Beyond Adult and COMPAS: Fairness in Multi-Class Prediction

Anonymous Author(s)

Abstract

We consider the problem of producing fair probabilistic classifiers for multi-class classification tasks. We formulate this problem in terms of "projecting" a pre-trained (and potentially unfair) classifier onto the set of models that satisfy target group-fairness requirements. The new, projected model is given by post-processing the outputs of the pre-trained classifier by a multiplicative factor. We provide a parallelizable iterative algorithm for computing the projected classifier and derive both sample complexity and convergence guarantees. Comprehensive numerical comparisons with state-of-the-art benchmarks demonstrate that our approach maintains competitive performance in terms of accuracy-fairness trade-off curves, while achieving favorable runtime on large datasets. We also introduce an open dataset with multiple classes, multiple intersectional protected groups, and over 1M samples for benchmarking fairness interventions at scale.

1 Introduction

Machine learning (ML) algorithms are increasingly used to automate decisions that have significant social consequences. This trend has led to a surge of research on designing and evaluating fairness interventions that prevent discrimination in ML models. When dealing with *group fairness*, fairness interventions aim to ensure that a ML model does not discriminate against different groups determined, for example, by race, sex, and/or nationality. Extensive comparisons between discrimination control methods can be found in [BDH+18, FSV+19, WRC21]. As these studies demonstrate, there is still no "best" fairness intervention for ML, with the majority of existing approaches being tailored to either binary classification tasks, binary population groups, or both. Moreover, discrimination control methods are often tested on overused datasets of modest size collected in either the US or Europe (e.g., UCI Adult [Lic13] and COMPAS [ALMK16]).

Most fairness interventions[1] in ML focus on binary outcomes. In this case, the classification output is either positive or negative, and group-fairness metrics are tailored to binary decisions [HPS16]. While binary classification covers a range of ML tasks of societal importance (e.g., whether to approve a loan, whether to admit a student), there are many cases where the predicted variable is not binary. For example, in education, grading algorithms assign one out of several grades to students. In healthcare, predicted outcomes are frequently not binary (e.g., severity of disease). Even the original COMPAS algorithm—a timeworn case-study in fair ML—assigned a score between 1 to 10 to each pre-trial defendant [ALMK16].

We introduce a theoretically-grounded discrimination control method that ensures group fairness in multi-class classification for several, potentially overlapping population groups. We consider group fairness metrics that are natural multi-class extensions of their binary classification counterparts, such

[1] See Related Work and Table 1 for notable exceptions.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

as statistical parity [FFM+15], equalized odds [HPS16], and error rate imbalance [PRW+17, Cho17]. When restricted to two predicted classes, our method performs competitively against state-of-the-art fairness interventions tailored to binary classification tasks. Our fairness intervention is model-agnostic (i.e., applicable to any model class) and scalable to datasets that are orders of magnitude larger than standard benchmarks found in the fair ML literature.

Our approach is based on an information-theoretic formulation, namely *information projection*. We show that this formulation is particularly well-suited for ensuring fairness in probabilistic classifiers with multi-class outputs. Given a probability distribution P and a convex set of distributions $\mathcal{P}$, the goal of information projection is to find the "closest" distribution to P in $\mathcal{P}$. The study of information projection can be traced back to [Csi75], which used KL-divergence to measure "distance" between distributions. Since then, information projection has been extended to other divergence measures, such as f-divergences [Csi95] and Rényi divergences [KS16, KS15]. Recently, [AAW+20] studied how to project a probabilistic classifier, viewed as a conditional distribution, onto the set of classifiers that satisfy target group-fairness requirements. Remarkably, the projected classifier is obtained by multiplying (i.e., post-processing) the predictions of the original classifier by a factor.

Prior work on information projection relies on a critical—and limiting—information-theoretic assumption: the underlying probability distributions are *known exactly*. This is infeasible in practical ML applications, where only a set of training samples from the underlying data distribution is available. We fill this gap by introducing an efficient procedure for computing the projected classifier with finite samples called `FairProjection`. We establish theoretical guarantees for our algorithm in terms of convergence and sample complexity. Notably, our procedure is parallelizable (e.g., on a GPU). As a result, `FairProjection` scales to datasets with the number of samples comparable to the population of many US states ($> 10^6$ samples). We provide a TensorFlow [AAB+15] implementation of our algorithm in the supplementary material (SM). We apply `FairProjection` to post-process the outputs of probabilistic classifiers in order to ensure group fairness.

We benchmark our post-processing approach against several state-of-the-art fairness interventions selected based on the availability of reproducible code, and qualitatively compare it against many others. Our numerical results are among the most comprehensive comparison of post-processing fairness interventions to date. We present performance results on the HSLS (High School Longitudinal Study, used in [JWC22]), Adult [Lic13], and COMPAS [ALMK16] datasets.

We also introduce a new dataset derived from open and anonymized data from Brazil's national high school exam—the *Exame Nacional do Ensino Médio* (ENEM)—with over 1M samples. We develop this dataset due to the need for large-scale benchmarks for evaluating fairness interventions in multi-class classification tasks. We also answer recent calls [BZZ+21, DHMS21] for moving away from overused datasets such as Adult [Lic13] and COMPAS [ALMK16]. We hope that the ENEM dataset encourages researchers in the field of fair ML to test their methods within broader contexts.

