Adaptive Sampling for Minimax Fair Classification

Anonymous Author(s)

Affiliation

Address

email

Abstract

Machine learning models trained on uncurated datasets can often end up adversely affecting inputs belonging to underrepresented groups. To address this issue, we consider the problem of adaptively constructing training sets which allow us to learn classifiers that are fair in a *minimax* sense. We first propose an adaptive sampling algorithm based on the principle of *optimism*, and derive theoretical bounds on its performance. We also propose heuristic extensions of this algorithm suitable for application to large scale, practical problems. Next, by deriving algorithm independent lower-bounds for a specific class of problems, we show that the performance achieved by our adaptive scheme cannot be improved in general. We then validate the benefits of adaptively constructing training sets via experiments on synthetic tasks with logistic regression classifiers, as well as on several real-world tasks using convolutional neural networks (CNNs).

1 Introduction

Machine learning (ML) models are increasingly being applied for automating the decision-making process in several sensitive applications, such as loan approval and employment screening. However, recent work has demonstrated that discriminatory behaviour might get encoded in the model at various stages of the ML pipeline, such as data collection, labelling, feature selection, and training, and as a result, adversely impact members of some protected groups in rather subtle ways (Barocas and Selbst, 2016). This is why ML researchers have started to introduce a large number of *fairness measures* to include the notion of fairness in the design of their algorithms. Some of the important measures of fairness include demographic parity (Zemel et al., 2013), equal odds and opportunity (Hardt et al., 2016; Woodworth et al., 2017), individual fairness (Dwork et al., 2012), and minimax fairness (Feldman et al., 2015). The *minimax* fairness is particularly important in scenarios in which it is necessary to be as close as possible to equality without introducing unnecessary harm (Ustun et al., 2019). These scenarios are common in areas such as healthcare and predicting domestic violence. A measure that has been explored to achieve this goal is predictive risk disparity (Feldman et al., 2015; Chen et al., 2018; Ustun et al., 2019). Instead of using the common approach of putting constraints on the norm of discrimination gaps, Martinez et al. (2020) has recently introduced the notion of *minimax Pareto fairness*. These are minimax classifiers that are on the Pareto frontier of prediction risk, i.e., no decrease in the predictive risk of one group is possible without increasing the risk of another one.

In this paper, we are primarily interested in the notion of minimax fairness in terms of the predictive risk. However, instead of studying the training phase of the ML pipeline, our focus is on the data-collection stage, motivated by Jo and Gebru (2019) and Holstein et al. (2018). In particular, we study the following question: *given a finite sampling budget, how should a learner construct a training set consisting of elements from different protected groups in appropriate proportions to ensure that a classifier trained on this dataset achieves minimax fairness?*

Our work is motivated by the following scenario: suppose we have to learn a ML model for performing a task (e.g., loan approval) for inputs belonging to different groups based on protected attributes, such as race or gender. Depending upon the distribution of input-label pairs from different groups, the given task may be statistically harder for some groups. Our goal is to ensure that the eventual ML

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

model has optimal predictive accuracy for the worst-off group. We show that, under certain technical conditions, this results in a model with comparable predictive accuracy over all groups. One approach for this would be to train separate classifiers for each group. However, this is often not possible as the group-membership information may not be available at the deployment time, or it may be forbidden by law to explicitly use the protected characteristics as an input in the prediction process (Lipton et al., 2017, § 1). To ensure having a higher proportion of samples from the *harder* groups without knowing the identity of the hard or easy groups apriori, we consider this problem in an *active setting*, where a learner has to incrementally construct a training set by drawing samples one at a time (or a batch at a time) from different groups. Towards this goal, we propose and analyze an adaptive sampling scheme based on the principle of *optimism*, used in bandits literature (e.g., Auer et al. 2002), that detects the *harder* groups and populates the training set with more samples from them in an adaptive manner. We also wish to note that bias in ML has multi-faceted origins and that our work here addresses dataset construction and cannot account for bias introduced by model selection, the underlying data distribution, or other sources as discussed in Hooker (2021), Suresh and Guttag (2019). We also endeavor to ensure *minimax* fairness, but in some contexts another notion of fairness, such as those mentioned above, may be more appropriate or equitable. In general, application of our algorithm is not a guarantee that the resulting model is wholly without bias.

Our *main contributions* are: **1)** We first propose an optimistic adaptive sampling strategy, $\mathcal{A}_{\texttt{opt}}$, for training set construction in Sec. 3. This strategy is suited to smaller problems and admits theoretical guarantees. We then introduce a heuristic variant of $\mathcal{A}_{\texttt{opt}}$ in Sec. 4 that is more suitable to practical problems involving CNNs. **2)** We obtain upper bounds on the convergence rate of $\mathcal{A}_{\texttt{opt}}$, and show its minimax near-optimality by constructing a matching lower bound in Sec. 3.1. **3)** Finally, we demonstrate the benefits of our algorithm with empirical results on several synthetic and real-world datasets in Sec. 5.

Related Work. The closest work to ours is by Abernethy et al. (2020), where they propose an ϵ-greedy adaptive sampling strategy. They present some preliminary theoretical analysis and also empirically demonstrate the benefits of their strategy over some baselines. We describe their results in more details in App. C.1, and employ the tools we develop to analyze our algorithm and perform a thorough analysis of the excess risk of their ϵ-greedy strategy. In another related work, Anahideh et al. (2020) propose a fair adaptive sampling strategy that selects points based on a linear combination of model accuracy and fairness measures, and empirically study its performance. These results, however, do not obtain convergence rates of the excess risk of their respective methods.

The above results study the problem of fair classification in an *active* setting and target the *data-collection* stage of the ML pipeline. There are also works that take a *passive* approach to this problem and focus of the *training* phase of the pipeline. Agarwal et al. (2018) design a scheme for learning fair binary classifiers with fairness metrics that can be written as linear constraints involving certain conditional moments. This class of fairness metrics, however, do not contain the minimax fairness measure. Diana et al. (2021) propose a method for constructing (randomized) classifiers for minimax fairness w.r.t. the empirical loss calculated on the given training set. Similarly, Martinez et al. (2020) derive an algorithm for learning a Pareto optimal minimax fair classifier, under the assumption that the learner has access to the true expected loss functions. Thus, the theoretical guarantees in Diana et al. (2021) and Martinez et al. (2020) hold under the assumption of large training sets. Our work, in contrast, constructs the training set incrementally (active setting) from scratch while carefully taking into account the effects of the finite sample size. Finally, we note that the data-collection strategies proposed in our paper can, in principle, be combined with the training methods presented in Martinez et al. (2020) and Diana et al. (2021) to further guarantee the (minimax) fairness of the resulting classifier. We leave the investigation of this approach for future work.

