \gamma \le \frac{\theta}{1 - \theta}\frac{q_v}{q_{v'}}.$$

Therefore,

$$\begin{aligned}
\theta q_v - \gamma(1 - \theta)q_{v'} &\ge 0. \\
\gamma(1 - \theta)(1 - q_{v'}) - \theta(1 - q_v) &\ge 0. \qquad (44)
\end{aligned}$$

Plugging (43) and (44) into (42) proves that (42) is nonnegative. Therefore correcting τ does not increase the objective.

After preforming this step for every pair v, v' that violate the rank-preserving property, the resulting transformation $\tilde{\tau}$ is rank-preserving.

The correction described above uses the exact value of $\theta, q_v, q_{v'}$. In order to implement such algorithm in practice, we approximate $\theta, q_v, q_{v'}$, and update τ based on our approximation. Using the approximation instead of the exact values can reduce objective by the approximation error. The running time of the algorithm is polynomial in $|V|, t, \delta$ when δ is the accuracy parameter. □

The previous theorem modified a transformation τ into a rank-preserving one by "correcting" its values for every violation. Allowing the correction to be randomized, the theorem holds for a larger collection of constraints. In order to do so, we first define rank-preserving for a randomized transformation.

Definition F.5. A randomized transformation $\tau : [0,1] \times [t] \to \Delta_A$ for $A = [0,1]$ is rank-preserving within groups, if for every $i \in [t]$, $v > v' \in V$ and $\gamma \in [0,1]$,

$$\Pr[\tau(i,v) \geq \gamma] \geq \Pr[\tau(i,v') \geq \gamma].$$

Let $p : X \to [0,1]$ be a predictor and $g : X \to [t]$ be a group index function. We denote by rp-$\mathcal{C}^{\mathsf{rand}}_{p,g}$ the set of all functions $c \in \mathcal{C}^{\mathsf{rand}}_{p,g}$ such that there exists a rank-preserving transformation $\tau : [t] \times [0,1] \to \Delta_A$ satisfying $c(x) = \tau(g(x), p(x))$ for every $x \in X$.

Lemma F.6. *Let $A \subseteq [0,1]$ be a discrete action set, $g : X \to [t]$ be a group partition function, and T be a task with constraints that are independent of the outcome. For a predictor p and a distribution $\mathcal{D}$, assuming p is monotone w.r.t. $\mathcal{D}$ and g (which is always satisfied when $\mathcal{D} = \mathcal{D}_p$), we have*

$$\mathsf{opt}_{\mathcal{D}}(T, \text{rp-}\mathcal{C}^{\mathsf{rand}}_{p,g}, \varepsilon) = \mathsf{opt}_{\mathcal{D}}(T, \mathcal{C}^{\mathsf{rand}}_{p,g}, \varepsilon).$$

Furthermore, given any $c \in \mathcal{C}^{\mathsf{rand}}_{p,g}$, such that $c(x) = \tau(g(x), p(x))$ for a transformation $\tau : [t] \times V \to A$, there exists an algorithm running in time polynomial in $t, |V|, \varepsilon$ and outputting a randomized transformation $\tilde{\tau} : [t] \times V \to A$ that is rank-preserving, and $c'(x) = \tilde{\tau}(g(x), p(x))$ has the same objective value as c up to a factor of ε with high probability.

Proof. The proof follows the same structure of the previous proof. Let τ be a randomized transformation, and assume that there exists $v > v'$ such that τ is not rank-preserving on v, v'. We describe a single step in an iterative process, transforming τ into τ'.

Intuitively, we take the histogram of the values of τ on the input set $\{x \in X | g(x) = i, p(x) \in \{v, v'\}\}$, and assign v' the lower values in the histogram and v the upper ones.

We define

$$\theta = \Pr_{(x,y) \in \mathcal{D}}[p(x) = v | g(x) = i, p(x) \in \{v, v'\}] \tag{45}$$

$$\theta_a = \theta \Pr[\tau(i,v) = a] + (1-\theta)\Pr[\tau(i,v') = a], \quad \forall a \in A \tag{46}$$

when the probability in the second definition is over the internal randomness of τ. For every $a \in A$, we define the function $u : A \to [0,1]$ indicating how much of θ_a is coming from $\tau(i,v)$. That is, for all $a \in A$ if $\theta \neq 0$ we have

$$u(a) = \theta \Pr[\tau(i,v) = a]/\theta_a.$$

When $\theta_a = 0, u(a)$ can take any value in $[0,1]$. Notice that by definition, $\Pr[\tau(i,v') = a] = \theta_a(1 - u(a))/(1-\theta)$.