In summary, our main contributions are: (i) We introduce a post-processing fairness intervention for multi-class (i.e., non-binary) classification problems that can account for multiple protected groups and is scalable to large datasets; (ii) We derive finite-sample guarantees and convergence-rate results for our post-processing method. Importantly, `FairProjection` makes information projection practical without requiring exact knowledge of probability distributions; (iii) We demonstrate the favourable performance of our approach through comprehensive benchmarks against state-of-the-art fairness interventions; (iv) We introduce a new large-scale dataset (ENEM) for benchmarking discrimination control methods in multi-class classification tasks.

Related work. We summarize key differentiating factors from prior work in Table 1 and provide a more in-depth discussion in the SM. The fairness interventions that are the most similar to ours are the FairScoreTransformer [WRC20, WRC21, FST] and the pre-processing method in [JN20]. The FST and [JN20] can be viewed as an instantiation of `FairProjection` restricted to binary classification and cross-entropy (for FST) or KL-divergence (for [JN20]) as the f-divergence of choice. Thus, our approach is a generalization of both methods to multiple f-divergences. We also note that, unlike our method, [JN20] requires retraining a classifier multiple times.

Method	Feature						
	Multiclass	Multigroup	Scores	Curve	Parallel	Rate	Metric
Reductions [ABD+18]	✗	✓	✓	✓	✗	✓	SP, (M)EO
Reject-option [KKZ12]	✗	✓	✗	✓	✗	✗	SP, (M)EO
EqOdds [HPS16]	✗	✓	✗	✗	✗	✓	EO
LevEqOpp [CDH+19]	✗	✗	✗	✗	✗	✗	FNR
CalEqOdds [PRW+17]	✗	✗	✓	✗	✗	✓	MEO
FACT [KCT20]	✗	✗	✗	✓	✗	✗	SP, (M)EO
Identifying [JN20]	✗	✓	✓	✓	✗	✗	SP, (M)EO
FST [WRC20, WRC21]	✗	✓	✓	✓	✗	✓	SP, (M)EO
Overlapping [YCK20]	✓	✓	✓	✓	✗	✗	SP, (M)EO
Adversarial [ZLM18]	✓	✓	N/A[2]	✓	✓	✗	SP, (M)EO
`FairProjection` (ours)	✓	✓	✓	✓	✓	✓	SP, (M)EO

Table 1: Comparison between benchmark methods. **Multiclass/multigroup**: implementation takes datasets with multiclass/multigroup labels; **Scores**: processes raw outputs of probabilistic classifiers; **Curve**: outputs fairness-accuracy tradeoff curves (instead of a single point); **Parallel**: parallel implementation (e.g., on GPU) is available; **Rate**: convergence rate or sample complexity guarantee is proved. **Metric**: applicable fairness metric, with SP↔Statistical Parity, EO↔Equalized Odds, MEO↔Mean EO;

[ABD+18] introduced a reductions approach for fair classification. When restricted to binary classification, the benchmarks in Section 5 indicate that the reductions approach consistently achieves the most competitive fairness-accuracy trade-off compared to ours. The approach described here has two key differences from [ABD+18]: it is not restricted to binary classification tasks and does not require refitting a classifier several times over the training dataset. These are also key differentiating points from [CHKV19], which presented a meta-algorithm for fair classification that accounts for multiple constraints and groups. The reductions approach was later significantly generalized in the GroupFair method by [YCK20] to account for non-overlapping groups and multiple predicted classes. Unlike [YCK20], we do not require retraining classifiers.

Several other recent efforts consider optimizing accuracy under group-fairness constraints. [CJG+19] proposed a "proxy-Lagrangian" formulation for incorporating non-differentiable rate constraints which may include group fairness constraints. We avoid non-differentiability issues by considering the probabilities (scores) at the output of the classifier instead of thresholded decisions. [ZVRG17] introduced a fairness-constrained optimization applicable to margin-based classifiers (our approach can be used on any probabilistic classifier). [MW18] and [CDPF+17] characterized fairness-accuracy trade-offs in binary classification tasks when the underlying distributions are known.

Notation. Boldface Latin letters will always refer to vectors or matrices. The entries of a vector $\boldsymbol{z}$ are denoted by z_j, and those of a matrix $\boldsymbol{G}$ by $G_{i,j}$. The all-1 and all-0 vectors are denoted by $\mathbf{1}$ and $\mathbf{0}$. We set $[N] := \{1, \cdots, N\}$ and $\mathbb{R}_+ \triangleq [0, \infty)$. The probability simplex over $[N]$ is denoted by $\boldsymbol{\Delta}_N \triangleq \{\boldsymbol{p} \in \mathbb{R}_+^N \,;\, \mathbf{1}^T\boldsymbol{p} = 1\}$. If P is a Borel probability measure over $\mathbb{R}^N$, $Z \sim P$ is a random variable, and $f : \mathbb{R}^N \to \mathbb{R}^K$ is Borel, then the expectation of $f(Z)$ is denoted by $\mathbb{E}[f(Z)] = \mathbb{E}_P[f] = \mathbb{E}_{Z\sim P}[f(Z)]$. We use the standard asymptotic notations O, Θ, and Ω.

2 Problem formulation and preliminaries

Classification tasks. The essential objects in classification are the input sample space $\mathcal{X}$, the predicted classes $\mathcal{Y}$, and the classifiers. We fix two random variables X and Y, taking values in sets $\mathcal{X}$ and $\mathcal{Y} \triangleq [C]$. Here, (X, Y) is a pair comprised of an input sample and corresponding class label randomly drawn from $\mathcal{X} \times \mathcal{Y}$ with distribution $P_{X,Y}$. A probabilistic classifier is a function

[2] [ZLM18] is an in-processing method unlike other benchmarks in the table. It does not take a pre-trained classifier as an input.