Besides data-collection and training, there have been studies in the fair ML literature on other aspects of the ML pipeline, such as pre-processing (Celis et al., 2020), learning feature representations (Zemel et al., 2013), post-processing (Hardt et al., 2016), and model documentation (Mitchell et al., 2018). We refer the readers to a recent survey by Caton and Haas (2020) for more detailed description of these results.

2 Problem Formulation

Consider a classification problem with the feature (input) space $\mathcal{X}$, label set $\mathcal{Y}$, and a protected attribute set $\mathcal{Z} = \{z_1, \ldots, z_m\}$. For any $z \in \mathcal{Z}$, we use $P_z(x, y)$ as a shorthand for $P_{XY|Z=z}\left(X = x, Y = y \mid Z = z\right)$ to denote the feature-label joint distribution given that the pro-

tected feature value is $Z = z$. We also use $\mathbb{E}_z[\cdot]$ as a shorthand for the expectation w.r.t. P_z. A (non-randomized) classifier is a mapping $f : \mathcal{X} \mapsto \mathcal{Y}$, which assigns a label $y \in \mathcal{Y}$ to every feature $x \in \mathcal{X}$. For a family of classifiers $\mathcal{F} \subset \mathcal{Y}^{\mathcal{X}}$, a loss function $\ell : \mathcal{F} \times \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}$, and a mixture distribution over the protected attributes $\pi \in \Delta_m$, we define the π-*optimal classifier* f_π as

$$f_\pi \in \operatorname*{arg\,min}_{f \in \mathcal{F}} \; \mathbb{E}_\pi \left[\ell(f, X, Y)\right] := \sum_{z \in \mathcal{Z}} \pi(z) \; \mathbb{E}_z \left[\ell(f, X, Y)\right]. \tag{1}$$

When π lies on the corners of the simplex Δ_m, i.e., $\pi(z) = 1$ for a $z \in \mathcal{Z}$, we use the notation f_z instead of f_π. For π in the interior of Δ_m, it is clear that the risk of f_π on group z, i.e., $\mathbb{E}_z \left[\ell(f_\pi, X, Y)\right]$, must be larger than the best possible classification loss for $P_{XY|Z=z}$. In this paper, our goal is to develop an *active sampling scheme* to find the fair mixture distribution π^* in a *minimax* sense (*minimizing the maximum risk among the groups*), i.e.,

$$\pi^* \in \operatorname*{arg\,min}_{\pi \in \Delta_m} \; \max_{z \in \mathcal{Z}} \; L(z, f_\pi) := \mathbb{E}_z \left[\ell(f_\pi, X, Y)\right]. \tag{2}$$

The active sampling problem that we study in this paper can be formally defined as follows:

Problem 1. *Suppose $\mathcal{F}$ denotes a family of classifiers, ℓ a loss function, n is a sampling budget, and $\mathcal{O} : \mathcal{Z} \mapsto \mathcal{X} \times \mathcal{Y}$ an oracle that maps any attribute $z \in \mathcal{Z}$ to a feature-label pair $(X, Y) \sim P_{XY|Z=z}$. The learner designs an adaptive sampling scheme $\mathcal{A}$ that comprises a sequence of mappings $(A_t)_{t=1}^n$ with $A_t : (\mathcal{X} \times \mathcal{Y})^{t-1} \mapsto \mathcal{Z}$ to adaptively query $\mathcal{O}$ and construct a dataset of size n. Let $\pi_n \in \Delta_m$ denote the resulting empirical mixture distribution over $\mathcal{Z}$, where $\pi_n(z) = N_{z,n}/n$ and $N_{z,n}$ is the number of times that $\mathcal{A}$ samples from P_z in n rounds. Then, the quality of the resulting dataset is measured by the excess risk, or sub-optimality, of the π_n-optimal classifier f_{π_n}, i.e.,*

$$\mathcal{R}_n(\mathcal{A}) := \max_{z \in \mathcal{Z}} \; L\left(z, f_{\pi_n}\right) \; - \; \max_{z \in \mathcal{Z}} \; L(z, f_{\pi^*}) \;, \tag{3}$$

where π^* is the fair mixture distribution defined by (2)*. Hence, the goal of the learner is to design a strategy $\mathcal{A}$ which has a small excess risk $\mathcal{R}_n(\mathcal{A})$.*

Informally, the algorithm should adaptively identify the harder groups z and dedicate a larger portion of the overall budget n to sample from their distributions (see Sec. 2.2 for an illustrative example).

2.1 Properties of the Fair Mixture

As discussed above, our goal is to derive an active sampling scheme that allocates the overall budget n over the attributes $z \in \mathcal{Z}$ in a similar manner as the *unknown* fair mixture π^*, defined by (2). Thus, it is important to better understand the properties of π^* and the π-optimal classifiers f_π, defined by (1). We state three properties of f_π and π^* in this section. We refer the readers to Martinez et al. (2020) for the definitions of Pareto front and convexity discussed in this section.

Property 1. As discussed in Martinez et al. (2020, Sec. 4), any f_π that solves (1) for a π with $\pi(z) > 0, \; \forall z \in \mathcal{Z}$, belongs to the Pareto front $\mathcal{P}_{\mathcal{Z},\mathcal{F}}$ of the risk functions $\{L(z, f)\}_{z \in \mathcal{Z}}, \; \forall f \in \mathcal{F}$.

Property 2. We prove in Proposition 1 (see App. A for the proof) that under the following two assumptions on the conditional distributions $\{P_z\}_{z \in \mathcal{Z}}$, function class $\mathcal{F}$, and loss function ℓ, there exists a unique fair mixture π^*, whose classifier f_{π^*} has equal risk over all attributes $z \in \mathcal{Z}$.

Assumption 1. *The mapping $\pi \mapsto L(z, f_\pi)$ is continuous for all attributes $z \in \mathcal{Z}$. Furthermore, if $\pi, \nu \in \Delta_m$ are such that $\pi(z) > \nu(z)$ for an attribute $z \in \mathcal{Z}$, then $L(z, f_\pi) < L(z, f_\nu)$.*

The above assumption indicates that increasing the weight of an attribute z in the mixture distribution must lead to an increase in the performance of the resulting classifier on the distribution P_z.

Assumption 2. *For any two distinct attributes $z, z' \in \mathcal{Z}$, we must have $L(z, f_z^*) < L(z', f_z^*)$, for any $f_z^* \in \arg\min_{f \in \mathcal{F}} \; L(z, f)$.*

This assumption requires that any optimal classifier corresponding to distribution P_z to have higher risk w.r.t. the distribution $P_{z'}$ of any other attribute $z' \in \mathcal{Z}$. Note that Assumption 2 may not always hold, for example, if one attribute is *significantly* easier to classify than another.