We define τ' by creating an analog function $u' : A \to [0,1]$, when u' indicates if a certain outcome $a \in A$ is in the upper part of the histogram (and should be assigned to $\tau'(i,v)$) or lower part (and should be assigned to $\tau'(i,v')$). Fractional values $u'(a)$ imply that a is in the middle of the histogram, i.e. assigned to both. For every $a \in A$ let

$$u'(a) = \begin{cases} 1 & \text{if } \sum_{a' \geq a} \theta_{a'} \leq \theta \\ 0 & \text{if } \sum_{a' \leq a} \theta_{a'} \leq 1 - \theta \\ \frac{1}{\theta_a}\left(\theta - \sum_{a' > a} \theta_{a'}\right) & \text{otherwise.} \end{cases} \tag{47}$$

We are now ready to define τ' to equal τ on all except $(i,v), (i,v')$, in which we have:

$$\forall a \in A \quad \Pr[\tau'(i,v) = a] = \frac{\theta_a u'(a)}{\theta} \tag{48}$$

$$\forall a \in A \quad \Pr[\tau'(i,v) = a] = \frac{\theta_a}{1-\theta}(1 - u'(a)). \tag{49}$$

Notice that τ' is rank-preserving on inputs $(i,v),(i,v')$ by definition.

We next show that τ' satisfies all of the constraints in the same way was τ. Let $f_j(i,a,y) = f(i,a)$ be any constraint that is not a function of y. Then we have

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[f(i,\tau(i,p(x)))] = \sum_{a\in A} \Pr_{(x,y)\sim\mathcal{D}}[\tau(i,p(x)) = a] f(i,a).$$

The transformations τ,τ' only differ on inputs $(i,v),(i,v')$, so it is enough to analyze the difference on these inputs. For every $a \in A$,

$$\Pr_{(x,y)\sim\mathcal{D}}[\tau(i,p(x)) = a | g(x) = i, p(x) \in \{v,v'\}] = \theta \Pr[\tau(i,v) = a] + (1-\theta)\Pr[\tau(i,v') = a] = \theta_a.$$

For the new transformation,

$$\begin{aligned}\Pr_{(x,y)\sim\mathcal{D}}[\tau'(i,p(x)) = a | g(x) = i, p(x) \in \{v,v'\}] &= \theta \Pr[\tau'(i,v) = a] + (1-\theta)\Pr[\tau'(i,v') = a] \\ &= \theta\frac{\theta_a u'(a)}{\theta} + (1-\theta)\frac{\theta_a}{1-\theta}(1-u'(a)) = \theta_a.\end{aligned}$$

Therefore, we get that $\mathbb{E}_{(x,y)\sim\mathcal{D}}[f(i,\tau(i,p(x)))] = \mathbb{E}_{(x,y)\sim\mathcal{D}}[f(i,\tau'(i,p(x)))]$.

We are left with proving that this correction does not increase the loss. We define $q_v, q_{v'}$ as in the previous proof.

$$q_v = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[y | g(x) i, p(x) = v] \tag{50}$$

$$q_{v'} = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[y | g(x) i, p(x) = v']. \tag{51}$$

The expected loss of τ on the relevant inputs:

$$\begin{aligned}&\mathop{\mathbb{E}}_{(x,y)\in\mathcal{D}}[f_0(i,\tau(i,x),y) | g(x) = i, p(x) \in \{v,v'\}] \\ &= \theta \sum_{a\in A} \Pr[\tau(i,v) = a]\left(q_v f_0(i,a,1) + (1-q_v) f_0(i,a,0)\right) \\ &\quad + (1-\theta)\sum_{a\in A} \Pr[\tau(i,v') = a]\left(q_{v'} f_0(i,a,1) + (1-q_{v'}) f_0(i,a,0)\right) \\ &= \sum_{a\in A} f_0(i,a,1)\theta_a\left(u(a)q_v + (1-u(a))q_{v'}\right) \\ &\quad + \sum_{a\in A} f_0(i,a,0)\theta_a\left(u(a)(1-q_v) + (1-u(a))(1-q_{v'})\right).\end{aligned}$$

By definition, the loss of τ' is exactly the same only with u' instead of u.

Comparing the two losses we get:

$$\mathop{\mathbb{E}}_{(x,y)\in\mathcal{D}}[f_0(i,\tau(i,x),y) | g(x) = i, p(x) \in \{v,v'\}] - \mathop{\mathbb{E}}_{(x,y)\in\mathcal{D}}[f_0(i,\tau'(i,x),y) | g(x) = i, p(x) \in \{v,v'\}] \tag{52}$$

$$= \sum_{a\in A} \theta_a (f_0(i,a,1) - f_0(i,a,0))(u(a) - u'(a))(q_v - q_{v'}). \tag{53}$$