Fairness Criterion	Statistical parity	Equalized odds	Overall accuracy equality
Expression	$\left\lvert \frac{P_{\widehat{Y}\mid S=a}(c')}{P_{\widehat{Y}}(c')} - 1 \right\rvert \leq \alpha$	$\left\lvert \frac{P_{\widehat{Y}\mid Y=c,S=a}(c')}{P_{\widehat{Y}\mid Y=c}(c')} - 1 \right\rvert \leq \alpha$	$\left\lvert \frac{P(\widehat{Y}=Y \mid S=a)}{P(\widehat{Y}=Y)} - 1 \right\rvert \leq \alpha$

Table 2: Standard group fairness criteria; one fixes $\alpha > 0$ and iterates over all $(a, c, c') \in [A] \times [C]^2$.

$\boldsymbol{h} : \mathcal{X} \to \boldsymbol{\Delta}_C$, where $h_c(x)$ represents the probability of sample $x \in \mathcal{X}$ falling in class $c \in \mathcal{Y}$. Thus, $\boldsymbol{h}$ gives rise to a $\mathcal{Y}$-valued random variable $\widehat{Y}$ via the distribution $P_{\widehat{Y}\mid X=x}(c) \triangleq h_c(x)$.

Group-fairness constraints. Let S be a group attribute (e.g., race and/or sex), taking values in $\mathcal{S} \triangleq [A]$. We consider multi-class generalization of three commonly used group fairness criteria in Table 2. As observed by existing works [see, e.g., ABD+18, MW18, CHKV19, WRC20, AAW+20], each of these fairness constraints[3] can be written in the vector-inequality form $\mathbb{E}_{P_X}[\boldsymbol{G}\boldsymbol{h}] \leq \boldsymbol{0}$ for a closed-form matrix-valued function $\boldsymbol{G} : \mathcal{X} \to \mathbb{R}^{K \times C}$. For instance, for statistical parity, the $\boldsymbol{G}$ matrix evaluated at a fixed individual $x \in \mathcal{X}$ has $K = 2AC$ rows indexed by $(\delta, a, c') \in \{0, 1\} \times [A] \times [C]$, where the (δ, a, c')-th row is equal to $\left((-1)^\delta P_S(a)^{-1} \sum_{c \in [C]} P_{S\mid X=x, Y=c}(a) h_c^{\text{base}}(x) - (\alpha + (-1)^\delta)\right) \boldsymbol{e}_{c'}$, with $\boldsymbol{e}_1, \cdots, \boldsymbol{e}_C$ denoting the standard basis for $\mathbb{R}^C$. The expressions for the $\boldsymbol{G}$ matrix corresponding to the other fairness metrics are given in the SM. Note that $\boldsymbol{G}$ depends on $P_{S\mid X,Y}$. If the group attribute S is part of the input feature X, then $P_{S\mid X,Y}$ is simply replaced with an indicator function. Otherwise, we approximate this conditional distribution by training a probabilistic classifier.

Goal. Our goal is to design an efficient post-processing method that takes a pre-trained classifier $\boldsymbol{h}^{\text{base}}$ that may violate some target group-fairness criteria and finds a fair classifier that has the closest utility performance to that of $\boldsymbol{h}^{\text{base}}$.

Fairness through information-projection. We formulate fair post-processing problem as follows. For a fixed search space $\mathcal{H} \subset \boldsymbol{\Delta}_C^{\mathcal{X}} \triangleq \{\boldsymbol{h} : \mathcal{X} \to \boldsymbol{\Delta}_C\}$, a loss function $\mathsf{err} : \boldsymbol{\Delta}_C^{\mathcal{X}} \times \boldsymbol{\Delta}_C^{\mathcal{X}} \to \mathbb{R}$, and a base classifier $\boldsymbol{h}^{\text{base}} \in \boldsymbol{\Delta}_C^{\mathcal{X}}$, one seeks to solve:

$$\underset{\boldsymbol{h} \in \mathcal{H}}{\text{minimize}}\ \mathsf{err}\left(\boldsymbol{h}, \boldsymbol{h}^{\text{base}}\right) \quad \text{subject to } \mathbb{E}_{P_X}\left[\boldsymbol{G}\boldsymbol{h}\right] \leq \boldsymbol{0}. \tag{1}$$

The function err quantifies the "closeness" between the scores given by $\boldsymbol{h}$ and $\boldsymbol{h}^{\text{base}}$ and we choose f-divergence to measure this:

$$\mathsf{err}\left(\boldsymbol{h}, \boldsymbol{h}^{\text{base}}\right) = D_f(\boldsymbol{h} \| \boldsymbol{h}^{\text{base}} \mid P_X) \triangleq \mathbb{E}_{P_X}\left[\sum_{c \in [C]} h_c^{\text{base}}(X) f\left(\frac{h_c(X)}{h_c^{\text{base}}(X)}\right)\right] - f(1), \tag{2}$$

where f is a convex function over $(0, \infty)$. By varying different choices of f, we can obtain e.g., cross-entropy (CE, $f(t) = -\log t$) and KL-divergence ($f(t) = t \log t$). For a chosen f-divergence, the optimization problem (1) becomes a generalization of *information projection* [Csi75].

Preliminaries on information-projection. In a recent work in [AAW+20], an optimal solution for the information projection formulation (1) was theoretically characterized. Let [4] $\mathcal{H} \triangleq \{\boldsymbol{h} \in \mathcal{C}(\mathcal{X}, \boldsymbol{\Delta}_C)\ ;\ \inf_{c,x} h_c(x) > 0\}$ and we introduce the following definition and assumption.