Proposition 1. *Let Assumptions 1 and 2 hold for the conditional distributions $\{P_z\}_{z \in \mathcal{Z}}$, the loss function ℓ, and the function class $\mathcal{F}$. Then, there exists a unique $\pi^* \in \Delta_m$ that achieves the optimal value for problem* (2) *and satisfies $L(z_1, f_{\pi^*}) = L(z_2, f_{\pi^*}) = \cdots = L(z_m, f_{\pi^*}) := M^*$.*

In other words, Assumptions 1 and 2 define a regime where there exists a fair mixture π^*, whose classifier f_{π^*} achieves complete parity in the classification performance across all attributes. Moreover, Property 1 indicates that f_{π^*} also belongs to the Pareto front $\mathcal{P}_{\mathcal{Z},\mathcal{F}}$. Therefore, under these two assumptions, the equal risk classifier not only belongs to the Pareto front, but it is also the minimax Pareto fair classifier (Lemma 3.1 in Martinez et al. 2020). Note that, in general, without Assumption 1, the classifier that attains equality of risk might have worse performance on all attributes than the minimax Pareto fair classifier (Lemma 3.2 in Martinez et al. 2020).

Property 3. We may show that when both the function class $\mathcal{F}$ and risk functions $\{L(z,\cdot)\}_{z\in\mathcal{Z}}$ are *convex*, f_{π^*} is a minimax Pareto fair classifier. As originally derived by Geoffrion (1968) and then restated by Martinez et al. (2020, Thm. 4.1), under these convexity assumptions, the Pareto front $\mathcal{P}_{\mathcal{Z},\mathcal{F}}$ is convex and any classifier on $\mathcal{P}_{\mathcal{Z},\mathcal{F}}$ is a solution to (1). This together with Property 1 indicates that the classifier f_{π^*} corresponding to the fair (minimax optimal) mixture π^* is on the Pareto front, and additionally, is a minimax Pareto fair classifier.

2.2 Synthetic Models

It is illustrative to consider a pair of classes of synthetic models, the first of which we use as a running example throughout the paper. The second scenario is used to explore the implications of the case where Assumption 2 does not hold.

Definition 1 (`SyntheticModel1`)**.** Set $\mathcal{X} = \mathbb{R}^2$, $\mathcal{Y} = \{0,1\}$, and $\mathcal{Z} = \{u,v\}$. Each instance of our synthetic model is defined by the set of distributions $P_{XY|Z}$ that satisfy the following: $P_{Y|Z}(Y=1|Z=z) = 0.5$ for both $z \in \mathcal{Z}$, and $P_{X|Y=y,Z=z} = \mathcal{N}(\mu_{yz}, I_2)$ for all $(y,z) \in \mathcal{Y}\times\mathcal{Z}$. Thus, each instance of this model can be represented by a mean vector $\boldsymbol{\mu} = (\mu_{yz} : y \in \mathcal{Y},\ z \in \mathcal{Z})$.

The idea behind this class of models is that the 'hardness' of the classification problem for a value of $z \in \mathcal{Z}$ depends on the distance between μ_{0z} and μ_{1z}. The more separated these two mean vectors are, the easier it is to distinguish the labels. Thus, depending on this distance, it is expected that different protected attributes may require different fractions of samples in the training set to achieve comparable test accuracy. To illustrate this, we consider two instances of the `SyntheticModel1`: **I.** $\mu_{0u} = (-2,2)$, $\mu_{1u} = (2,-2)$, $\mu_{0v} = (-1,-1)$, and $\mu_{1v} = (1,1)$, and **II.** $\mu_{0u} = (-1.5,1.5)$, $\mu_{1u} = (1.5,-1.5)$, $\mu_{0v} = (-2,-2)$, and $\mu_{1v} = (2,2)$.

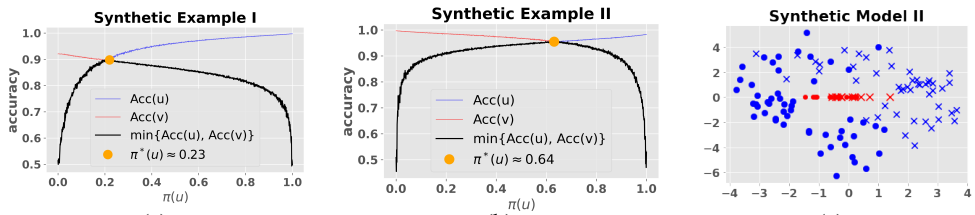
For both model instances, we trained a logistic regression classifier over training sets with 1001 equally spaced values of $\pi(u)$ in $[0,1]$. Fig. 1 shows the test accuracy of the learned classifier for both attributes (blue and red curves) as well as their minimum (black curve) for different values of $\pi(u)$. Since the pair (μ_{0u},μ_{1u}) is better separated than (μ_{0v},μ_{1v}) in the first instance, it is easier to classify inputs with $Z=u$ than those with $Z=v$, and thus, it requires fewer training samples from $P_{z=u}$ than $P_{z=v}$ to achieve the same accuracy. This is reflected in Fig. 1 *(left)* that shows the best (min-max) performance is achieved at $\pi(u) \approx 0.23$. An opposite trend is observed in the second instance (*right* plot in Fig. 1), where the mean vectors corresponding to the attribute $Z=v$ are better separated, and hence, require fewer training samples to achieve the same accuracy.

An instance of `SyntheticModel2` is illustrated in Fig. 1c) and specified in detail in App. E. Here $\mathcal{Y} = \{0,1\}$ is displayed with circles and crosses, respectively, in the figure, and $\mathcal{Z} = \{u,v\}$ is shown in red and blue, respectively. This data is designed such that for $Z=u$, where $P_{X|Y=y,Z=u}$ is uniform for both y and a large proportion of examples from both labels overlap, a linear classifier cannot attain high accuracy and does not benefit from training on many additional samples. In contrast, both $P_{X|Y=y,Z=v}$ are mixtures of Gaussians designed such that a linear classifier can attain high accuracy with few samples, but then continue to improve its accuracy with many samples from sparser parts of the distributions. We will use this data to explore heuristic extensions of our algorithm which address the case where Assumption 2 does not hold.

3 Optimistic Adaptive Sampling Strategy

In this section, we present our *optimistic adaptive sampling* algorithm $\mathcal{A}_{\texttt{opt}}$ for Problem 1. App. 1 contains a pseudo-code for $\mathcal{A}_{\texttt{opt}}$. We also report a more detailed and formal pseudo-code for $\mathcal{A}_{\texttt{opt}}$ in Algorithm 2 in App. B.1. Algorithm 1 proceeds in the following phases:

Phase 1 In the initial phase $t \le m = |\mathcal{Z}|$, we draw two independent samples from $P_z = P_{XY|Z=z}$, for each attribute $z \in \mathcal{Z}$, and add one to the training dataset $\mathcal{D}_t$ and the other one to $\mathcal{D}_z$. We use $\mathcal{D}_t$ to learn a common (over all $z \in \mathcal{Z}$) classifier $\hat{f}_t$ via *empirical risk minimization* (ERM). The independent datasets $\{\mathcal{D}_{z_i}\}_{i=1}^m$ are used for estimating the performance of $\hat{f}_t$ for each $z \in \mathcal{Z}$.