Denote $\gamma_a = (u(a) - u'(a))(q_v - q_{v'})$. From our assumption, $q_v \geq q_{v'}$. From the definition of $u'(a)$, for every $a \in A$ we have

$$\sum_{a' \geq a \in A} u'(a) \geq \sum_{a' \geq a \in A} u(a).$$

Since $\sum_{a\in A} u(a) = \sum_{a\in A} u'(a)$, we have that $\sum_a \gamma_a = 0$, and that there exists $\tilde{a}$ such that $\gamma_a \le 0$ for all $a > \tilde{a}$, and $\gamma_a \ge 0$ for all $a \ge \tilde{a}$. Since the function f_0 is rank preserving, we have that for every $a > a'$,

$$f_0(i,a,1) - f_0(i,a,0) \le f_0(i,a',1) - f_0(i,a',0).$$

Therefore,

$$\sum_{a,\gamma_a \le 0} \gamma_a(f_0(i,a,1) - f_0(i,a,0)) \le \sum_{a,\gamma_a \ge 0} \gamma_a(f_0(i,a,1) - f_0(i,a,0)).$$

Which implies that $\sum_a \gamma_a(f_0(i,a,1) - f_0(i,a,0)) \ge 0$ and the loss of τ' is at most the loss of τ.

The final transformation $\tilde{\tau}$ is created by repeatedly applying the above step until $\tilde{\tau}$ is rank-preserving. The process ends after $|V|^2$ such switching steps.

When performing the algorithm in practice we do not know $u, \theta, q_v, q_{v'}$ exactly and need to approximate them at every step. This adds an error to the algorithm. □

F.1. Monotone predictor

In the following claim we show that a calibrated predictor with a discrete range can be modified to one that is monotone (as in Definition F.2) with high probability, by merging small level sets and level sets that are close together. This claim only holds for functions w with bounded range, although the rest of the section holds more generally. We remark that as long as the hypothesis class H contains bounded functions $h : X \to [0,1]$, then the claim below holds for all classes W defining group or level-set calibration on Section 2.3. In case of group multi-accuracy or calibration with negative value of τ, the claim below should be run on each part $\{x|g(x) = i\}$ separately.

Claim F.7. Let $V \subset [0,1]$ be a discrete set, and let W be a class of functions $w : X \times [0,1] \to [0,1]$ containing a function $f_v(x,v') = \mathbf{1}(v = v')$ for all $v \in V$. Let $p : X \to [0,1]$ be a predictor with a discrete range V such that $p \in \mathsf{GenMC}_{\mathcal{D}}(W,\varepsilon)$. Then there is an algorithm running in time $O(|V|^3 \frac{1}{\varepsilon^2\delta})$, uses $O(|V|^3 \frac{1}{\varepsilon^2\delta})$ samples, that with probability $1-\delta$ outputs a monotone predictor $p' \in \mathsf{GenMC}_{\mathcal{D}}(W, 6\varepsilon)$.

Proof. We describe a simple algorithm for merging the levels of p that are too close to each other or too small. We start by looking at the partition of X defined by p, then merge parts that are too small or too close to each other. Let $P = P_1, \ldots P_{|V|}$ be the partition of x defined by p.

The algorithm sample S of size $O(|V|^3 \frac{1}{\varepsilon^2\delta})$ of $(x,y) \sim \mathcal{D}$, and do:

1. While there exists a part P_i such that $\Pr_{(x,y)\in S}[x \in P_i] < \frac{2\varepsilon}{|V|}$, merge P_i with its neighbor.

2. While there are $P_i, P_j \in P$ such that

$$\left| \mathop{\mathbb{E}}_{(x,y)\in S}[y|x \in P_i] - \mathop{\mathbb{E}}_{(x,y)\in S}[y|x \in P_j] \right| < \frac{2\varepsilon}{|V|},$$

merge P_i, P_j.

3. Set $p' : X \to [0,1]$ by choosing for every part $x \in P_i$ the value $\mathbb{E}_{(x',y')\in S}[y'|x' \in P_i]$.

From Claim J.1, by taking $O(|V|^3 \frac{1}{\varepsilon^2\delta})$, with probability $1-\delta/2$ the algorithm approximates $\Pr_{(x,y)\sim\mathcal{D}}[p(x) = v]$ up to an error of $\frac{\varepsilon}{|V|}$. After the first step of the algorithm, each P_i has size at least $\frac{\varepsilon}{|V|}$. Therefore, from Claim J.1 the algorithm approximates $\mathbb{E}_{(x,y)\in S}[y|x \in P_i]$ up to an additive error of $\frac{\varepsilon}{|V|}$ with probability $1-\delta/2$ for all parts. Assuming all approximations are correct, the predictor p' is monotone. Therefore, p' is monotone with probability at least $1-\delta$.