Definition 1. For $\boldsymbol{p} \in \boldsymbol{\Delta}_C$, let $D_f^{\text{conj}}(\,\cdot\,, \boldsymbol{p})$ denote the convex conjugate of $D_f(\,\cdot\,\|\,\boldsymbol{p})$:

$$D_f^{\text{conj}}(\boldsymbol{v}, \boldsymbol{p}) \triangleq \sup_{\boldsymbol{q} \in \boldsymbol{\Delta}_C} \boldsymbol{v}^T \boldsymbol{q} - D_f(\boldsymbol{q} \,\|\, \boldsymbol{p}). \tag{3}$$

[3] We remark that our framework can be applied to other fairness constraints, e.g., the ones in [WRC20].

[4] Here, $\mathcal{C}(\mathcal{X}, \boldsymbol{\Delta}_C)$ denotes the complete metric space of continuous functions from $\mathcal{X}$ to $\boldsymbol{\Delta}_C$, equipped with the sup-norm, i.e., $\|\boldsymbol{h}\| \triangleq \sup_{x \in \mathcal{X}} \|\boldsymbol{h}(x)\|_1$.

Assumption 1. Assume that: **(i)** $f \in \mathcal{C}^2(\mathbb{R})$, $f(1) = 0$, $f'(0^+) = -\infty$, and $f''(t) > 0$ for all $t > 0$; **(ii)** each $G_{k,c}$ is bounded, differentiable, and has bounded gradient; **(iii)** $\boldsymbol{h}^{\text{base}} \in \mathcal{H}$, and each h_c^{base} has bounded partial derivatives; and **(iv)** there is an $\boldsymbol{h} \in \mathcal{H}$ such that $\mathbb{E}_{P_X}[\boldsymbol{G}\boldsymbol{h}] < \boldsymbol{0}$.

Now, the solution for (1) can be obtained by simple tilting as described below.

Theorem 1 ([AAW+20])**.** *If $f, \boldsymbol{h}^{\text{base}}$, and $\boldsymbol{G}$ satisfy Assumption 1, then there is a unique solution $\boldsymbol{h}^{\text{opt}}$ for the optimization problem (1) for the f-divergence objective (2). Furthermore, $\boldsymbol{h}^{\text{opt}}$ is given by the tilt*

$$h_c^{\text{opt}}(x) = h_c^{\text{base}}(x) \cdot \phi\left(\gamma(x; \boldsymbol{\lambda}^\star) + v_c(x; \boldsymbol{\lambda}^\star)\right), \qquad (x, c) \in \mathcal{X} \times [C], \tag{4}$$

where: **(i)** *the function $\boldsymbol{v} : \mathcal{X} \times \mathbb{R}^K \to \mathbb{R}^C$ is defined by $\boldsymbol{v}(x; \boldsymbol{\lambda}) \triangleq -\boldsymbol{G}(x)^T \boldsymbol{\lambda}$;* **(ii)** *the function ϕ denotes the inverse of f';* **(iii)** *the function $\gamma : \mathcal{X} \times \mathbb{R}^K \to \mathbb{R}$ is characterized by $\mathbb{E}_{c \sim \boldsymbol{h}^{\text{base}}(x)}\left[\phi\left(\gamma(x; \boldsymbol{\lambda}) + v_c(x; \boldsymbol{\lambda})\right)\right] = 1$;* **(iv)** *$\boldsymbol{\lambda}^\star \in \mathbb{R}^K$ is a solution to the convex problem*

$$D^* \triangleq \min_{\boldsymbol{\lambda} \in \mathbb{R}_+^K} \mathbb{E}\left[D_f^{\text{conj}}\left(\boldsymbol{v}(X; \boldsymbol{\lambda}), \boldsymbol{h}^{\text{base}}(X)\right)\right]. \tag{5}$$

If the underlying distribution is known, Theorem 1 yields an expression for the projected classifier as a post-processing of the base classifier. However, in practical scenarios, we do not know the underlying distribution and have to approximate it with finite samples. In Section 3, we first describe how we approximate the solution given in Theorem 1 with finite samples. Then, we propose a parallelizable algorithm to solve the approximation in Section 4.

3 A finite-sample approximation of information projection

In practice, P_X is unknown and only data points $\mathbb{X} \triangleq \{X_i\}_{i \in [N]} \subset \mathcal{X}$, drawn from P_X, are available. Thus, we propose the following fairness optimization problem. We search for a (multi-class) classifier $\boldsymbol{h} : \mathbb{X} \to \boldsymbol{\Delta}_C$ that solves the following:

$$\begin{aligned} \underset{\substack{\boldsymbol{h}:\mathbb{X}\to\boldsymbol{\Delta}_C \\ \boldsymbol{a}:\mathbb{X}\to\mathbb{R}^C, \boldsymbol{b}\in\mathbb{R}^K}}{\text{minimize}} \quad & D_f\left(\boldsymbol{h} \,\|\, \boldsymbol{h}^{\text{base}} \mid \hat{P}_X\right) + \tau_1 \cdot \left(\mathbb{E}_{X \sim \hat{P}_X}\left[\|\boldsymbol{a}(X)\|_2^2\right] + \|\boldsymbol{b}\|_2^2\right) \\ \text{subject to} \quad & \mathbb{E}_{\hat{P}_X}\left[\boldsymbol{G} \cdot (\boldsymbol{h} + \tau_2 \boldsymbol{a})\right] \leq \tau_2 \boldsymbol{b}, \end{aligned} \tag{6}$$

with $\hat{P}_X$ being the empirical measure, and $\tau_1, \tau_2 > 0$ prescribed constants. The terms $\boldsymbol{a}$ and $\boldsymbol{b}$ are added to circumvent infeasibility issues and aid convergence of our numerical procedure. We show in the following theorem that there is a unique solution for (6), and that it is given by a tilt (i.e., multiplicative factor) of $\boldsymbol{h}^{\text{base}}$. The tilting parameter is the solution of a finite-dimensional strongly convex optimization problem (all proofs are given in the SM).