(a) (b) (c)

Figure 1: (a) and (b) show show the variation of the minimum prediction accuracy over the two attributes $\mathcal{Z} = \{u, v\}$ with $\pi(u)$ for Instance I and Instance II, respectively, of the `SyntheticModel1`. (c) shows an instance of `SyntheticModel2`.

Phase 2 In each round $t > m$, we choose an attribute z_t according to the following **selection rule**: *if* there exists an attribute z whose number of samples in $\mathcal{D}_t$, denoted by $N_{z_t,t}$, is fewer than t^ξ for some input $\xi \in (0, 1)$ (Line 5), we set it as z_t (Line 6), *else* we set z_t as the attribute which has the largest upper confidence bound (UCB) (described below) for the risk of the classifier f_{π_t} (Line 8). Here π_t denote the empirical mixture distribution (over the attributes z) of $\mathcal{D}_t$.

Phase 3 When the attribute z_t is selected in *Phase 2*, the algorithm draws a pair of independent samples from P_{z_t}, adds one to $\mathcal{D}_t$ and the other to $\mathcal{D}_{z_t}$, and updates $N_{z_t,t}$, π_t, and the uniform deviation bound $e_{z_t}(N_{z_t,t})$ (described below) (Lines 10 to 12). The updated dataset $\mathcal{D}_t$ is then used to learn a new candidate classifier $\hat{f}_t$ (Line 13). *Phases 2 and 3* are repeated until the sampling budget is exhausted, i.e., $t = n/2$ (note that we sample twice at each round).

Algorithm 1: Optimistic Adaptive Sampling for Minimax Fair Classification.

Input: n (budget), $\mathcal{F}$ (function class), ℓ (loss function), ξ (forced exploration term)

1 **Initialize:** $\mathcal{D}_1 = \emptyset$; $\{\mathcal{D}_{z_i}\}_{i=1}^m = \emptyset$; $\{N_{z_i,1}\}_{i=1}^m = 0$;
2 **for** $t = 1, \ldots, n/2$ **do**
3 **if** $t \leq m$ **then**
4 $z_t = z_t$;
5 **else if** $\min_{z\in\mathcal{Z}} N_{z,t} < t^\xi$ **then**
6 $z_t \in \arg\min_{z\in\mathcal{Z}} N_{z,t}$;
7 **else**
8 $z_t = \arg\max_{z\in\mathcal{Z}} U_t(z, \hat{f}_t)$; `// see Eq. 4`
9 **end**
10 Draw two independent samples from P_{z_t};
11 Add the first one to $\mathcal{D}_t$ and the second one to $\mathcal{D}_{z_t}$;
12 Update $N_{z_t,t}$; π_t; $e_{z_t}(N_{z_t,t})$;
13 Learn classifier $\hat{f}_t$ by ERM using $\mathcal{D}_t$;
14 **end**

Output: $\pi_{n/2}$ and $\hat{f}_{n/2}$

Calculating UCB. To construct the UCB, we assume that $\mathcal{F}$ and ℓ satisfy a uniform deviation property. More specifically, for $\delta \in [0, 1]$, $\xi \in (1/2, 1)$ and for every $z \in \mathcal{Z}$, we assume that there exists a monotonically non-increasing sequence $\{e_z(N, \mathcal{F}, \delta) : N \geq 1\}$ with $\lim_{N\to\infty} 1/\left(N^\xi e_z(N, \mathcal{F}, \delta)\right) = 0$, such that the following event holds with probability at least $1 - \delta/2$: $\Omega = \bigcap_{z\in\mathcal{Z}} \bigcap_{N=1}^\infty \left\{\sup_{f\in\mathcal{F}} |\widehat{L}_N(z, f) - L(z, f)| \leq e_z(N, \mathcal{F}, \delta)\right\}$. Here $\widehat{L}_N(z, f) := \frac{1}{N}\sum_{i=1}^N \ell\left(f, X_z^{(i)}, Y_z^{(i)}\right)$, $\forall z \in \mathcal{Z}$, with $\left(X_z^{(i)}, Y_z^{(i)}\right)_{i=1}^N$ being an i.i.d. sequence of input-label pairs drawn from P_z. Such uniform deviation conditions are satisfied by various $\mathcal{F}$ and ℓ pairs: for example if $\mathcal{F}$ has finite VC-dimension d_{VC}, then an appropriate choice of $e_z(N, \mathcal{F}, \delta)$ is $2\sqrt{\left(8d_{VC}\log(en/d_{VC}) + 2\log(8n^2\pi^2/6\delta)\right)/n}$. In the sequel, we suppress the δ and $\mathcal{F}$ dependence of $e_z(N, \mathcal{F}, \delta)$ and use $e_z(N)$. Following the above discussion, we construct the UCB for the risk function $L(z, f_{\pi_t})$, defined by (2), by the following expression:

$$U_t(z, \hat{f}_t) := \frac{1}{|\mathcal{D}_z|} \sum_{(x,y)\in\mathcal{D}_z} \ell(\hat{f}_t, x, y) \; + \; e_z\left(N_{z,t}\right). \tag{4}$$

3.1 Theoretical Analysis

In this section, we derive an upper-bound on the excess risk $\mathcal{R}_n(\mathcal{A}_{\texttt{opt}})$ of Algorithm 1. We also show that the performance achieved this algorithm cannot in general be improved, by obtaining an algorithm independent lower-bound on the excess risk for a particular class of problems.

Upper Bound. We begin by obtaining an upper bound on the convergence rate of the excess risk of $\mathcal{A}_{\texttt{opt}}$. The detailed proof of this result is in App. B.2.

Theorem 1. *Let Assumptions 1-3 (Assumption 3 stated in App. B.2) hold and define* $\pi_{\min} := \min_{z\in\mathcal{Z}} \pi^*(z)$. *Fix any* A *such that* $\pi_{\min}/2 \leq A < \pi_{\min}$. *Suppose the query budget* n *is sufficiently*

large, as defined in Eq. 18 in App. B.2. Then, with probability $1-\delta$, the excess risk of Algorithm 1 can be upper-bounded as

$$\mathcal{R}_n(\mathcal{A}_{\texttt{opt}}) = \max_{z\in\mathcal{Z}} \; L(z, f_{\pi_n}) - M^* = \mathcal{O}\Big(\frac{|\mathcal{Z}|}{\pi_{\min}} \; \max_{z\in\mathcal{Z}} \; e_z(N_A)\Big), \tag{5}$$

where $M^* = L(z, f_{\pi^*}),\ \forall z \in \mathcal{Z}$ (see Proposition 1) and $N_A = n\pi_{\min}^2/(2\pi_{\min} - A)$.

