Popularizing Fairness: Group Fairness and Individual Welfare

Andrew Estornell[1], Sanmay Das[2], Brendan Juba[1], Yevgeniy Vorobeychik[1]

[1] Washington University in Saint Louis
[2] George Mason University
{aestornell, yvorobeychik, bjuba}@wustl.edu, sanmay@gmu

Abstract

Group-fair learning methods typically seek to ensure that some measure of prediction efficacy for (often historically) disadvantaged minority groups is comparable to that for the majority of the population. When a principal seeks to adopt a group-fair approach to replace another, more conventional approach, the principal may face opposition from those who feel that they have been disadvantaged as a result of the switch, and this, in turn, may deter adoption. We propose to mitigate this concern by ensuring that a group-fair model is also *popular*, in the sense that it yields a preferred distribution over outcomes compared with the conventional model for a majority of the target population. First, we show that state of the art fair learning approaches are often unpopular in this sense. We then present several efficient algorithms for postprocessing an existing group-fair learning scheme to improve its popularity while retaining fairness. Through extensive experiments, we demonstrate that the proposed postprocessing approaches are highly effective.

Introduction

Increasing adoption of machine learning approaches in high-stakes domains, such as healthcare and social assistance, has led to increased scrutiny of their impact on vulnerable groups. A number of studies demonstrating the disparate impact of automation on such groups (Citron and Pasquale 2014; Angwin et al. 2016; Dastin 2018; Lee 2018; Koenecke et al. 2020) has motivated an extensive literature that aims at achieving group fairness of machine learning (Kearns et al. 2018; Agarwal et al. 2018; Pleiss et al. 2017; Hardt, Price, and Srebro 2016; Chouldechova and Roth 2018; Mehrabi et al. 2021; Barocas, Hardt, and Narayanan 2017; Angwin et al. 2016; Dwork et al. 2012) by imposing an explicit constraint that prediction efficacy (which can be measured in many different ways) is similar across groups. However, a principal contemplating a change from a conventional, and potentially biased, prediction model to a group-fair approach must contend with the perception that such a switch could inadvertently harm many individuals in the process of improving fairness (for example, making them less likely to receive a scarce resource such as a welfare benefit or admission to a college). Such perceptions could make any change from the status quo contentious and, consequently, less likely. Our central question is whether it is possible to make group-fair classifiers sufficiently *popular*—reducing the prevalence of realized or perceived harm—so as to make their adoption less contested.

We model the principal's problem as a comparison between a conventional approach f_C and a group-fair approach f_F, with the principal considering a switch from the former to the latter. Both algorithms select a subset of individuals from a target population to obtain a particular desirable outcome (e.g., a resource, such as admission to a college). We examine popularity in this context through the lens of preferences of individuals in a target population over selection outcomes (which we can encode as positive outcomes of binary classification): an individual *weakly* prefers f_F to f_C if the probability of being selected is not lower under the former than under the latter. Popularity of a group-fair approach f_F then amounts to ensuring that a given fraction (e.g., majority) of a target population prefers f_F to f_C.

To illustrate the relationship between fairness, accuracy, and popularity, consider the following example. Let G_1 and G_0 have four and two members respectively, with true labels $\langle 1,1,1,1\rangle$ and $\langle 0,0\rangle$. A randomized conventional classifier f_C, predicts each member of G_1 to be positive with probability 0.75 and each member of G_0 to be positive with probability 0.25. Under demographic parity fairness, G_1 is advantaged as this group has a positive rate 0.5 greater than that of G_0. Consider two choices for a fair model. f_{F_1} predicts members of G_1 to be positive with probability 0.75 and members of G_0 to be positive with probability 0.55. f_{F_2} predicts one member of G_1 to be positive with probability 1, and the others with probability $\frac{2}{3}$; it predicts one member of G_0 to be positive with probability 1 and the other with probability 0.1. Note that both models have identical accuracy and unfairness, namely $.65$ and $.2$ respectively. However, f_{F_1} has *not* decreased the score of any agent in the population; all six prefer f_{F_1} at least as much as the original f_C. In contrast, f_{F_2} has decreased the scores of three agents from G_1 and one agent from G_0; only two agents prefer f_{F_2} at least as much as f_C. This example illustrates that popularity should be viewed as a different axis than either accuracy or fairness, and there may be space to innovate by enabling popularity comparisons among fair(er) models.

We start this paper by asking an empirical question: Do typical group-fair classification approaches yield models that are, in fact, unpopular in the sense above? We demonstrate that they are: in experiments on several standard datasets, more than half the target population can strictly prefer the conventional scheme to several prominent group-fair learning methods. Given that the group-fair approaches have significant motivation and momentum behind them, instead of designing an entirely new approach to finding popular and fair classifier, we ask whether it is possible to *minimally postprocess* the output of a group-fair classifier in order to achieve some target popularity while maintaining a high level of fairness. We answer this question in the affirmative. Specifically, we describe two approaches to efficiently postprocess the outputs from a given group-fair classifier in order to boost its popularity. The first approach formalizes the problem as a minimal change of outcome probabilities over the target population to guarantee a target level of fairness and popularity. We show that this problem can be solved in polynomial time. Our second approach involves a form of regularized empirical risk minimization with fairness and popularity constraints. This approach relies on partitioning prediction scores into a set of quantiles, and we show that, in general, the problem is strongly NP-Hard. However, we also show that if the number of quantiles is constant, this problem can be solved in polynomial time. Our methods are applicable in both the classification and scarce resource allocation settings, and allow a model designer to directly control the level of popularity and fairness.

In summary, our contributions are:

1. We propose the notion of *popularity* of group-fair classifiers and allocation schemes, measuring the fraction of a population that is weakly better off when switching from a conventional to a fair learning scheme.
2. We demonstrate the degree to which state of the art group-fair approaches are *unpopular* compared to their conventional counterparts.
3. We introduce two postprocessing algorithms which allow a principal to directly control the popularity of a given fair model, while maintaining good fairness properties. The first post-processing technique, dubbed DOS (Direct Outcome Shift), is polynomial time solvable for both deterministic and randomized classifiers, and can also be applied to the scarce resource allocation setting. The second technique, k-QLS (k-Quantile Lottery Shift), works by grouping agents into k quantiles (where k is chosen by the model designer), and running lotteries on each quantile. k-QLS is polynomial time solvable for deterministic classifiers. While we show that k-QLS is NP-hard in the randomized case, it becomes tractable for constant k, as would be standard in practice.
4. We empirically demonstrate that the proposed postprocessing techniques can achieve high levels of popularity and fairness with minimal impact on prediction accuracy.