To prove the generalized calibration, we first use the function $f_v \in W$ and get that for every $v \in V$,

$$\left| \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y - v)\mathbf{1}(p(x) = v)] \right| \le \varepsilon, \tag{54}$$

Assume that the algorithm skips Item 2, and only preforms merging for small sets and asigns new values. Let p'' be this predictor. Then for p'' we have,

$$\begin{aligned}
&\left|\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y-p''(x))w(x,v)]\right| \\
&\leq \left|\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y-p''(x))w(x,v)] + \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(p''(x)-p(x))w(x,v)]\right| \\
&\leq \varepsilon + \left|\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(p''(x)-p(x))w(x,v)]\right| \\
&\leq \varepsilon + \left|\mathop{\Pr}_{(x,y)\sim\mathcal{D}}[p(x) \text{ in small } P_i] + \sum_{\text{large } P_i} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(p''(x)-p(x))w(x,v)\mathbf{1}(x\in P_i)]\right| \\
&\leq 3\varepsilon + \left|\sum_{\text{large } P_i} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(p''(x)-p(x))\mathbf{1}(x\in P_i)]\right| \\
&\leq 3\varepsilon + \left|\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y-v)\mathbf{1}(p(x)=v)]\right| + \sum_{\text{large } P_i}\left|\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[(y-v)\mathbf{1}(p''(x)=v)]\right|. \qquad (55)
\end{aligned}$$

Where large P_i's are those that the algorithm does not merge in Item 1. From Equation (54), the first expectation is bounded by ε. From the paragraph above, with probability at least $1-\delta/2$ the we have $\left|\mathbb{E}_{(x,y)\in S}[y|x\in P_i] - \mathbb{E}_{(x,y)\sim\mathcal{D}}[y|x\in P_i]\right| \leq \varepsilon/\left|V\right|$ for all large partitions P_i. Together we get $\left|\mathbb{E}_{(x,y)\sim\mathcal{D}}[(y-p''(x))w(x,v)]\right| \leq 5\varepsilon$.

Our monotone predictor p' has an extra step in Item 2, in which the algorithm merges parts P_i, P_j. The algorithm only merges parts in which the expected value of y, $\mathbb{E}[y|x\in P_i]$ is within distance $\frac{\varepsilon}{V}$. Therefore, even if we preform $|V|$ merges, we have that

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[|p'(x)-p''(x)|] \leq \varepsilon.$$

Substituting $p'(x)$ instead of $p''(x)$ on equation Equation (55) can only increase the expected value by ε. □

G. Algorithms for Multiaccuracy and Multicalibration

The computational and sample complexity of learning a multiaccuracy/multicalibrated predictor w.r.t. a function class $\mathcal{C}$ using i.i.d. data points from the true distribution $\mathcal{D}$ depends on the complexity and structure of the class $\mathcal{C}$. In (Hébert-Johnson et al., 2018), the authors show that the task can be *efficiently* reduced to *weak agnostic learning* for $\mathcal{C}$ (Kalai et al., 2008; Feldman, 2010). This implies that the sample and computational complexity of learning a multicalibrated predictor cannot be much larger than weak agnostic learning. Hu et al. (2022a) concretely characterize the sample complexity of learning a multiaccurate/multicalibrated predictor in terms of the *fat-shattering dimension* of $\mathcal{C}$ (Kearns & Schapire, 1990), and they also study the sample complexity of multiaccuracy/multicalibration with additional realizability assumptions about $\mathcal{D}$, which is a setting further explored by Hu & Peale (2023) (results in our paper do not require any assumption on $\mathcal{D}$). Gopalan et al. (2023) propose and implement algorithms for calibrated multiaccuracy and demonstrate their efficiency compared to achieving multicalibration. Many of our results in this paper require group multiaccuracy/multicalibration, and such a predictor can be obtained by first learning a multiaccurate/multicalibrated predictor w.r.t. $\mathcal{C}$ on each group and then combining. Some of our results in this paper require group level-set multiaccuracy. This can be equivalently viewed as multiaccuracy w.r.t. a larger class $\mathcal{C}^{(g)}$ of binary functions $c' : X \to \{-1,1\}$ such that there exist $c\in\mathcal{C}$ and $\tau : [t]\times A \to \{-1,1\}$ satisfying $c'(x) = \tau(g(x), c(x))$ for every $x\in X$. The complexity of $\mathcal{C}^{(g)}$ depends on the complexity of $\mathcal{C}$ and the group partition g.

H. Optimization Algorithms on the Simulated Distribution

An omnipredictor p, as in Definition 2.1, allows us to solve downstream tasks $T\in\mathcal{T}$ on the true distribution $\mathcal{D}$ by solving the task on the simulated distribution $\mathcal{D}_p$. In this section, we show very efficient algorithms for solving the task on the simulated distribution for all the settings we consider in Section 4.