Remark 1 (Analysis of ϵ-`greedy`)**.** *Abernethy et al. (2020) only analyze the greedy version (i.e., $\epsilon = 0$) of their ϵ-`greedy` sampling strategy with $|\mathcal{Z}| = 2$, and show that at time n, either the excess risk is of $\mathcal{O}\big(\max_{z\in\mathcal{Z}} \sqrt{2d_{VC}(\ell\circ\mathcal{F})\log(2/\delta)/N_{z,n}}\big)$, or the algorithm draws a sample from the attribute with the largest loss. However, due to the greedy nature of the algorithm they analyzed, there are no guarantees that $N_{z,n} = \Omega(n)$, and thus, in the worst case the above excess risk bound is $\mathcal{O}(1)$. We show in App. C how the techniques we developed for the analysis of Algorithm 1 can be suitably employed to study their ϵ-`greedy` strategy. In particular, we obtain sufficient conditions on ϵ under which the excess risk of ϵ-`greedy` converges to zero, and the rate at which this convergence occurs.*

Lower Bound. Let $\mathcal{Q} = (\boldsymbol{\mu}, \mathcal{F}, \ell_{01})$ denote the class of problems where $\boldsymbol{\mu} \in \mathcal{M}$ is an instance of the `SyntheticModelI` described in Section 2.2, $\mathcal{F}$ is the class of linear classifiers in two dimensions, and ℓ_{01} is the $0-1$ loss. For this function class $\mathcal{F}$, $e_z(N) = \mathcal{O}(\sqrt{\log(N)/N})$ for $z \in \mathcal{Z} = \{u, v\}$, which implies that the excess risk achieved by both $\mathcal{A}_{\texttt{opt}}$ and ϵ-`greedy` strategies is of $\mathcal{O}\left(\sqrt{\log(n)/n}\right)$. We prove in App. D that this convergence rate (in terms of n) cannot in general be improved by showing that $\max_{Q\in\mathcal{Q}} \; \mathbb{E}_Q\left[\mathcal{R}_n\left(\mathcal{A}\right)\right] = \Omega\left(1/\sqrt{n}\right)$.

4 Heuristic Extensions

In practical applications of our optimistic adaptive sampling strategy we run into two primary challenges. As described in Sec. 3, $\mathcal{A}_{\texttt{opt}}$ requires re-training the classifier in every iteration. While this can be implemented in small problems, it becomes infeasible for problems involving large models, such as CNNs. Also, as noted in Sec. 2.1, for data where one attribute is significantly easier than the other, Assumption 2 may not hold. In this case it may not be beneficial to continue to sample from the attribute with the largest loss. To address these issues we present a heuristic variant of Algorithm 1.

For the first challenge we make the following modifications to Algorithm 1. **1)** We expand Phase 1 to n_0 rounds, **2)** at each subsequent round we draw two batches of size b_0 from the chosen attribute, **3)** instead of re-training from scratch each round, we update the previous model, $\hat{f}_{t-1}$, and **4)** instead of training to convergence, we perform one gradient step over the entire accumulated training set, $\mathcal{D}_t$.

To address the second challenge we modify the UCB by adding a term based on the Mann-Kendall statistic (Mann, 1945) (Kendall, 1948) which, for time series data $(X_i)_{i=1}^n$, is given by $S = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \operatorname{sgn}\left(X_j - X_i\right)$. This is designed to identify monotonic upwards or downward trends in time series. We calculate the statistic for each attribute, and denote them S_{z_t}, for the accuracy of the classifier, $\hat{f}_t$, on the attribute's validation set, $\mathcal{D}_{z_t}$, over time. Intuitively, this is tracking whether training on additional samples from an attribute is improving the classifier's accuracy on that attribute. We incorporate the statistic into the UCB as follows:

$$\widetilde{U}_t(z, \hat{f}_t) \coloneqq \frac{1}{|\mathcal{D}_z|} \sum_{(x,y)\in\mathcal{D}_z} \ell(\hat{f}_t, x, y) \; + \; \frac{c_0}{\sqrt{N_{z,t}}} \; + c_1 \frac{S_{z_t}}{\sqrt{\mathrm{var}(\mathrm{S}_{z_t})}} \; , \tag{6}$$

to incentivize the algorithm to sample attributes on which accuracy is increasing. c_0 and c_1 are free parameters controlling the amount of exploration and importance of per-attribute loss trends, respectively. We note that the general ability to modify the UCB in this manner is a strength of our algorithm, as it allows for a great deal of interpretability and versatility in practical applications. The parameters c_0, c_1, n_0, and b_0 are chosen based on the specific problem and available computational resources. We describe our selection of these terms App. E. In the experiments we will refer to the variant of this algorithm with $c_1 = 0$ as $\mathcal{A}_{\texttt{opt}}$, and the variant with $c_1 > 0$ as $\mathcal{A}_{\texttt{opt}}+$.

5 Empirical Results

We evaluate the performance of our proposed active sampling algorithm $\mathcal{A}_{\texttt{opt}}$ on both synthetic and real datasets, and compare it with the following baselines: **1)** ϵ-`greedy` scheme of Abernethy et al. (2020), **2)** `Greedy` scheme, which is equivalent to ϵ-`greedy` with $\epsilon = 0$, **3)** `Uniform`, where an equal number of samples are drawn for each attribute, and **4)** `Uncurated`, where samples are

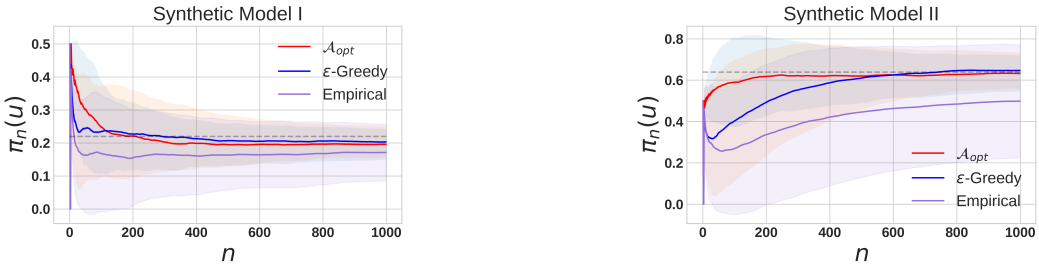


Figure 2: The figure shows the convergence of $\pi_n(u)$ for the three algorithms $\mathcal{A}_{\texttt{opt}}$, ϵ-`greedy` (with $\epsilon = 0.1$) and `Empirical` (i.e., ϵ-`greedy` with $\epsilon = 0$), to the optimal value $\pi^*(u)$ for the two instances of the `SyntheticModel1` introduced in Sec. 2.2.

drawn according to the natural distribution of the dataset. We note that, for some datasets, the natural distribution is uniform, for those we omit the results for the `Uncurated` scheme.