Related Work: Our work is broadly related to the field of algorithmic group fairness which is concerned with both defining what it means for a model to be fair, as well as operationalizing these definitions to produce fair models (Hardt, Price, and Srebro 2016; Pleiss et al. 2017; Feldman et al. 2015; Dwork et al. 2012; Agarwal et al. 2018; Kearns et al. 2018; Jang, Shi, and Wang 2021; Kusner et al. 2017). In particular our algorithms work through postprocessing, a common technique for for achieving fairness (Pleiss et al. 2017; Hardt, Price, and Srebro 2016; Kamiran, Karim, and Zhang 2012; Canetti et al. 2019; Lohia et al. 2019; Jang, Shi, and Wang 2021). In these works, the scores or decisions of a conventional classifier are modified in order to achieve fairness. Most post processing techniques for fairness work through "inclusion/exclusion" systems where a potentially randomized procedure is uniformly applied across groups, e.g. random selection of group-specific thresholds (Hardt, Price, and Srebro 2016; Jang, Shi, and Wang 2021), or randomly selecting agents from one group to receive positive classification with constant probability (Pleiss et al. 2017). Our postprocessing techniques, while concerned not exclusively with fairness, follow a similar inclusion/exclusion system.

More generally, randomized prediction methods are common in prior literature. In some cases, randomization is inherently desirable, for example, to explore or correct existing bias in domains such as hiring (Berger et al. 2020; Tassier and Menczer 2008; Hong and Page 2004) or lending (Karlan and Zinman 2010a,b). In other settings, the aim is to increase model robustness (Pinot et al. 2019; Salman et al. 2019), or to achieve better trade-offs between model performance and fairness, as is common in many group-fair classification approaches (Agarwal et al. 2018; Kearns et al. 2018; Pleiss et al. 2017).

Several recent papers look at the potential negative consequences of applying group fairness (Liu et al. 2018; Zhang et al. 2020; Corbett-Davies and Goel 2018; Kasy and Abebe 2021; Ben-Porat, Sandomirskiy, and Tennenholtz 2019). In particular (Liu et al. 2018; Ben-Porat, Sandomirskiy, and Tennenholtz 2019) demonstrate that specific types of group equity can be decreased by the use of fair algorithms. Others have merged notions of welfare and fairness (Hu and Chen 2020; Cousins 2021; Xinying Chen and Hooker 2023). Both the notion of popularity, as well as our proposed techniques for satisfying popularity and fairness, differ from these lines of work in that popularity casts welfare in terms of the fraction of a population which prefers a fair model compared with a fairness-agnostic model. While the idea of agent preference over models has received some recent attention (Ustun, Liu, and Parkes 2019) (which aims to classify a population using multiple models such that each agent prefers their assigned model over all others), popularity in the context of group fair learning has remained unexplored thus far.

Preliminaries

We begin by formalizing our models of conventional and fair learning, as well as our definition of popularity. Let $D = (\mathbf{X}, Y, G)$ be a dataset of n examples where the i^{th} example $(\mathbf{x}_i, y_i, g_i)$ consists of features $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^d$, binary labels $y_i \in \{0, 1\}$ and binary group membership $g_i \in \{0, 1\}$. We assume throughout that the positive label $y = 1$ corresponds to the *preferred outcome*, such as being selected to receive a valuable resource (e.g., college admission). Consider two learning schemes, say C and F, where

C is a *conventional* learning scheme, designed to minimize some fairness-agnostic objective, and F is a *fair* learning scheme, designed to achieve a desired level of fairness between groups. Then, C solves a problem of the form:

$$f_C \in \arg\min_{f\in\mathcal{H}_C} \mathcal{L}_C\big(f, \mathbf{X}, Y\big) \tag{1}$$

i.e., choosing an optimal model f_C from the hypothesis class $\mathcal{H}_C$ with respect to the loss $\mathcal{L}_C$. However, we do not *require* that f_C is the result of strict error minimization, only that it maps $\mathcal{X}$ to $\{0,1\}$. In the context of conventional learning, the objective $\mathcal{L}_C$ and learning scheme C may have exogenous considerations aside from error minimization, such as robustness or interoperability.

We further assume that the learned classifier is of the type that produces a *score function* $h : \mathcal{X} \rightarrow [0,1]$ which is used to induce the classification $f(\mathbf{X})$. Most classifiers used in practice yield such score functions (e.g., SVM, Logistic Regression, Neural Nets, Decision Trees, etc.). We study both deterministic and randomized classifiers in this framework. While deterministic predictions are most common, randomization can offer flexibility that can play a useful role both in achieving fairness (Dwork et al. 2012; Kearns et al. 2018) and robustness (Pinot et al. 2019; Li and Vorobeychik 2015; Salman et al. 2019; Vorobeychik and Li 2014). A deterministic classifier f can be thought of as a threshold on scores from h, i.e., $f(\mathbf{x}) = \mathbb{I}\big[h(\mathbf{x}) \geq \theta\big]$ for threshold θ. A randomized classifier f, in turn, can be viewed as a Bernoulli random variable with a mean given by h, i.e., $\mathbb{E}[f(\mathbf{x})] = h(\mathbf{x})$.

In addition to the classification setting above in which, in principle, anyone can be selected (i.e., assigned a positive outcome $y = 1$), we consider scarce resource allocation (henceforth simply *allocation*). In the allocation setting, unlike the classification setting, the model designer is limited in the number of positive predictions—that is, the number of individuals that can be selected. Specifically, the score function h is used to allocated $k < n$ homogeneous, indivisible, goods among a population of n agents. This follows a well-established paradigm of allocating scarce resources among individuals using a score function learned on a binary prediction task (Kube, Das, and Fowler 2019). Let $I_i(\mathbf{X}, h, k) \in \{0,1\}$ indicate allocation of a resource to agent i when score function h is applied to a population $\mathbf{X}$ and there are k resources. Similar to the classification setting, the allocation function I can be deterministic or randomized. In the case of deterministic allocation, I_i is obtained directly from the set of scores $h(\mathbf{X})$, e.g., allocating resources to the k highest scoring individuals. In randomized allocation, I_i is a Bernoulli random variable, but unlike in the classification setting, I_i may have an arbitrary joint relationship with allocation decisions made for other agents, e.g., sampling without replacement weighted by $h(\mathbf{X})$.