Specifically, in Definition 2.1, we define $\beta' := \mathsf{opt}_{\mathcal{D}_p}(T, \mathcal{C}', \varepsilon/3) \in \mathbb{R}$. Suppose the objective of T is $f_0 : X \times A \times$

$\{0,1\} \to \mathbb{R}$ and the constraints of T are $f_j : X \times A \times \{0,1\} \to \mathbb{R}$ for every $j \in J$. The task of finding a solution in $\mathcal{C}' \cap \mathsf{sol}_{\mathcal{D}_p}(T, \beta' + \varepsilon/3, 2\varepsilon/3)$ is to solve the following optimization problem approximately:

$$\begin{aligned} \underset{c \in \mathcal{C}'}{\text{minimize}} \quad & \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c(x)} f_0(x,a,y) \\ \text{s.t.} \quad & \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim c(x)} f_j(x,a,y) \le 0 \quad \text{for every } j \in J. \end{aligned} \tag{56}$$

In Theorems 4.4 and 4.6, the action set A is the interval $[0,1]$, and the objective f_0 and the constraints f_j are convex group objective/constraints. That is, for every $j \in \{0\} \cup J$, there exists $f'_j : [t] \times A \times \{0,1\} \to \mathbb{R}$ such that $f_j(x,a,y) = f'_j(g(x),a,y)$ for every $(x,a,y) \in X \times A \times \{0,1\}$, and the function $f'_j(i,\cdot,y)$ is convex for every $i \in [t]$ and $y \in \{0,1\}$. Moreover, the class $\mathcal{C}'$ is the class $\mathcal{C}_{p,g}$ in Definition 4.3, i.e., $\mathcal{C}'$ consists of all functions $c : X \to A$ such that there exists $\tau : [t] \times [0,1] \to A$ satisfying $c(x) = \tau(g(x), p(x))$ for every $x \in X$. Thus, (56) becomes the following equivalent problem:

$$\begin{aligned} \underset{\tau:[t]\times[0,1]\to[0,1]}{\text{minimize}} \quad & \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f'_0(g(x), \tau(g(x),p(x)), y) \\ \text{s.t.} \quad & \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} f'_j(g(x), \tau(g(x),p(x)), y) \le \varepsilon/3 \quad \text{for every } j \in J. \end{aligned} \tag{57}$$

Let $V := \mathsf{range}(p)$ denote the range of p. Since the functions $c \in \mathcal{C}$ in Theorem 4.4 and Theorem 4.6 output bounded values $c(x) \in A = [0,1]$, we can always make sure that V is finite and has size $O(1/\varepsilon')$ when we require p to be $(\mathcal{C}, g, \varepsilon')$-multiaccurate and/or $(\mathcal{C}, g, \varepsilon')$-multicalibrated because discretizing the values $p(x)$ to multiples of $\varepsilon'/2$ can only increase the group multiaccuracy/multicalibration error by at most $\varepsilon'/2$. Let $\mathsf{prob}(i,v)$ denote $\Pr_{(x,y)\sim\mathcal{D}_p}[g(x) = i, p(x) = v]$. The optimization problem (57) above is equivalent to

$$\begin{aligned} \underset{\tau:[t]\times V\to[0,1]}{\text{minimize}} \quad & \sum_{i\in[t]} \sum_{v\in V} \mathsf{prob}(i,v) \Big(v f'_0(i, \tau(i,v), 1) + (1-v) f'_0(i, \tau(i,v), 0) \Big) \\ \text{s.t.} \quad & \sum_{i\in[t]} \sum_{v\in V} \mathsf{prob}(i,v) \Big(v f'_j(i, \tau(i,v), 1) + (1-v) f'_j(i, \tau(i,v), 0) \Big) \le \varepsilon/3 \quad \text{for every } j \in J. \end{aligned} \tag{58}$$

Suppose for now that we know the probabilities $\mathsf{prob}(i,v)$. The optimization problem (58) above is a convex program with size $O(t\,|V| \cdot |J|)$ and thus can be solved efficiently assuming that we can efficiently compute f'_j for every $j \in \{0\} \cup J$ and its sub-gradient. When we do not know $\mathsf{prob}(i,v)$, we can estimate it to sufficient accuracy using i.i.d. data points from the marginal distribution of x in $(x,y) \sim \mathcal{D}_p$, which is the same marginal distribution of x in $(x,y) \sim \mathcal{D}$. Thus these data points are exactly unlabeled data points from the true distribution $\mathcal{D}$. By standard concentration results (e.g. Claim J.1), using $n = O(\varepsilon_1^{-2}(|V|t + \log(1/\delta)))$ data points we can compute an estimate $\mathsf{est}(i,v)$ for each $\mathsf{prob}(i,v)$ such that with probability at least $1 - \delta$,

$$\sum_{i\in[t]} \sum_{v\in V} |\mathsf{est}(i,v) - \mathsf{prob}(i,v)| \le \varepsilon_1/3.$$

The computation of these estimates est is independent of the actual loss minimization task. Thus it can be done when we train the omnipredictor p, in which case no data is needed when solving a downstream task using p and these estimates.