Theorem 2. *Suppose Assumption 1 holds, and set $\zeta \triangleq \tau_2^2/\tau_1$. There exists a unique solution $\boldsymbol{h}^{\text{opt},N}$ to (6), and it is given by the formula*

$$h_c^{\text{opt},N}(x) = h_c^{\text{base}}(x) \cdot \phi\left(\boldsymbol{v}_c(x; \boldsymbol{\lambda}_{\zeta,N}^*) + \gamma(x; \boldsymbol{\lambda}_{\zeta,N}^*)\right), \quad (x, c) \in \mathbb{X} \times [C], \tag{7}$$

with $\boldsymbol{v}, \phi, \gamma$ as in Theorem 1, and $\boldsymbol{\lambda}_{\zeta,N}^\star \in \mathbb{R}^K$ is the unique solution to the strongly convex problem

$$D_{\zeta,N}^* \triangleq \min_{\boldsymbol{\lambda} \in \mathbb{R}_+^K} \mathbb{E}_{\hat{P}_X}\left[D_f^{\text{conj}}\left(\boldsymbol{v}(X; \boldsymbol{\lambda}), \boldsymbol{h}^{\text{base}}(X)\right)\right] + \frac{\zeta}{2}\left\|\boldsymbol{\mathcal{G}}_N^T \boldsymbol{\lambda}\right\|_2^2 \tag{8}$$

where $\boldsymbol{\mathcal{G}}_N \triangleq \left(\boldsymbol{G}(X_1)/\sqrt{N}, \cdots, \boldsymbol{G}(X_N)/\sqrt{N}, \boldsymbol{I}_K\right) \in \mathbb{R}^{K \times (NC+K)}$.

Theorem 2 shows that: strong duality holds between the primal (6) and (the negative of) the dual (8); there is a unique classifier $\boldsymbol{h}^{\text{opt},N}$ minimizing our fairness formulation (6); there is a unique solution $\boldsymbol{\lambda}_{\zeta,N}^*$ to the dual (5); and there is an explicit functional form of $\boldsymbol{h}^{\text{opt},N}$ in terms of $\boldsymbol{\lambda}_{\zeta,N}^*$ (in (7)).

The key distinction between our formulation and Theorem 1 is that we use the empirical measure $\hat{P}_X$ (e.g., produced using a dataset with i.i.d. samples) and we have a *strongly* convex dual problem.

Hence, Theorem 2 yields a *practical* two-step procedure for solving the functional optimization in equation (6): (i) compute the dual variables by solving the strongly convex optimization in (8); (ii) tilt the base classifier by using the dual variables according to (7). This process is applied on real-world datasets using `FairProjection` (see Algorithm 1) in the next section.

The results of Theorem 2 are tightly related to those in information projection shown in Theorem 1, which corresponds to the case $\tau_1 = \tau_2 = \zeta = 0$ and P_X in place of $\hat{P}_X$. We show in Theorem 4 in the next section that the choice $\zeta \propto N^{-1/2}$ yields a sense in which our numerically obtained (from `FairProjection`) tilting parameters $\boldsymbol{\lambda}$ work well for the population problem (5).

Remark 1. In practice, Assumption 1 is not a limiting factor for Theorem 2 and `FairProjection`. This is because: we are considering here a finite-set domain so continuity is automatic; we can perturb $\boldsymbol{h}^{\text{base}}$ by negligible noise to push it away from the simplex boundary; and the uniform classifier is strictly feasible. Nevertheless, Assumption 1 simplifies the derivation of our theoretical results.

4 Fair projection and theoretical guarantees

We introduce a parallelizable algorithm, `FairProjection`, that solves (6) using N i.i.d. data points. We prove that its utility converges to D^* (see (5)) in the population limit and establish both sample-complexity and convergence rate guarantees. Applying `FairProjection` to the group-fairness intervention problem in (1) yields the optimal parameters in (7) for post-processing (i.e., tilting) the output of a multi-class classifier in order to satisfy target fairness constraints.

The `FairProjection` algorithm uses ADMM [BPC+11] to solve the convex program in (8). Recall that it suffices to optimize (8) for computing (6) as proved in Theorem 2. Algorithm 1 presents the steps of `FairProjection`(detailed derivation in the SM). A salient feature of `FairProjection` is its *parallelizability*. Each step that is done for i varying over $[N]$ can be executed for each i separately and in parallel. In particular, this applies to the most computationally intensive step, the $\boldsymbol{v}_i$-update step. We discuss next how the $\boldsymbol{v}_i$-update step is carried out.

Algorithm 1 : `FairProjection` for solving (8).

Input: divergence f, predictions $\{\boldsymbol{p}_i \triangleq \boldsymbol{h}^{\text{base}}(X_i)\}_{i\in[N]}$, constraints $\{\boldsymbol{G}_i \triangleq \boldsymbol{G}(X_i)\}_{i\in[N]}$, regularizer ζ, ADMM penalty ρ, and initializers $\boldsymbol{\lambda}$ and $(\boldsymbol{w}_i)_{i\in[N]}$.
Output: $h_c^{\text{opt},N}(x) \triangleq h_c^{\text{base}}(x) \cdot \phi(\gamma(x;\boldsymbol{\lambda}) + v_c(x;\boldsymbol{\lambda}))$.