Let $\mathcal{M}(f(\mathbf{X}), Y; g)$ be an efficacy metric computed with respect to group membership $g \in \{0,1\}$ (for example, false positive rate (FPR) or error rate (ERR)). Define *group disparity* $\mathcal{U}(f, D) = |\mathcal{M}(f(\mathbf{X}), Y; 1) - \mathcal{M}(f(\mathbf{X}), Y; 0)|$, i.e., the difference in efficacy between two groups. Then the group-fair learning scheme F solves a problem of the form

$$f_F = \arg\min_{f\in\mathcal{H}_F} \mathcal{L}_F\big(f, \mathbf{X}, Y\big) \quad \text{s.t. } \mathcal{U}(f, D) \leq \beta. \tag{2}$$

i.e., f_F is an optimal *group-fair* model from hypothesis class $\mathcal{H}_F$, with fairness captured by the constraint that group disparity $\mathcal{U}$ is bounded by β. We refer to the fair learning scheme and model f_F as β-fair. Note that when resources are scarce, fairness is defined over allocation decisions $I_i(\mathbf{X}, h, k)$, not over scores h; an example of a fairness objective would be selection rate parity of $I_i(\mathbf{X}, h, k)$ between groups. Our analysis that follows applies to the broad class of *additive efficacy metrics* in both the classification and allocation settings.

Definition 0.1. *(Additive Efficacy Metric): An efficacy metric $\mathcal{M}$ is* additive *if for any population* $(\mathbf{X}, Y, G)$*,*

$$\mathcal{M}\big(f(\mathbf{X}), Y; g\big) = \sum_{y\in\{0,1\}} \sum_{\substack{i\in G_g:\\ y_i=y}} f(\mathbf{x}_i)c^{(g)}_{y,1} + (1\text{-}f(\mathbf{x}_i))c^{(g)}_{y,0}$$

for some $c^{(g)}_{y,0}, c^{(g)}_{y,1} \in [0,1]$*. In the case of scarce resources* $f(\mathbf{x}_i)$ *is interchangeable with* $I_i(\mathbf{X}, h, k)$*. In the case of randomized models,* $f(\mathbf{x}_i)$ *is replaced with* $\mathbb{E}[f(\mathbf{x}_i)]$ *or* $\mathbb{E}[I_i(\mathbf{X}, h, k)]$.

In an additive efficacy metric, the coefficients $c^{(g)}_{y,0}, c^{(g)}_{y,1}$ give the respective "costs" of classifying an example from group G_g, with true label y, as negative or positive, respectively. Thus, unfairness $\mathcal{U}$ is given as the difference in the total efficacy cost between groups. Additive metrics are widely studied in the literature and include metrics such as error rate (ER), positive (or selection) rate (PR), false positive rate (FPR), and true positive rate (TPR). As an example, in the case of PR fairness $c^{(g)}_{y,1} = {}^{1}/_{|G_g|}$ and $c^{(g)}_{y,0} = 0$ for each $y, g \in \{0,1\}$.

We consider the situation in which a conventional learning scheme C is initially in place, and a *principal* considers a switch from C to a group-fair scheme F, and wishes to ensure that F is *γ-popular* in the sense that it is preferred to C by at least a fraction γ of the target population. We formalize preference over learning schemes by assuming that an individual prefers schemes which yield higher expected outcomes for them, that is, they prefer being selected to not being selected, as in Hardt, Price, and Srebro (2016). Thus, an individual i with features $\mathbf{x}_i$ prefers F over C if

$$f_C(\mathbf{x}_i) \leq f_F(\mathbf{x}_i) \text{ or } I_{C,i}(\mathbf{X}, h, k) \leq I_{F,i}(\mathbf{X}, h, k) \tag{3}$$

when decisions are deterministic and

$$\begin{aligned} \mathbb{E}\big[f_C(\mathbf{x}_i)\big] \leq \mathbb{E}\big[f_F(\mathbf{x}_i)\big] \quad &\text{or} \\ \mathbb{E}\big[I_{C,i}(\mathbf{X}, h, k)\big] \leq \mathbb{E}\big[I_{F,i}(\mathbf{X}, h, k)\big]& \end{aligned} \tag{4}$$

when decisions are stochastic.

Note that our analysis is in the space of outcomes, rather than scores. Consequently, if decisions are deterministic, either in classification or allocation settings, agents only have a definitive preference over scores produced by h if this is consequential to outcomes (e.g., pushing them above or below θ). In the stochastic case, on the other hand, agents prefer the classifier or allocation scheme which yields the higher expected outcome (that is, higher probability of being selected).

Armed with this model of individual preference, we now define what it means for F to be popular.

Definition 0.2. ***(γ-popularity)****: A learning scheme F is said to be γ-popular with respect to a population $(\mathbf{X}, Y, G)$ and conventional scheme C, if Condition* (3) *(for deterministic models), or Condition* (4) *(for randomized models), holds for at least $\gamma|\mathbf{X}|$ individuals.*

Popularity thus captures the fraction γ of a population which is weakly better off (or, equivalently, *not* made worse) from the use of F over C. Similar to the concept of β-fairness, in which a model designer can specify the desired level of fairness β, the definition of popularity, as well as our postprocessing techniques described later, allow the model designer to *directly* specify, and control, the desired level of popularity. Note that we do not capture the *degree* to which individuals are made better or worse off as a result of switching from C to F, but only *whether* they are.

As mentioned earlier, our setting is one of a concrete choice by a principal between a particular conventional approach C and a particular group-fair approach F. This reflects a decision by the principal to switch from C—which is currently deployed—to F in order to reduce impact to a disadvantaged group (or groups). Of course, different pairs of C and F (e.g., using different loss functions, different learning algorithms, etc) would yield different judgments about popularity of F, which is, by construction, relative to C. Consequently, these will also yield different decisions about improving popularity of F based on algorithms we discuss below. Nevertheless, our framework generalizes immediately to a setting in which neither C nor F are fixed, and there is uncertain about either, or both. In such a case, we treat uncertainty about either C or F as a distribution over approaches and, consequently, over outcomes induced. This can then be immediately captured within our framework dealing with randomized schemes, and all definitions above, and technical results below, go through unchanged.