In Theorems 4.7 and D.9, the action set A is a finite set, and the objective f_0 and the constraints f_j are group objective/constraints. The class $\mathcal{C}'$ is the class $\mathcal{C}^{\mathsf{rand}}_{p,g}$, i.e., $\mathcal{C}'$ consists of all functions $c : X \to \Delta_A$ such that there exists $\tau : [t] \times [0,1] \to \Delta_A$ satisfying $c(x) = \tau(g(x), p(x))$ for every $x \in X$. Thus, (56) becomes the following equivalent problem:

$$\begin{aligned} \underset{\tau:[t]\times[0,1]\to\Delta_A}{\text{minimize}} \quad & \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim\tau(g(x),p(x))} f'_0(g(x), a, y) \\ \text{s.t.} \quad & \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_p} \mathop{\mathbb{E}}_{a\sim\tau(g(x),p(x))} f'_j(g(x), a, y) \le \varepsilon/3 \quad \text{for every } j \in J. \end{aligned} \tag{59}$$

Defining V and $\mathsf{prob}(i,v)$ as before and using $\tau'(i,v,a)$ to denote the probability mass on $a \in A$ in $\tau(i,v)$, the optimization problem (59) above is equivalent to the following:

$$\underset{\tau':[t]\times V\times A\to\mathbb{R}}{\text{minimize}} \quad \sum_{i\in[t]} \sum_{v\in V} \sum_{a\in A} \mathsf{prob}(i,v)\tau'(i,v,a) \Big(v f'_0(i,a,1) + (1-v) f'_0(i,a,0) \Big) \tag{60}$$

$$\begin{aligned}
\text{s.t.} \quad & \sum_{i\in[t]}\sum_{v\in V}\sum_{a\in A}\mathsf{prob}(i,v)\tau'(i,v,a)\Big(vf_j'(i,a,1)+(1-v)f_j'(i,a,0)\Big)\le\varepsilon/3, && \forall j\in J,\\
& \sum_{a\in A}\tau'(i,v,a)=1, && \forall(i,v)\in[t]\times V,\\
& \tau'(i,v,a)\ge 0, && \forall(i,v,a)\in[t]\times V\times A.
\end{aligned}$$

This optimization problem (60) is a linear program of size $O(t\,|V|\cdot|A|\cdot|J|)$ and thus can be solved efficiently.

I. Counterexamples

I.1. Group Multiaccuracy is Necessary

We show that the group multiaccuracy and group calibration assumptions in Theorem 4.4 cannot be replaced by standard (non-group-wise) multicalibration.

Claim I.1. Let $A=[0,1]$ be an action set. There exists a non-empty set X over individuals, a group partition function $g: X\to[t]$, a distribution $\mathcal{D}$ over $X\times\{0,1\}$, a task T, a class $\mathcal{C}$ of functions $c: X\to A$, a predictor $p: X\to[0,1]$ with the following properties. The task T has the ℓ_1 objective $f_0(x,a,y)=|a-y|$ and linear constraints (as in (4)). The predictor p belongs to $\mathsf{MC}(\mathcal{C},0)\cap\mathsf{Cal}(0)$. However, p is not a $(\{T\},\mathcal{C},\mathcal{C}_{p,g},\varepsilon)$-omnipredictor for sufficiently small $\varepsilon>0$.

Proof. We assume that $X=\{x_1,x_2,x_3,x_4\}$ and $(x,y)\sim\mathcal{D}$ can be sampled by first drawing x from the uniform distribution over X, and then drawing $y\sim\mathsf{Ber}(p^*(x))$ for

$$p^*(x)=\begin{cases}0.5, & \text{if } x=x_1,\\ 0.5, & \text{if } x=x_2,\\ 0, & \text{if } x=x_3,\\ 1, & \text{if } x=x_4.\end{cases}$$

The function class $\mathcal{C}$ consists of a single function c defined by

$$c(x)=\begin{cases}0.75, & x=x_1,\\ 0.25, & x=x_2,\\ 0, & x\in\{x_3,x_4\}.\end{cases}$$

The groups are defined by

$$g(x)=\begin{cases}1, & x\in\{x_1,x_3\},\\ 2, & x\in\{x_2,x_4\}.\end{cases}$$

The constraints f_j of the task T are defined by

$$\begin{aligned}
f_1(x,a,y)&=\mathbf{1}(i=1)0.375-\mathbf{1}(i=1)a\\
f_2(x,a,y)&=-\mathbf{1}(i=1)0.375+\mathbf{1}(i=1)a\\
f_3(x,a,y)&=\mathbf{1}(i=2)0.125-\mathbf{1}(i=2)a\\
f_4(x,a,y)&=-\mathbf{1}(i=2)0.125+\mathbf{1}(i=2)a
\end{aligned}$$