$\boldsymbol{Q} \leftarrow \frac{\zeta}{2}\boldsymbol{I} + \frac{\rho}{2N}\sum_{i\in[N]} \boldsymbol{G}_i\boldsymbol{G}_i^T$
for $t = 1, 2, \cdots, t'$ **do**
  $\boldsymbol{a}_i \leftarrow \boldsymbol{w}_i + \rho\boldsymbol{G}_i^T\boldsymbol{\lambda}$ $\quad i \in [N]$
  $\boldsymbol{v}_i \leftarrow \operatorname{argmin}_{\boldsymbol{v}\in\mathbb{R}^C} D_f^{\text{conj}}(\boldsymbol{v},\boldsymbol{p}_i) + \frac{\rho+\zeta}{2}\|\boldsymbol{v}\|_2^2 + \boldsymbol{a}_i^T\boldsymbol{v}$ $\quad i \in [N]$
  $\boldsymbol{q} \leftarrow \frac{1}{N}\sum_{i\in[N]} \boldsymbol{G}_i \cdot (\boldsymbol{w}_i + \boldsymbol{v}_i)$
  $\boldsymbol{\lambda} \leftarrow \operatorname{argmin}_{\boldsymbol{\ell}\in\mathbb{R}_+^K} \boldsymbol{\ell}^T\boldsymbol{Q}\boldsymbol{\ell} + \boldsymbol{q}^T\boldsymbol{\ell}$
  $\boldsymbol{w}_i \leftarrow \boldsymbol{w}_i + \rho \cdot (\boldsymbol{v}_i + \boldsymbol{G}_i^T\boldsymbol{\lambda})$ $\quad i \in [N]$
end for

Inner iterations. One way to carry out the inner iteration in Algorithm 1 that updates $\boldsymbol{v}_i$ is to study the vanishing of the gradient of $\boldsymbol{v} \mapsto D_f^{\text{conj}}(\boldsymbol{v},\boldsymbol{p}_i) + \xi\|\boldsymbol{v}\|_2^2 + \boldsymbol{a}_i^T\boldsymbol{v}$ (where $\xi = (\rho+\zeta)/2$ and $\boldsymbol{a}_i \in \mathbb{R}^C$ is some vector). In the KL-divergence case, $D_{\mathsf{KL}}^{\text{conj}}$ is given by a log-sum-exp function, so its gradient is given by a softmax function, and equating the gradient to zero becomes a fixed-point equation. We give an iterative routine to solve this fixed point equation, after showing that the softmax function is $\frac{1}{2}$-Lipschitz, in the SM. Beyond the KL-divergence case, setting the gradient to zero does not seem to be an analytically tractable problem. Nevertheless, we may reduce the vector minimization in the $\boldsymbol{v}_i$-step to a tractable 1-dimensional root-finding problem, as the following result aids in showing.

Lemma 1. *For $\boldsymbol{p} \in \boldsymbol{\Delta}_C$, $\boldsymbol{a} \in \mathbb{R}^C$, and $\xi > 0$, if f satisfies Assumption 1, we have that*

$$\min_{\boldsymbol{v}\in\mathbb{R}^C} D_f^{\text{conj}}(\boldsymbol{v},\boldsymbol{p}) + \xi\|\boldsymbol{v}\|_2^2 + \boldsymbol{a}^T\boldsymbol{v} = -\sup_{\theta\in\mathbb{R}} -\theta + \sum_{c\in[C]} \min_{q_c\geq 0} p_c f\left(\frac{q_c}{p_c}\right) + \frac{(a_c+q_c)^2}{4\xi} + \theta q_c. \quad (9)$$

We note that the $\boldsymbol{v}_i$-update steps for both KL and CE (provided in detail in the SM) give, as a byproduct, the implicitly defined function $\gamma(x;\boldsymbol{\lambda})$.

Convergence guarantees. Our proposed algorithm, `FairProjection`, enjoys the following convergence guarantees. First, the output after the t-th iteration $\boldsymbol{\lambda}_{\zeta,N}^{(t)}$ converges exponentially fast to $\boldsymbol{\lambda}_{\zeta,N}^{*}$ (see (8)).

Theorem 3. *If Assumption 1 holds, the* `FairProjection` *algorithm for KL-divergence converges in $t' = O(\log N)$ steps, and runs in time $O(N \log N)$, to the unique solution $\boldsymbol{\lambda}_{\zeta,N}^{*}$ of (8). Further, if $\boldsymbol{\lambda}_{\zeta,N}^{(t)}$ and $\boldsymbol{h}^{(t)}$ are the t-th iteration outputs of* `FairProjection`*, then $\|\boldsymbol{\lambda}_{\zeta,N}^{(t)} - \boldsymbol{\lambda}_{\zeta,N}^{*}\|_2 = O(e^{-t})$ and $\boldsymbol{h}^{(t)}(x) = \boldsymbol{h}^{\mathrm{opt},N}(x) \cdot (1 + O(e^{-t}))$ uniformly in x as $t \to \infty$.*

Second, the parameter $\boldsymbol{\lambda}_{\zeta,N}^{(\log N)}$ obtainable from `FairProjection` performs well for the *population* problem for information projection (5).