Our goal is to investigate the following three questions: 1) Are common group-fair learning techniques popular? 2) For a given γ and β, can we compute β-fair and γ-popular decisions in polynomial time? 3) What is the nature of the tradeoff between popularity, fairness, and accuracy?

Improving Popularity through Postprocessing

We consider two approaches to minimally postprocess a β-fair scheme f_F such that the resulting decisions also become γ-popular, for exogenously specified β and γ: 1) *direct outcome shift (DOS)* and 2) *k-quantile lottery shift (k-QLS)*. Postprocessing is performed in a transductive setting, in which the populations' features $(\mathbf{X}, G)$ (and possibly also labels Y) are known in advance. Throughout, we use f_P to refer to either approach we propose that combines both popularity and group fairness.

Direct Outcome Shift (DOS) DOS-based postprocessing arises from solving the problem of finding a minimal perturbation to the agents' outcomes that achieves both fairness and popularity, e.g. Program 5 for randomized classification. For a target population with feature vectors $\mathbf{X}$, we shift individuals' outcomes $f_F(\mathbf{X})$ or expected outcomes $\mathbb{E}[f_F(\mathbf{X})]$ by a *perturbation* vector $\mathbf{p}$. For deterministic decisions, $\mathbf{p} \in \{-1, 0, 1\}^n$, while for stochastic decisions $\mathbf{p} \in [-1, 1]^n$. The optimization goal in either case is to minimize $\|\mathbf{p}\|_q$ for some ℓ_q-norm ($q \in \{1, 2, \infty\}$) such that the final decisions, whether they involve predictions ($f_F(\mathbf{X})+\mathbf{p}$, or $\mathbb{E}[f_F(\mathbf{X})]+\mathbf{p}$) or allocations ($I(\mathbf{X}, h, k)+\mathbf{p}$, or $\mathbb{E}\big[I(\mathbf{X}, h, k)\big] + \mathbf{p}$) are both β-fair and γ-popular. Since DOS does not use knowledge of true labels Y, it can be applied directly at prediction time to a population of individuals. However, this also means that it can only be applied when the measure of fairness is independent of the true labels Y (for example, ensuring equality of positive rates).

***k*-Quantile Lottery Shift (*k*-QLS)** Another option for creating popular and fair classifiers is to directly minimize a loss function regularized by the distance of the fair-and-popular classifier from the fair classifier (distance is measured on predictions at training time), e.g. Program 12 for randomized classifiers. k-QLS-based postprocessing achieves this goal by partitioning scores $h_F(\mathbf{X})$ for a population $\mathbf{X}$ into k bins (based on quantiles). The goal is then to compute probabilities $p_\ell^{(g)}$ for each bin ℓ and group g, which minimize empirical risk and change to each agent's outcome, while achieving γ-popularity and β-fairness. This is done at training time. Then at prediction time, we take all agents in group g with scores in bin ℓ and run a lottery, where each agent is classified as 1 with probability $p_\ell^{(g)}$, and 0 otherwise. Since k-QLS is applied on the training dataset, it also allows us to use fairness metrics that depend on labels Y; for this reason k-QLS is not used in allocation, where Y is typically unknown.

k-QLS is motivated by works such as (Hardt, Price, and Srebro 2016; Pleiss et al. 2017; Kamiran, Karim, and Zhang 2012; Canetti et al. 2019; Lohia et al. 2019) which aim to postprocess a conventional model to achieve β-fairness by running an "inclusion/exclusion" lottery on groups of agents. However, k-QLS differs from these approaches: shifting all outcomes of a group, even in a randomized manner, is too granular to achieve γ-popularity, and thus we shift outcomes within k quantiles. In Section C.3 of the Supplement we demonstrate the poor performance of group level shifts compared the higher precision shifts of both the quantile shifts of k-QLS and the individual shifts of DOS.

Remark 0.3. *Achieving γ-popularity and β-fairness may be infeasible in general. However, for common efficacy metrics (e.g., PR, FPR, and TPR), doing so is always possible. Both DOS and k-QLS have a feasible solution for any level of γ-popularity and β-fairness, for both randomized and deterministic models.*

Postprocessing for Deterministic Models

When the conventional model f_C, and β-fair model f_F are deterministic, the optimization problems defined for both the DOS approach and the k-QLS approach can be efficiently solved for any $\mathcal{U}$ defined by an additive efficacy metric $\mathcal{M}$. In both cases, since model decisions are binary, post processing amounts to finding some set of agents negatively classified by f_C, which minimally impact loss while not violating fairness, when positively classified.

Theorem 0.4 (Informal)**.** *When classifiers produce deterministic outcomes and $\mathcal{U}$ is defined by an additive fairness metric, the optimization problems for both DOS and k-QLS can be solved in polynomial time.*

We defer the formal statement of this claim, and a full discussion of deterministic postprocessing, to the Supplement.

DOS for Randomized Classification

Next we investigate popularity as it relates to randomized classifiers. Recall that in the case of randomized classifiers DOS aims to minimally shift the expected outcomes of f_F on a population $(\mathbf{X}, G)$, with unknown true labels Y, to produce the γ-popular β-fair model, which we denote by f_P, where $\mathbb{E}\big[f_P(\mathbf{x}_i)\big] = \mathbb{E}\big[f_F(\mathbf{x}_i)\big] + p_i$, and $0 \leq \mathbb{E}\big[f_P(\mathbf{x}_i)\big] \leq 1$. Thus, DOS aims to solve the following optimization problem:

$$\min_{\mathbf{p}\in[-1,1]^n} \|\mathbf{p}\|_q \tag{5}$$

$$\text{s.t. } \mathcal{U}\big(\mathbb{E}\big[f_F(\mathbf{X})\big] + \mathbf{p},\ G\big) \leq \beta \tag{6}$$

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{I}\big[\mathbb{E}\big[f_C(\mathbf{x}_i)\big] \leq \mathbb{E}\big[f_F(\mathbf{x}_i)\big] + p_i\big] \geq \gamma \tag{7}$$

for $q \in \{1, 2, \infty\}$. A key challenge is that the popularity constraint (7) is discrete and non-convex, amounting to a combinatorial problem of identifying a subset of $\gamma|\mathbf{X}|$ individuals who prefer the f_P to its conventional counterpart f_C. Nevertheless, this problem can be solved in polynomial time.