We will make the code for these experiments available in the supplemental materials. All experiments were run for multiple trials–the number of which is indicated in the results–and we report the average over trials with shaded regions on plots indicating ± 1 standard deviation. Experiments for image datasets were run on a single GPU provided by Google Colab or other shared computing resources.

Synthetic Dataset. In this experiment, we compare how the $\pi_n(u)$ returned by the different algorithms converge to $\pi^*(u)$ for the two synthetic models introduced in Sec. 2.2 with $\mathcal{F}$ chosen as the family of logistic regression (LR) classifiers. Since the feature space is two dimensional, we use the version of $\mathcal{A}_{\texttt{opt}}$, ϵ-`greedy`, and `Empirical` schemes in which we train (from scratch) the classifier in each round. For $\mathcal{A}_{\texttt{opt}}$, we use UCB given in 6 with $c_0 = 0.1$ and for ϵ-`greedy`, we use $\epsilon = 0.1$. These values of c_0 and ϵ are selected via a grid search.
Fig. 2 shows how $\pi_n(u)$ changes with n for the three algorithms averaged over 100 trials. As expected, the algorithms with an exploration component (i.e., $\mathcal{A}_{\texttt{opt}}$ and ϵ-`greedy`) eventually converge to the optimal $\pi^*(u)$ value in both cases, whereas the `Empirical` scheme that acts greedily often gets stuck with a wrong mixture distribution, resulting in high variability in its performance.

***Adult* and *German* Datasets.** For the remaining experiments we find that, for properly tuned values of ϵ and c_0, both algorithms attain comparable minimum test error. So we omit the ϵ-`greedy` results for the purpose of clarity, see App. C for a detailed comparison of the two algorithms.
We now analyze the performance of the remaining algorithms on two datasets from the UCI ML Repository (Dua and Graff, 2017) that are commonly used in the fairness literature: the *Adult* and *German* datasets. Both are low dimensional problems, so we use LR classifiers and the exact version of each sampling algorithm, where the optimal classifier is computed with each new sample.
For *Adult* we set $\mathcal{Z}$ to be white men, non-white men, white women, and non-white women. The minimum test accuracy, over all attributes, at each iteration is displayed in Fig. 3a. Each sampling scheme approaches 80.5% accuracy as sample size grows, this matches the results achieved under the LR algorithms reported in Table 4a) in Martinez et al. (2020). This is approximately the maximum accuracy an LR classifier can achieve on the white male sub-population in isolation and so additional sampling, or other fairness algorithms, cannot improve on this performance in a minimax sense. We do note, however, that the adaptive algorithms hold an advantage over `Uniform` at small sample sizes.
For *German* we set $\mathcal{Z}$ to be male or female. The results, in Fig. 3b, show, on average, an advantage for $\mathcal{A}_{\texttt{opt}}$ over both `Uniform` and `Greedy` schemes. But this dataset contains only 1000 examples in total. We generate a 70/30 training/test split, but find that our results strongly depend on this split. To mitigate this, we run 500 trials and generate a new random split for each, but still find very high variance in our results, indicated by the shaded region in the figure.

Image Datasets. We also compare the performance of the different sampling schemes on larger scale problems and more complex hypothesis classes. To this end we use three image datasets: *UTKFace* (Zhang et al., 2017), *FashionMNIST* (Xiao et al., 2017), and *Cifar10* (Krizhevsky and Hinton, 2009) with CNNs. We use the heuristic variants of the $\mathcal{A}_{\texttt{opt}}$, with $c_1 = 0$, and `Greedy` algorithms with batch sizes of 50. The CNN architecture, data transforms, and further algorithm parameters are detailed in App. E.

The *UTKFace* dataset consists of face images annotated with age, gender, and ethnicity. We choose $Y = \{\texttt{Male},\texttt{Female}\}$ and set $\mathcal{Z}$ to the five ethnicities. The minimum test accuracy, over all attributes, is shown in Fig. 4b and demonstrates a clear separation between the adaptive algorithms

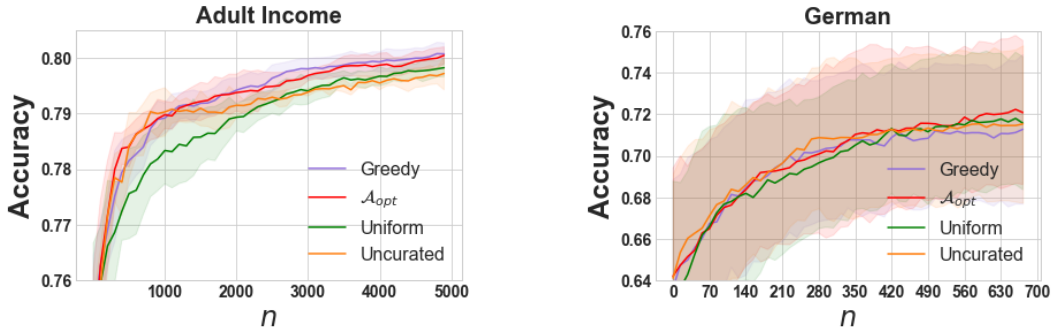


Figure 3: Minimum test accuracy, over all attributes, for both *Adult* and *German* datasets as a function of the sampling budget n, averaged over 10 trials for *Adult* and 500 for *German*.

and both `Uniform` and `Uncurated` sampling. `Uncurated` does particularly poorly in the low sample regime here, in contrast with the *Adult* and *German* datasets, where `Uncurated` performed comparably to the adaptive algorithms. This is because, for *Adult* and *German* datasets, the lowest accuracy attribute is over-represented in the dataset: white men at 63% of all samples in *Adult* and men at 69% of *German*. Whereas, for *UTKFace*, the accuracy was lowest for Asian people, who are under-represented in the available data at only 14% of all samples.

The other two datasets, *FashionMNIST* and *Cifar10*, were chosen to provide a controlled setting to demonstrate the existence of hard and easy attributes on real-world data. For both, we divide the labels into 5 pairs and assign each an "attribute". The pairs were chosen according to existing confusion matrices to have pairs that are both easy and hard to distinguish from each other, see App. E for more details.