Theorem 4. *Suppose Assumption 1 holds, and consider the KL-divergence case. Then, choosing $\zeta \propto N^{-1/2}$ and $t = \Omega(\log N)$ we obtain for any $\delta \in (0,1)$ and $N = \Omega(\log \frac{1}{\delta})$ that (see (5))*

$$\Pr\left\{\mathbb{E}_X\left[D_{\mathsf{KL}}^{\mathrm{conj}}\left(\boldsymbol{v}\left(X;\boldsymbol{\lambda}_{\zeta,N}^{(t)}\right), \boldsymbol{h}^{\mathrm{base}}(X)\right)\right] > D^{\star} + O\left(\frac{1}{\sqrt{N}}\right)\right\} \leq \delta. \tag{10}$$

Remark 2. Though Theorems 3–4 are shown for the KL-divergence, the majority of the proofs applies to general f-divergences. In fact, only Lipschitz continuity of the gradient of $D_{\mathsf{KL}}^{\mathrm{conj}}$ is the specific property that we apply to get the KL-divergence case.

Benefit of parallelization. The parallelizability of `FairProjection` provides significant speedup. In the SM, we provide an ablation study comparing the speedup due to parallelization. For the ENEM dataset (discussed next section), parallelization yields a 15-fold reduction in runtime. In addition to the parallel advantage of `FairProjection`, its inherent mathematical approach is more advantageous than gradient-based solutions. When numerically solving the dual problem (8) (or any close variant) via gradient methods, the gradient of D_f^{conj} (the convex conjugate of an f-divergence) must be computed. However, this gradient is tractable in only a very limited number of relevant instances of f-divergences. `FairProjection` tackles this intractability via a sequence of mathematical reductions, as Lemma 1 and the discussion preceding it show.

5 Numerical benchmarks

We present empirical results and show that `FairProjection` has competitive performance both in terms of runtime and fairness-accuracy trade-off curves compared to benchmarks—most notably [ABD+18], which requires retraining. Extensive additional benchmarks and experiment details are reported in the SM.

Setup. We consider three base classifiers (`Base`): gradient boosting (GBM), logistic regression (LR), and random forest (RF), implemented by Scikit-learn [PVG+11]. For `FairProjection` (the constrained optimization in (6)), we use cross-entropy (`FairProjection-CE`) and KL-divergence (`FairProjection-KL`) as the loss function.[5] We consider two fairness constraints: mean equalized odds (MEO) and statistical parity (SP) (cf. Table 2). All values reported in this section are from the test set with 70/30 train-test split. When benchmarking against methods tailored to binary classification, we restrict our results to both binary Y and S since, unlike `FairProjection`, competing methods cannot necessarily handle multi-class predictions and multiple groups.

Datasets. We evaluate `FairProjection` and all benchmarks on four datasets. We use two datasets in the education domain: the high-school longitudinal study (HSLS) dataset [IPH+11, JWC22] and a novel dataset we introduce here called ENEM [INE20] (details in SM). The ENEM dataset contains Brazilian college entrance exam scores along with student demographic information and socio-economic questionnaire answers (e.g., if they own a computer). After pre-processing, the

[5] We focus on `FairProjection-CE` and random forest here; results for `FairProjection-KL` and other models are in the SM.

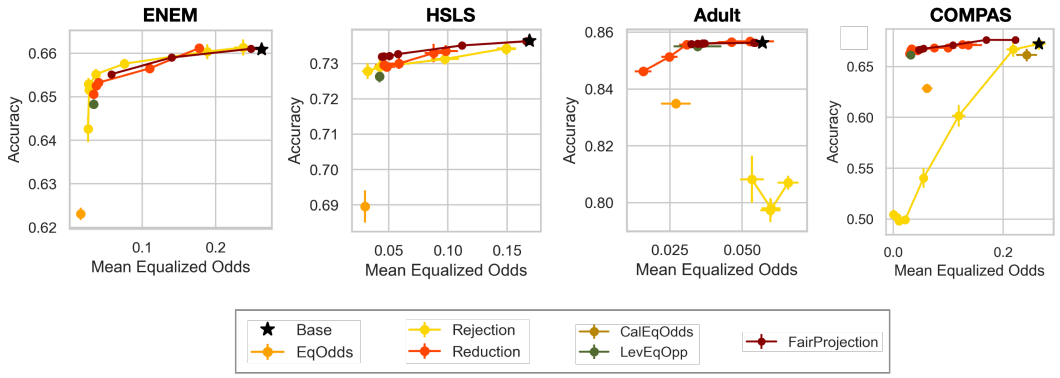

Figure 1: Fairness-accuracy trade-off comparisons between `FairProjection` and five baselines on ENEM-50k-2C, HSLS, Adult and COMPAS datasets. For all methods, we used random forest as a base classifier.

dataset contains $\sim$1.4 million samples with 139 features. Race was used as the group attribute S, and Humanities exam score is used as the label Y. The score can be quantized into an arbitrary number of classes. For binary experiments, we quantize Y into two classes, and for multi-class, we quantize it to 5 classes. The race feature S has 5 categories, but we binarize it into White and Asian ($S = 1$) and others ($S = 0$). We call the entire ENEM dataset ENEM-1.4M. We also created smaller versions of the dataset with 50k samples: ENEM-50k-2C (binary classes) and ENEM-50k-5C (5 classes)[6]. For completeness, we report results on UCI Adult [Lic13] and COMPAS [ALMK16].