Theorem 0.5. *Let f_C and f_F be respectively a conventional and β-fair randomized classifier. Let $\mathcal{U}$ be derived from an additive efficacy metric $\mathcal{M}$ which is independent of Y (e.g., PR). Then for $q \in \{1, 2, \infty\}$ Program 5 can be solved in time $\Theta(\gamma n T)$ (where $\Theta(T)$ is the time required to solve a linear program or semi-definite program, as appropriate) by Algorithm 1, which returns a γ-popular, β-fair model f_P.*

Algorithm 1: **(Randomized DOS)** Postprocessing technique for converting a β-fair model f_F into a γ-popular β-fair model f_P.

```
Input: population: $(\mathbf{X}, Y, G)$, $\beta$-fair model: $f_F$, conventional model: $f_C$, popularity: $\gamma$
Result: weights $\mathbf{p}$ s.t. $f_P = f_F + \mathbf{p}$ is $\gamma$-popular and $\beta$-fair
$G_g := \{i : g_i = g\}$ s.t. $\mathbb{E}\big[f_C(\mathbf{x}_i)\big] - \mathbb{E}\big[f_F(\mathbf{x}_i)\big] \leq \mathbb{E}\big[f_C(\mathbf{x}_{i+1})\big] - \mathbb{E}\big[f_F(\mathbf{x}_{i+1})\big]$
$m := \lceil \gamma n \rceil$
for $i = 1$ to $m$ do
    $S_i = \big\{\mathbb{E}\big[f_C(\mathbf{x}_j)\big] \leq \mathbb{E}\big[f_F(\mathbf{x}_j)\big] + p_j : j \in G_1[:i]\big\}$
        $\cup \big\{\mathbb{E}\big[f_C(\mathbf{x}_j)\big] \leq \mathbb{E}\big[f_F(\mathbf{x}_j)\big] + p_j : j \in G_0[:m-i]\big\}$
    build Program 5 and replace Constraint 7 with $S_i$
    $\mathbf{p}_i$ = solution to the modified program
end for
return $\mathbf{p}^* = \arg\min_i \|\mathbf{p}_i\|$
```

Proof Sketch. Recall that $\mathbb{E}[f(\mathbf{x})] = h(\mathbf{x})$, an agent i prefers f_P to f_C if $h_C(\mathbf{x}_i) \leq h_P(\mathbf{x}_i) = h_F(\mathbf{x}_i) + p_i$, and if this holds for at least $m = \gamma n$ agents then f_P is γ-popular. In the case of DOS postprocessing, if a *specific* set of m constraints is required to hold, rather than *any* m constraints, the problem is tractable as it is a linear program ($q = 1, \infty$) or semi-definite program ($q = 2$).

To order the set of possible constraints such that only a polynomial number must be examined, we make use of the following observations: for any two agents $i, j \in G_g$, 1.) since $\mathcal{U}$ is additive and independent of Y, unfairness is invariant under any change to p_i, p_j which preserves $p_i + p_j$, and 2.) if $h_C(\mathbf{x}_i) - h_F(\mathbf{x}_i) \geq h_C(\mathbf{x}_j) - h_F(\mathbf{x}_j)$ then $h_C(\mathbf{x}_i) \leq h_F(\mathbf{x}_i) + p_i$ iff $h_C(\mathbf{x}_j) \leq h_F(\mathbf{x}_j) + p_i$. Thus, for any solution $\mathbf{p}$ where $h_C(\mathbf{x}_i) \leq h_F(\mathbf{x}_i) + p_i$, but $h_C(\mathbf{x}_j) > h_F(\mathbf{x}_j) + p_j$, permuting p_i and p_j does not affect loss, fairness, or popularity, (when permutation is infeasible, shifting the maximum allowed weight from p_i to p_j is sufficient). Since the problem is invariant under such permutations, we need only consider imposing $h_C(\mathbf{x}_i) \leq h_F(\mathbf{x}_i) + p_i$ if $h_C(\mathbf{x}_j) \leq h_F(\mathbf{x}_j) + p_j$ is already imposed.

Thus, each G_g can be ordered such that for $i, j \in G_g$, if $j < i$ then $h_C(\mathbf{x}_j) - h_F(\mathbf{x}_j) \leq h_C(\mathbf{x}_i) - h_F(\mathbf{x}_i)$. Since the intragroup decisions are made trivial via this ordering, only the intergroup decisions remain. Since at least m popularity constraints need to hold, and there are m ways to select exactly m total constraints between the two groups while preserving the intragroup ordering, there are only m sets of constraints that need investigation. Each set corresponds to solving either a LP or SDP which takes time $\Theta(T)$ to solve. The specific running time of each program type is outlined in the Supplement. Thus the total running time of DOS is $\Theta(\gamma n T)$. □

DOS for Randomized Resource Allocation

Next we turn our attention to resource allocation, in which $k < n$ equally desirable resources are allocated to a population of size n. Recall that the randomized allocation scheme given by $I(\mathbf{X}, G)$ assigns resources to agents where $\mathbb{E}\big[I_i(\mathbf{X}, G)\big] \in [0, 1]$ gives the probability that agent i receives a resource with allocation performed over population $(\mathbf{X}, G)$. For notational convenience, we use $I(i) = \mathbb{E}\big[I_i(\mathbf{X}, G)\big]$ to represent the probability that agent i receives the resources and suppress the expectation and implicit dependence on the population $(\mathbf{X}, G)$.

Scarce resource allocation is particularly well suited for DOS as true labels (with respect to the allocation decision) are typically unknown. In this case, DOS postprocessing is given by,

$$\min_{\mathbf{p}\in[-1,1]^n} \|\mathbf{p}\|_q \tag{8}$$

$$\text{s.t. } \sum_{i=1}^{n} I_F(i) + p_i \leq k \tag{9}$$

$$\mathcal{U}\big(I_F + \mathbf{p},\ G\big) \leq \beta \tag{10}$$

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{I}\big[I_C(i) \leq I_F(i) + p_i\big] \geq \gamma \tag{11}$$

We now show that DOS in resource allocation settings remains tractable.