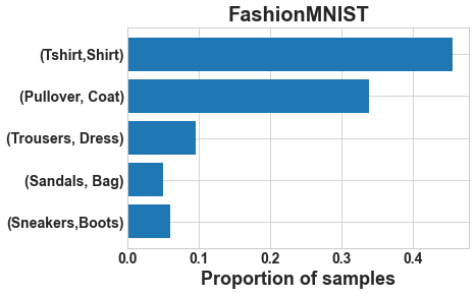


Figure 5: Mixture distribution learned by $\mathcal{A}_{\texttt{opt}}$ over 500 training rounds on *FashionMNIST*

For *FashionMNIST*, each pair in Fig. 5 is one attribute and within each pair one item is assigned $Y = 0$ and the other $Y = 1$. Then a single binary CNN classifier is learned simultaneously over all 5 pairs. Fig. 5 shows the mixture distribution generated by the $\mathcal{A}_{\texttt{opt}}$ sampling scheme, which allocated the vast majority of samples to the (Tshirt, Shirt) and (Pullover, Coat) pairs. This makes intuitive sense, since both pairs of items are qualitatively very similar to each other, and aligns with common confusion matrices which indicate that those items are frequently misclassified as each other by standard classifiers. Fig. 4a displays the worst case accuracy for each scheme on *FashionMNIST* as a function of time step and shows that both adaptive algorithms outperform `Uniform` sampling throughout the training process. Finally, Fig. 6 shows the test accuracy on each attribute for both $\mathcal{A}_{\texttt{opt}}$ and `Uniform` sampling schemes. $\mathcal{A}_{\texttt{opt}}$ maintains a much smaller spread between the accuracy over different attributes. This equity is particularly desirable from a fairness perspective as it ensures no attribute will have a large advantage over any other, in the sense of expected accuracy. It also illustrates our assumption that the optimal distribution will tend to equalize the losses across attributes.

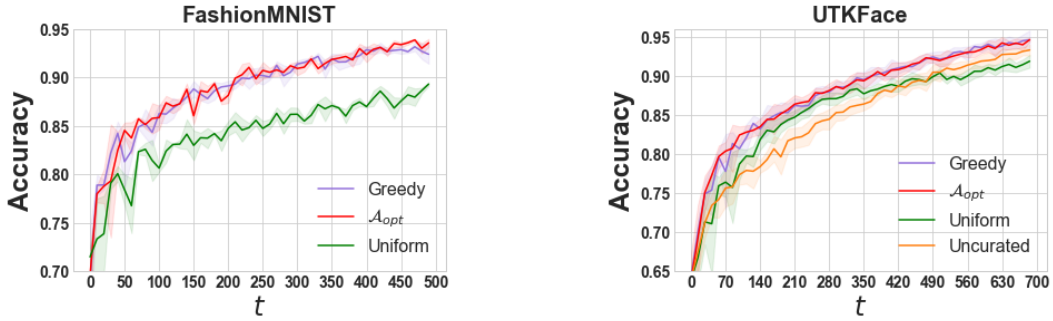


Figure 4: Minimum test accuracy, over all attributes, for both *FashionMNIST* and *UTKFace* as a function of the time step t, averaged over 10 trials.

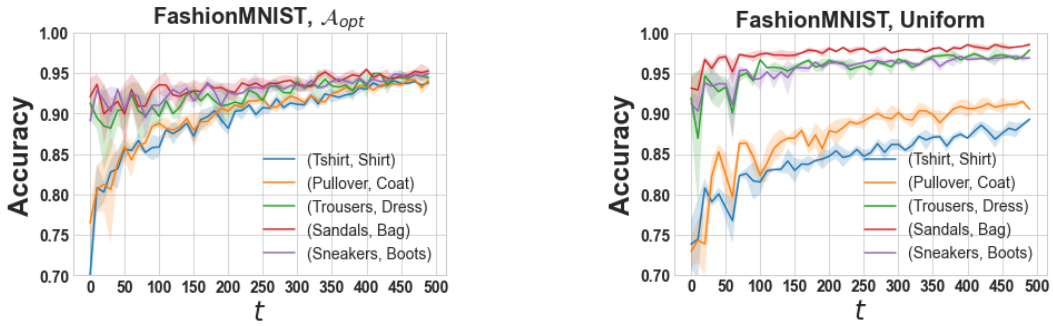


Figure 6: Test accuracy for each attribute in *FashionMNIST* as a function of the time step, t, for both $\mathcal{A}_{\texttt{opt}}$ and `Uniform` sampling schemes, averaged over 10 trials.

Final accuracy for all experiments, per-attribute accuracy for *UTKFace*, all results for *CIFAR10*, and details of the *Adult* and *German* are included in App. E.

Empirical results when Assumption 2 is violated. Finally, we consider the case where our assumption of a unique, risk equalizing mixture distribution may not hold. The practical effects of this situation can be seen in the *Adult* dataset where additional samples of the white, male attribute show diminishing returns, while other attributes can attain higher accuracy given more samples. In such a scenario, each adaptive algorithm will continue to select samples mainly from the worst case group despite this granting little improvement in performance.

To evaluate this scenario, we compared the performance of $\mathcal{A}_{\texttt{opt}}+$, the heuristic variation of our algorithm with a trend-based statistic included, $\mathcal{A}_{\texttt{opt}}$, and `Greedy` on the synthetic data from Fig. 1c. Fig. 7 shows the results, each of the three algorithms achieves similar minimax accuracy and performs worst on attribute $Z = u$. But $\mathcal{A}_{\texttt{opt}}+$ achieves 4 points higher average accuracy than both `Greedy` and $\mathcal{A}_{\texttt{opt}}$ with the original UCB on the other attribute, $Z = v$. This demonstrates that the additional term in the $\mathcal{A}_{\texttt{opt}}+$ UCB is effective at recognizing when the hardest group is not benefiting from additional training samples and redistributing them more effectively.

6 Conclusion

We considered the problem of actively constructing a training set in order to learn a classifier that achieves minimax fairness in terms of predictive loss. We proposed a new strategy for this problem ($\mathcal{A}_{\texttt{opt}}$) and obtained theoretical guarantees on its performance. We then showed that the theoretical performance achieved by $\mathcal{A}_{\texttt{opt}}$ and ϵ-`greedy` (Abernethy et al., 2020) cannot be improved in general, by obtaining algorithm independent lower-bounds for the problem.

Our experiments demonstrated that adaptive sampling schemes can achieve superior minimax risk [Fig. 3, Fig. 4] and smaller disparity between per-attribute risks [Fig. 6] compared to both uniform and uncurated schemes. The results in Fig. 2 show the necessity of exploration, as purely greedy algorithms can converge to sub-optimal mixture distributions. And finally Fig. 7 shows the versatility of our general UCB-based strategy in its ability to readily incorporate new terms to accommodate the practical challenges posed by real-world problems.