That is, they require that $\mathbb{E}[c(x)|g(x)=1]=0.375, \mathbb{E}[c(x)|g(x)=2]=0.125$. We can easily see that c satisfies the constraint:

$$\begin{aligned}
\mathbb{E}_x[c(x)|g(x)=1]&=0.75\cdot 0.5=0.375,\\
\mathbb{E}_x[c(x)|g(x)=2]&=0.25\cdot 0.5=0.125.
\end{aligned}$$

We choose $p : X \to [0,1]$ to be the constant function satisfying $p(x) = 0.5$ for all $x \in X$. We show that $p \in \mathsf{MC}(\mathcal{C}, 0) \cap \mathsf{Cal}(0)$. We start from calibration:

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[y] = 0.5 = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[p(x)].$$

Now we show multicalibration with respect to $c \in \mathcal{C}$:

$$\begin{aligned}&\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[c(x)\cdot(y-p(x))]\\ =\ &\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[c(x)\cdot(y-0.5)]\\ =\ &0.25\left(0.75(0.5-0.5)+0.25(0.5-0.5)+0\cdot(0-0.5)+0\cdot(1-0.5)\right)=0.\end{aligned}$$

The objective value of c is:

$$\begin{aligned}\beta &:= \mathsf{opt}_{\mathcal{D}}(T,\mathcal{C},0)\\ &= \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[f_0(i,c(x),y)]\\ &= 0.125\left(|1-0.75|+|0-0.75|+|1-0.25|+|0-0.25|\right)+0.25\left(|0,0|+|1,0|\right)\\ &= 0.125\left(2\cdot 0.25+2\cdot 0.75\right)+0.25 = 0.25+0.25\\ &= 0.5.\end{aligned}$$

Since p is a constant function, any $c' \in \mathcal{C}_{p,g}$ must satisfy $c'(x_1) = c'(x_3)$ and $c'(x_2) = c'(x_4)$ because $g(x_1) = g(x_3)$ and $g(x_2) = g(x_4)$. To satisfy the constraints up to a small error ε, c' must be close to assigning 0.375 to x_1 and x_3, and assigning 0.125 to x_2 and x_4. We calculate the loss for this c':

$$\begin{aligned}\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[f_0(i,c'(x),y)] &= 0.125\left(|1-0.375|+|0-0.375|+|1-0.125|+|0-0.125|\right)\\ &\quad+0.25\left(|0-0.325|+|1-0.125|\right)\\ &= 0.125\left(0.625+0.375+0.125+0.875\right)+0.25\left(0.375+0.875\right)\\ &= 0.25+0.25\cdot 1.25\\ &= 0.5625\\ &> \beta.\end{aligned}$$

This implies that for small enough ε, we have $\mathcal{C}_{p,g} \cap \mathsf{sol}_{\mathcal{D}}(T, \beta+\varepsilon, \varepsilon) = \emptyset$, and thus p cannot be a $(\{T\}, \mathcal{C}, \mathcal{C}_{p,g}, \varepsilon)$-omnipredictor. □

I.2. Group Level-Set Multiaccuracy is Necessary

We show an example task with non-convex constraints and a non-special objective, and thus none of our Theorems 4.4, 4.6 and D.9 could be applied to the example. Theorem 4.7 is applicable, but it requires group level-set multiaccuracy. Below we show that for this task group multicalibration is indeed not enough and the level-set variant is necessary to guarantee omniprediction.

Claim I.2. Let $A = [0,1]$ be an action set. There exists a non-empty set X over individuals, a group partition function $g : X \to [t]$, a distribution $\mathcal{D}$ over $X \times \{0,1\}$, a task T, a class $\mathcal{C}$ of functions $c : X \to A$, a predictor $p : X \to [0,1]$ with the following properties. The task T only has group constraints and objectives with 1-bounded differences. The predictor p belongs to $\mathsf{GrpMC}_{\mathcal{D}}(\mathcal{C}, g, 0) \cap \mathsf{GrpCal}_{\mathcal{D}}(g, 0)$. However, p is not a $(\{T\}, \mathcal{C}, \mathcal{C}^{\mathsf{rand}}_{p,g}, \varepsilon)$-omnipredictor for sufficiently small $\varepsilon > 0$.

Proof. Let $X = \{x_1, x_2, x_3\}$ and let $g : X \to [t]$ be the trivial group partition that assigns every individual $x \in X$ to the same group $g(x) = 1$. The distribution $\mathcal{D}$ is defined by first choosing $x \in X$ uniformly at random, and then choosing $y \sim \mathsf{Ber}(p^*(x))$ for

$$p^*(x) = \begin{cases} 0.25, & x = x_1,\\ 1, & x = x_2,\\ 0.25, & x = x_3.\end{cases}$$

The function class $\mathcal{C}$ contains only a single function $\mathcal{C} = \{c\}$ defined by:

$$c(x) = \begin{cases} 0.1, & x = x_1, \\ 0.2, & x = x_2, \\ 0.3, & x = x_3. \end{cases}$$