Theorem 0.6. *Let I_C and I_F be a conventional and β-fair allocation scheme, respectively, and $\mathcal{U}$ be derived from an additive efficacy metric $\mathcal{M}$ which is independent of Y (e.g., PR). Then for $q \in \{1, 2, \infty\}$ Program 8 can be solved in time $\Theta(\gamma n T)$ by Algorithm 1 which returns a γ-popular, β-fair allocation if one exists.*

Proof Sketch. In the case of scarce resources, agents can again be ordered in an identical fashion to the classification setting (Theorem 0.5). Note that for any solution $\mathbf{p}$ and any $i, j \in G_g$, the resource constraint $\sum_{i=1}^{n} I_F(i)+p_i \leq k$ is invariant to any change in p_i, p_j, which preserves p_i+p_j. Thus a similar argument to Theorem 0.5, with a few caveats relating to infeasible solutions, holds. Specifically, this yields γn programs (either LPs or SDPs), each of which is solvable in time $\Theta(T)$. Thus DOS post processing for resource allocation can be computed in time $\Theta(\gamma n T)$. □

k-QLS for Randomized Classification

Finally, we explore k-QLS postprocessing for randomized classifiers. k-QLS creates k intervals by the quantiles of $h_F(\mathbf{X})$, where k is chosen by the model designer. Specifically, let ρ_ℓ be the maximum score associated with quantile ℓ of $h_F(\mathbf{X})$. Each interval is given as $I_\ell = [\rho_{\ell-1}, \rho_\ell]$, with the understanding that $\rho_0 = 0$ and $\rho_k = 1$. On each interval I_ℓ, and for each group g, a parameter $p_\ell^{(g)}$ is learned. At prediction time, $\mathbb{E}\big[f_P(\mathbf{x}_i)\big] = p_\ell^{(g_i)}$ for i s.t. $h_F(\mathbf{x}_i) \in I_\ell$, .

Finding the optimal lottery probabilities can formulated as the following optimization problem:

$$\min_{\mathbf{p}\in[0,1]^{2k}} \mathcal{L}\big(f_P, \mathbf{X}, Y\big) + \lambda \|f_F(\mathbf{X}) - f_P(\mathbf{X})\|_q^q \tag{12}$$

$$\text{s.t. } \mathcal{U}\big(f_P, D\big) \leq \beta \tag{13}$$

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{I}\big[f_C(\mathbf{x}_i) \leq f_P(\mathbf{x}_i, g_i)\big] \geq \gamma, \tag{14}$$

where $\mathcal{L}$ is expected training error. As was the case for DOS postprocessing with randomized classifiers, the constraint that γ fraction of the population prefers f_P over f_C is discrete and non-convex. Indeed, unlike DOS, the k-QLS problem becomes strongly NP-hard.

Theorem 0.7. *Postprocessing to achieve γ-popularity and β-fairness with k-QLS (i.e., solving Program 12) is strongly NP-hard when models are randomized, and $\mathcal{U}$ is derived from an additive efficacy metric.*

We defer this proof to Section B.3 of the Supplement.

The intractability stems entirely from the model designer's ability to choose the number of quantiles k: if k is fixed, the problem can be solved in polynomial time as shown in the following theorem. In practice, we can fix k to be small, thus obtaining a tractable algorithm.

Theorem 0.8. *Let f_C and f_F be a conventional and a β-fair randomized classifier respectively. Let U be derived from an additive efficacy metric $\mathcal{M}$. Then for a fixed number of quantiles k, Program 12 for $q = \{1, 2, \infty\}$ can be solved in polynomial time, thus obtaining γ-popular β-fair decisions.*

Proof. As was the case for DOS applied to randomized classifiers, k-QLS applied to randomized classifiers is tractable if a specific set of $m = \gamma n$ agents is required to prefer f_P, rather than any m agents. When the number of intervals is constant it is straightforward to induce an ordering on agents which explores only a polynomial number of constraint sets. Specifically, let $G_{(g,\ell)} = \{i \in [n] : g_i = g \text{ and } h_F(\mathbf{x}_i) \in I_\ell\}$. Then agents in each G_g can be ordered by the magnitude of $p_\ell^{(g)}$ required such that they prefer f_P to f_C. Order G_g such that for $i, j \in G_g$ if $i < j$ then $h_C(\mathbf{x}_j) \leq h_C(\mathbf{x}_i)$, then if agent $i \in G_g$ prefers f_P to f_C, so does every $j \leq i$. There are $2k$ such sets, each containing at most n/k agents. Since the popularity over each G_g can be parameterized by the identity of the agent with the largest value of $h_C(\mathbf{x})$ who prefers f_P, there are no more than $(\gamma n)^k$ unique values under this parameterization, and thus no more than $(\gamma n)^k$ sets of constraints need be examined; each examination requires only polynomial time. □

Experiments

In this section we empirically investigate the relationship between popularity and fairness, and evaluate the efficacy of the proposed postprocessing algorithms. Each experiment is conducted on four data sets: 1) the **Recidivism** dataset, 2) the **Income** dataset, 3) the **Community Crime** dataset, and 4) the **Law School** dataset. In each dataset features can be continuous or categorical; each label is binary and defined such that 1 is always the more desirable outcome, e.g. in the Recidivism dataset $y = 1$ indicates *not* reoffending. A specific description of the label is given in the Supplement. Group membership is defined by race for Community Crime and Law School, and by gender for Recidivism and Income; either feature is assumed to be binary. All other sensitive features, such as age, are removed from the dataset. We consider three fair learning schemes: the **Reductions** algorithm (Agarwal et al. 2018), the **CalEqOdds** algorithm (Pleiss et al. 2017), the **KDE** algorithm (Cho, Hwang, and Suh 2020). Results for the latter two are provided in Section C of the supplement.

Popularity of Current Fair Learning Schemes: We begin by considering popularity of group-fair classifiers. The fractions of the overall population, and subgroup population, which prefer the fair classifier are shown in Figure 1, where fairness is achieved using the **Reductions** method.

Not surprisingly, we see that in all instances the disadvantaged group G_0 prefers f_F at far higher rates than G_1. With the exception of the **CalEqOdds** algorithm (which achieves fairness via group specific score shifts, resulting in far stronger group-level preference over classifiers), results for other methods are similar; these are provided in the supplement. Overall, randomized fair classifiers frequently have popularity of less than 50% . On the other hand, fair deterministic classifiers are relatively popular in most cases.