Our theoretical results rely on the existence of a unique risk equalizing mixture distribution π^*, a condition that may not always hold. Thus, an important future work is to relax this assumption, and to design and analyze algorithms that identify pareto optimal mixture distributions that achieve minimax fairness. Another important direction is to derive and analyze active sampling strategies for other fairness measures.

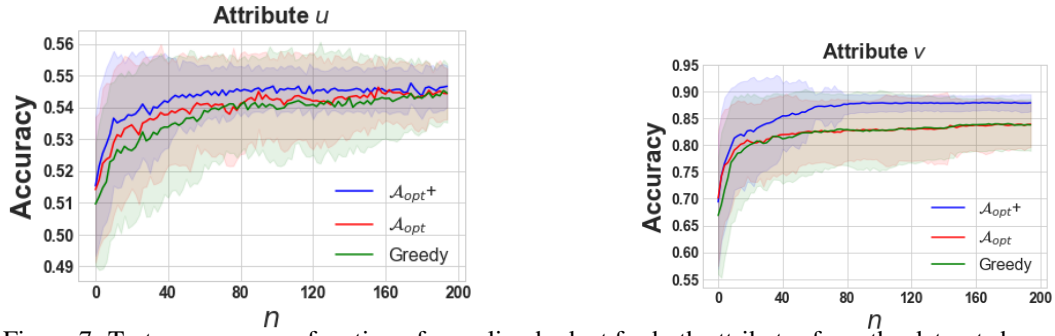


Figure 7: Test accuracy as a function of sampling budget for both attributes from the dataset shown in Fig. 1c, averaged over 100 trials.

References

Abernethy, J., Awasthi, P., Kleindessner, M., Morgenstern, J., and Zhang, J. (2020). Adaptive sampling to reduce disparate performance. *arXiv:2006.06879*.

Agarwal, A., Beygelzimer, A., Dudík, M., Langford, J., and Wallach, H. (2018). A reductions approach to fair classification. In *Proceedings of the 35th International Conference on Machine Learning*, pages 60–69.

Anahideh, H., Asudeh, A., and Thirumuruganathan, S. (2020). Fair active learning. *arXiv:2001.01796*.

Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47:235–256.

Barocas, S. and Selbst, A. (2016). Big data's disparate impact. *California Law Review*, 104.

Caton, S. and Haas, C. (2020). Fairness in machine learning: A survey. *arXiv:2010.04053*.

Celis, L. E., Keswani, V., and Vishnoi, N. (2020). Data preprocessing to mitigate bias: A maximum entropy based approach. In *International Conference on Machine Learning*, pages 1349–1359.

Chen, I., Johansson, F., and Sontag, D. (2018). Why is my classifier discriminatory? In *Advances in Neural Information Processing Systems*.

Diana, E., Gill, W., Kearns, M., Kenthapadi, K., and Roth, A. (2021). Minimax group fairness: Algorithms and experiments. *arXiv:2011.03108v2*.

Dua, D. and Graff, C. (2017). UCI machine learning repository.

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*, pages 214–226.

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

Geoffrion, A. (1968). Proper efficiency and the theory of vector maximization. *Journal of mathematical analysis and applications*, 22(3):618–630.

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

Holstein, K., Vaughan, J. W., III, H. D., Dudík, M., and Wallach, H. M. (2018). Improving fairness in machine learning systems: What do industry practitioners need? *CoRR*, abs/1812.05239.

Hooker, S. (2021). Moving beyond "algorithmic bias is a data problem". *Patterns*, 2(4):100241.

Jo, E. S. and Gebru, T. (2019). Lessons from archives: Strategies for collecting sociocultural data in machine learning. *CoRR*, abs/1912.10389.

Kendall, M. G. (1948). Rank correlation methods.

Krizhevsky, A. and Hinton, G. (2009). Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario.

Lattimore, T. and Szepesvári, C. (2020). *Bandit algorithms*. Cambridge University Press.

Lipton, Z., Chouldechova, A., and McAuley, J. (2017). Does mitigating ML's impact disparity require treatment disparity? *arXiv:1711.07076*.

Mann, H. B. (1945). Nonparametric tests against trend. *Econometrica*, 13(3):245–259.

Martinez, N., Bertran, M., and Sapiro, G. (2020). Minimax pareto fairness: A multi objective perspective. In *International Conference on Machine Learning*, pages 6755–6764.

Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I. D., and Gebru, T. (2018). Model cards for model reporting. *CoRR*, abs/1810.03993.

Polyanskiy, Y. and Wu, Y. (2015). Lecture notes on information theory.

Suresh, H. and Guttag, J. V. (2019). A framework for understanding unintended consequences of machine learning. *CoRR*, abs/1901.10002.

Ustun, B., Liu, Y., and Parkes, D. (2019). Fairness without harm: Decoupled classifiers with preference guarantees. In *Proceedings of the 36th International Conference on Machine Learning*, pages 6373–6382.

Woodworth, B., Gunasekar, S., Ohannessian, M., and Srebro, N. (2017). Learning non-discriminatory predictors. In *Proceedings of the 2017 Conference on Learning Theory*, pages 1920–1953.

Xiao, H., Rasul, K., and Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747.

Zemel, R., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. (2013). Learning fair representations. In *International Conference on Machine Learning*, pages 325–333.

Zhang, Song, and Qi (2017). Age progression/regression by conditional adversarial autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] In particular, we list the assumptions underlying our theoretical results and note that we focus on a particular notion of fairness, but that there exist other notions which may be more appropriate in certain contexts.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] The primary concern we see is a mistaken assumption that application of our algorithm removes any potential for bias in a ML model. We explicitly address this in the introduction.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] Some are deferred to the appendix for clarity and concision, we include proof sketches and intuition where we feel they are appropriate.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] All data we use is publicly available and we will make the code, with instructions, available in the supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] These are only mentioned in the main text when they're of particularly import, e.g. explaining the large error bars in the *German* experiments. For most experiments they are detailed in the supplemental material.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] We're unable to specify the type of GPUs, as we used Google Colab and a shared cluster, both of which provide different types of GPUs at different times. We've provided what details we can, and our experiments were not very compute heavy–anyone with access to a GPU will be able to reproduce them in a reasonable amount of time.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] Though, we note that the two UCI ML repository datasets we used both request that the repository itself, rather than any specific work, be cited.
   (b) Did you mention the license of the assets? [No] All data we've used is available on open access or non-commercial licenses and is well known and commonly used in the ML and fairness communities.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We will include the code used to run our experiments in the supplemental material, as Jupyter notebooks
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] This is only in question for the German dataset, where we were unable to find the data collection methodology.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] Two of the datasets we use contain information about people, but it is publicly available and anonymous, and we report only the accuracy of models that we have trained on the dataset. So we believe our paper causes no additional risk here, nor is there any offensive content involved.
5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]