We choose the objective f_0 of T to be the cubic loss: $f_0(x, a, y) = |a - y|^3$. We choose the collection of constraints f_j of T to be

$$\begin{aligned} f_1(x, a, y) &= \mathbf{1}(a = 0.1) - \frac{1}{3} \\ f_2(x, a, y) &= -\mathbf{1}(a = 0.1) + \frac{1}{3} \\ f_3(x, a, y) &= \mathbf{1}(a = 0.2) - \frac{1}{3} \\ f_4(x, a, y) &= -\mathbf{1}(a = 0.2) + \frac{1}{3} \\ f_5(x, a, y) &= \mathbf{1}(a = 0.3) - \frac{1}{3} \\ f_6(x, a, y) &= -\mathbf{1}(a = 0.3) + \frac{1}{3} \end{aligned}$$

For an action function $c' : X \to A$ to satisfy these constraints exactly, it must satisfy

$$\Pr_{(x,y)\sim\mathcal{D}}[c'(x) = a] = 1/3 \quad \text{for every } a \in \{0.1, 0.2, 0.3\}\,.$$

It is clear that the only function $c \in \mathcal{C}$ satisfies the constraints. The objective value achieved by c is

$$\begin{aligned} \beta := \mathsf{opt}_{\mathcal{D}}(T, \mathcal{C}, 0) &= \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[f_0(x, c(x), y)] \\ &= \sum_j \Pr[x = x_j] \left(\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[y|x \in U_j] |1 - c_j|^3 + (1 - \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[y|x \in U_j]) |c_j|^3 \right) \\ &= \frac{1}{3} \left(\frac{1}{4}(0.9)^3 + \frac{3}{4}(0.1)^3 + 1(0.8)^3 + 0(0.2)^3 + \frac{1}{4}(0.7)^3 + \frac{3}{4}(0.3)^3 \right) \\ &= 0.267. \end{aligned}$$

The predictor $p : X \to [0, 1]$ defined by $p(x) = 0.5$ for all $x \in X$. We show that $p \in \mathsf{GrpMC}_{\mathcal{D}}(\mathcal{C}, g, 0) \cap \mathsf{GrpCal}_{\mathcal{D}}(g, 0)$. We show it, starting from calibration:

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[y] = 0.5 = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[p(x)].$$

For group multicalibration with respect to $c \in \mathcal{C}$:

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[c(x)\,(y - p(x))] = \frac{1}{3} \left(-\frac{1}{10} \cdot \frac{1}{4} + \frac{2}{10} \cdot \frac{1}{2} - \frac{3}{10} \cdot \frac{1}{4} \right) = 0$$

Since both p and g are constant functions, any $c' \in \mathcal{C}^{\mathsf{rand}}_{p,g}$ has to give all $x \in X$ the same distribution $c(x)$ of actions. To satisfy the constraints up to a small error ε, $c'(x)$ must be close to the uniform distribution over $\{0.1, 0.2, 0, 3\}$ for every x. When $c'(x)$ is this uniform distribution for every x, we have

$$\begin{aligned} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{a\sim c(x)}[f_0(x, a, y)] &= \sum_{b\in\{0,1\}, a\in\{0.1,0.2,0.3\}} \Pr_{(x,y)\sim\mathcal{D}}[y = b, c(x) = a]\, |y - a|^3 \\ &= \frac{1}{2} \cdot \frac{1}{3} \left((0.9)^3 + (0.1)^3 + (0.8)^3 + (0.2)^3 + (0.7)^3 + (0.3)^3 \right) \end{aligned}$$

$$\begin{aligned} &= 0.27 \\ &> \beta. \end{aligned}$$

Therefore, for small enough $\varepsilon > 0$, we have $\mathcal{C}^{\mathsf{rand}}_{p,g} \cap \mathsf{sol}_{\mathcal{D}}(T, \beta + \varepsilon, \varepsilon) = \emptyset$, and thus p cannot be a $(\{T\}, \mathcal{C}, \mathcal{C}^{\mathsf{rand}}_{p,g}, \varepsilon)$-omnipredictor. □

J. Helper Claims

The following claim is a standard result (see e.g. (Canonne, 2020, Theorem 1)):

Claim J.1. Let Z be a non-empty set partitioned into $Z^{(1)}, \ldots, Z^{(m)}$. For $\varepsilon, \delta \in (0, 1/2)$ and an integer $n \geq W(\varepsilon^{-2}(m + \log(1/\delta)))$ for a sufficiently large absolute constant $W > 0$, let $z_1, \ldots, z_n \in Z$ be n data points drawn i.i.d. from any distribution $\mathcal{D}$ over Z. Then with probability at least $1 - \delta$, the following inequality holds:

$$\sum_{j=1}^{m} \left| \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}(z_i \in Z^{(j)}) - \Pr_{z \sim \mathcal{D}}[z \in Z^{(j)}] \right| \leq \varepsilon.$$