In either case, however, postprocessing can be used to further boost popularity of group-fair methods.

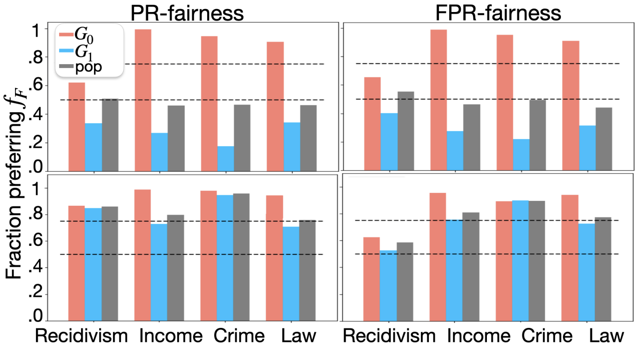

Figure 1: Fraction of each population or group preferring f_F over f_C for randomized classifiers (top) and deterministic classifiers (bottom), when f_F is learned via the Reductions algorithm.

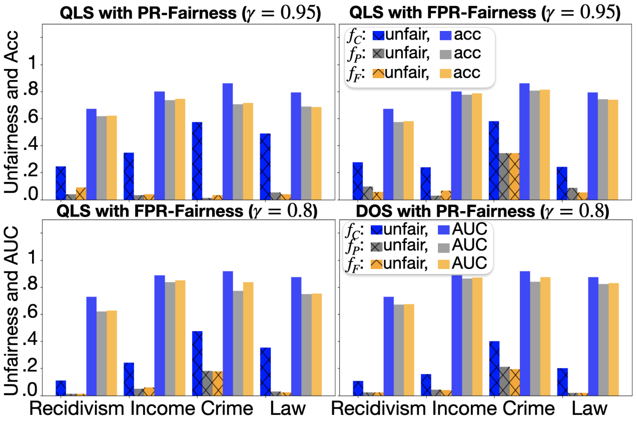

Figure 2: Model performance and unfairness on test data (3-fold average) for deterministic models with $\gamma = 0.95$ (top) and randomized models with $\gamma = 0.8$ (bottom). The conventional classifier f_C, fair classifier f_F (learned via reductions), and the fair popular classifier f_P (learned via our postprocessing technique), each using Logistic Regression.

Postprocessing for Fairness and Popularity: Next we examine the efficacy of our proposed postprocessing techniques DOS and k-QLS (k=10). When classifiers are deterministic, performance is measured using balanced accuracy (balanced w.r.t. Y). When classifiers are randomized, performance is measured using ROC-AUC, calculated over model scores (i.e., expected outcomes).

Remark 0.9. *Both k-QLS and DOS may require solving a large number of LPs or SDPs, which may be expensive. However, both methods can be efficiently implemented in practice by either solving the programs in parallel, trimming down the number of programs with heuristics, or replacing all programs with a single integer program. The latter being the most efficient, typically finishing in under 60 seconds. Further details on these methods, and exact running times, are provided in Section C.3 of the Supplement.*

Figure 2 shows that both k-QLS and DOS are able to achieve high levels of γ-popularity and β-fairness with little degradation in performance. In particular, deterministic classifiers (due to their higher natural popularity) are able to achieve greater levels of popularity compared to randomized models, with similar levels of degradation to performance. We observe similar results for other combinations of dataset, efficacy metric, and classifier type (Section C of the supplement).

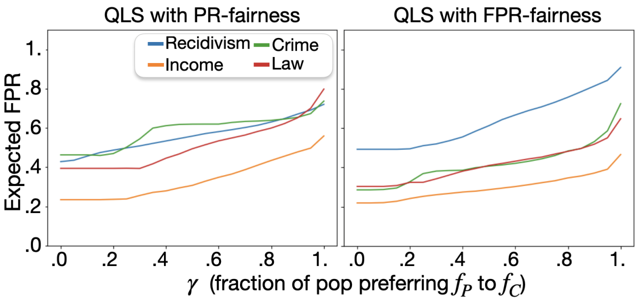

Figure 3: Expected False Positive Rate (FPR) of k-QLS, on randomized classifiers, as a function of γ.

Finally, we consider the extent to which popularity may skew model efficacy. In particular, as the popularity coefficient γ increases, a larger fraction of the population is guaranteed to have scores from f_P, which are at least as large as those from f_C. Since popularity constraints ensure that agents scores do not decrease, achieving higher levels of popularity (i.e., higher γ) also incentivize the resulting f_P to maintain false positive errors made by f_C. Thus one would expect FPR to increase with γ. This phenomenon is shown in Figure 3, which demonstrates that as γ increases, so does expected FPR. Although the expected FPRs vary between datasets and fairness definitions, the rate of increase is relatively similar across instances.

In Section C of the supplement, we further explore the tradeoffs between error, fairness, and popularity via the Pareto frontiers of these values. Similar to the classic results involving fairness and accuracy, we find that there is a fundamental tradeoff between model accuracy and popularity.

Conclusion

The deployment of group-fair classifiers, in place of conventional classifiers, may result large fractions of a population perceiving that they are made worse off by the change. We introduce the notion of popularity, which captures the fraction of agents preferring one classifier over another, and propose two postprocessing techniques (DOS and k-QLS) for achieving popularity while retaining good fairness properties. Both techniques provide efficient solutions for both deterministic and randomized classifiers. We note that while in practice postprocessing can achieve popularity and fairness with minimal degradation to model performance, requiring higher levels of popularity can actually entrench any false positive errors made by the conventional model. Consequently, application of the proposed techniques need to carefully analyze the tradeoffs not merely between popularity, group fairness, and overall accuracy, but also with specific measures of error, particularly the false positive rate.

Acknowledgments

This research was partially supported by the National Science Foundation (IIS-1905558, IIS-1903207, IIS-1939677, IIS-2127752, IIS-2127754, IIS-1908287 and IIS-1942336) and Amazon.

References

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

Angwin, J.; Larson, J.; Mattu, S.; and Kirchner, L. 2016. Machine bias. In *Ethics of Data and Analytics*, 254–264. Auerbach Publications.

Barocas, S.; Hardt, M.; and Narayanan, A. 2017. Fairness in machine learning. *Nips tutorial*, 1: 2.

Ben-Porat, O.; Sandomirskiy, F.; and Tennenholtz, M. 2019. Protecting the protected group: Circumventing harmful fairness. *arXiv preprint arXiv:1905.10546*.

Berger, J.; Osterloh, M.; Rost, K.; and Ehrmann, T. 2020. How to prevent leadership hubris? Comparing competitive selections, lotteries, and their combination. *The Leadership Quarterly*, 31(5): 101388.

Canetti, R.; Cohen, A.; Dikkala, N.; Ramnarayan, G.; Scheffler, S.; and Smith, A. 2019. From soft classifiers to hard decisions: How fair can we be? In *Proceedings of the conference on fairness, accountability, and transparency*, 309–318.

Cho, J.; Hwang, G.; and Suh, C. 2020. A fair classifier using kernel density estimation. *Advances in Neural Information Processing Systems*, 33: 15088–15099.

Chouldechova, A.; and Roth, A. 2018. The frontiers of fairness in machine learning. *arXiv preprint arXiv:1810.08810*.

Citron, D. K.; and Pasquale, F. 2014. The scored society: Due process for automated predictions. *Wash. L. Rev.*, 89: 1.

Corbett-Davies, S.; and Goel, S. 2018. The measure and mismeasure of fairness: A critical review of fair machine learning. *arXiv preprint arXiv:1808.00023*.

Cousins, C. 2021. An axiomatic theory of provably-fair welfare-centric machine learning. *Advances in Neural Information Processing Systems*, 34.

Dastin, J. 2018. Amazon scraps secret AI recruiting tool that showed bias against women. In *Ethics of Data and Analytics*, 296–299. Auerbach Publications.

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

Feldman, M.; Friedler, S. A.; Moeller, J.; Scheidegger, C.; and Venkatasubramanian, S. 2015. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 259–268.

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

Hong, L.; and Page, S. E. 2004. Groups of diverse problem solvers can outperform groups of high-ability problem solvers. *Proceedings of the National Academy of Sciences*, 101(46): 16385–16389.

Hu, L.; and Chen, Y. 2020. Fair classification and social welfare. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, 535–545.

Jang, T.; Shi, P.; and Wang, X. 2021. Group-Aware Threshold Adaptation for Fair Classification. *arXiv preprint arXiv:2111.04271*.

Kamiran, F.; Karim, A.; and Zhang, X. 2012. Decision theory for discrimination-aware classification. In *2012 IEEE 12th International Conference on Data Mining*, 924–929. IEEE.

Karlan, D.; and Zinman, J. 2010a. Expanding credit access: Using randomized supply decisions to estimate the impacts. *The Review of Financial Studies*, 23(1): 433–464.

Karlan, D.; and Zinman, J. 2010b. Expanding credit access: Using randomized supply decisions to estimate the impacts. *The Review of Financial Studies*, 23(1): 433–464.

Kasy, M.; and Abebe, R. 2021. Fairness, equality, and power in algorithmic decision-making. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 576–586.

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

Koenecke, A.; Nam, A.; Lake, E.; Nudell, J.; Quartey, M.; Mengesha, Z.; Toups, C.; Rickford, J. R.; Jurafsky, D.; and Goel, S. 2020. Racial disparities in automated speech recognition. *Proceedings of the National Academy of Sciences*, 117(14): 7684–7689.

Kube, A.; Das, S.; and Fowler, P. J. 2019. Allocating interventions based on predicted outcomes: A case study on homelessness services. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 622–629.

Kusner, M. J.; Loftus, J.; Russell, C.; and Silva, R. 2017. Counterfactual Fairness. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Lee, N. T. 2018. Detecting racial bias in algorithms and machine learning. *Journal of Information, Communication and Ethics in Society*.

Li, B.; and Vorobeychik, Y. 2015. Scalable optimization of randomized operational decisions in adversarial classification settings. In *Artificial Intelligence and Statistics*, 599–607.

Liu, L. T.; Dean, S.; Rolf, E.; Simchowitz, M.; and Hardt, M. 2018. Delayed impact of fair machine learning. In *International Conference on Machine Learning*, 3150–3158. PMLR.

Lohia, P. K.; Ramamurthy, K. N.; Bhide, M.; Saha, D.; Varshney, K. R.; and Puri, R. 2019. Bias mitigation post-processing for individual and group fairness. In *Icassp 2019-*

2019 ieee international conference on acoustics, speech and signal processing (icassp), 2847–2851. IEEE.

Mehrabi, N.; Morstatter, F.; Saxena, N.; Lerman, K.; and Galstyan, A. 2021. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6): 1–35.

Pinot, R.; Meunier, L.; Araujo, A.; Kashima, H.; Yger, F.; Gouy-Pailler, C.; and Atif, J. 2019. Theoretical evidence for adversarial robustness through randomization. In *Neural Information Processing Systems*.

Pleiss, G.; Raghavan, M.; Wu, F.; Kleinberg, J.; and Weinberger, K. Q. 2017. On fairness and calibration. *Advances in neural information processing systems*, 30.

Salman, H.; Li, J.; Razenshteyn, I.; Zhang, P.; Zhang, H.; Bubeck, S.; and Yang, G. 2019. Provably robust deep learning via adversarially trained smoothed classifiers. In *Neural Information Processing Systems*, volume 32.

Tassier, T.; and Menczer, F. 2008. Social network structure, segregation, and equality in a labor market with referral hiring. *Journal of Economic Behavior & Organization*, 66(3-4): 514–528.

Ustun, B.; Liu, Y.; and Parkes, D. 2019. Fairness without harm: Decoupled classifiers with preference guarantees. In *International Conference on Machine Learning*, 6373–6382. PMLR.

Vorobeychik, Y.; and Li, B. 2014. Optimal randomized classification in adversarial settings. In *International Conference on Autonomous Agents and Multiagent Systems*, 485–492.

Xinying Chen, V.; and Hooker, J. 2023. A guide to formulating fairness in an optimization model. *Annals of Operations Research*, 1–39.

Zhang, X.; Tu, R.; Liu, Y.; Liu, M.; Kjellstrom, H.; Zhang, K.; and Zhang, C. 2020. How do fair decisions fare in long-term qualification? *Advances in Neural Information Processing Systems*, 33: 18457–18469.