Benchmarks. We compare our method with five existing fair learning algorithms: `Reduction` [ABD+18], reject-option classifier [KKZ12, `Rejection`], equalized-odds [HPS16, `EqOdds`], calibrated equalized-odds [PRW+17, `CalEqOdds`], and leveraging equal opportunity [CDH+19, `LevEqOpp`][7]. The choice of benchmarks is based on the availability of reproducible codes. For the first four baselines, we use IBM AIF360 library [BDH+18]. For `Reduction` and `Rejection`, we vary the tolerance to achieve different operation points on the fairness-accuracy trade-off curves. As `EqOdds`, `CalEqOdds` and `LevEqOpp` only allow hard equality constraint on equalized odds, they each produce a single point on the plot (see Fig. 1). We include the group attribute as a feature in the training set following the same benchmark procedure described in [ABD+18, WRC21] for a consistent comparison. Additional comparisons to [KCT20] are given in the SM.

Binary classification results. We compare `FairProjection` with benchmarks tailored to binary classification in terms of the MEO-accuracy trade-off on the ENEM-50k-2C, HSLS, Adult, and COMPAS datasets in Fig. 1. Each point is obtained by averaging 10 runs with different train-test splits. `FairProjection-CE` curves were obtained by varying α values (cf. Table 2). When $\alpha = 1.0$, the outputs of `FairProjection-CE` are equivalent to the base classifier RF.

We observe that `FairProjection-CE` and `Reduction` have the overall best and most consistent performances. On ENEM-50k-2C and HSLS datasets, although `EqOdds` achieves the best fairness, that fairness comes at the cost of 4% accuracy drop. The other four methods, on the other hand, produce comparatively good fairness with an accuracy loss of $< 1\%$. In particular, `FairProjection-CE` has the smallest accuracy drop whilst improving MEO from 0.17 to 0.04 on HSLS. `CalEqOdds` requires strict calibration requirements and yields inconsistent performance when this requirement is not met. On ENEM-50k-2C and HSLS, `LevEqOpp` achieves comparable MEO with a slight accuracy drop, and on COMPAS, `LevEqOpp` performs equally well as `FairProjection-CE` and `Reduction`. Note that with high fairness constraints (i.e., small tolerance), the accuracy of `Rejection` deteriorates.

[6] A datasheet (see [GMV+21]) for ENEM is given in the SM.
[7] https://github.com/lucaoneto/NIPS2019_Fairness.

Method	***Reduction*** [ABD+18]	**Rejection** [KKZ12]	EqOdds [HPS16]	LevEqOpp [CDH+19]	CalEqOdds [PRW+17]	***FairProjection*** (ours)
Runtime	223.6	16.9	5.9	7.9	5.3	11.2

Table 3: Execution time of `FairProjection` on the ENEM-1.4M-2C compared with five baseline methods (time shown in minutes). Methods in **bold** are capable of producing the full fairness-accuracy trade-off curves. Methods that are *italicized* have a uniformly superior performance.

Multi-Class results. We illustrate how `FairProjection`performs on multi-class prediction using ENEM-50k-5C. To measure multi-class MEO, we define:

$$\mathsf{MEO} = \max_{i \in \mathcal{Y}} \max_{s_1, s_2 \in \mathcal{S}} (|\mathsf{TPR}_i(s_1) - \mathsf{TPR}_i(s_2)| + |\mathsf{FPR}_i(s_1) - \mathsf{FPR}_i(s_2)|)/2 \tag{11}$$

where $\mathsf{TPR}_i(s) = P(\widehat{Y} = i | Y = i, S = s)$, and $\mathsf{FPR}_i(s) = P(\widehat{Y} = i | Y \neq i, S = s)$.

In Figure 2, we plot fairness-accuracy trade-off of `FairProjection-CE` with logistic regression and adversarial debiasing [ZLM18, `Adversarial`]. As their base classifiers are different (`Adversarial` is a GAN-based method), we plot accuracy difference compared to the base classifier instead of plotting the absolute value of accuracy[8]. `FairProjection`reduces MEO significantly with very small loss in accuracy. While `Adversarial` is also able to reduce MEO with negligible accuracy drop, it does not reduce the MEO as much as `FairProjection`. We show more extensive results with multi-group and multi-class ($|\mathcal{Y}| = 5, = |\mathcal{S}| = 5$) in the SM.

Runtime comparisons. In Table 3, we record the runtime of `FairProjection-CE` with the five benchmarks on ENEM-1.4M-2C. These experiments were run on a machine with AMD Ryzen 2990WX 64-thread 32-Core CPU and NVIDIA TITAN Xp 12-GB GPU. For consistency, we used the same fairness metric (MEO, $\alpha = 0.01$), base classifier (GBM), and train/test split, and each number is the average of 2 repeated experiments. `EqOdds`, `LevEqOpp`, and `CalEqOdds` are faster than `FairProjection` since they are optimized to produce one trade-off point (cf. Fig. 1). Compared to baselines that produce full fairness-accuracy trade-off curves, i.e., `Reduction` and `Rejection`, `FairProjection` has the fastest runtime. Note that the 11.2 mins reported here for `FairProjection` includes the time to fit the base classifiers. If base classifiers are pre-trained, the runtime of `FairProjection` is **1.63 mins**. Also, the non-parallel implementation of `FairProjection` takes 25.3 mins—parallelization attains 15x speedup (detailed results in the SM).

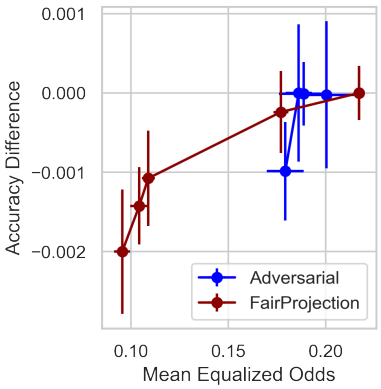

Figure 2: Fairness-accuracy trade-off for multi-class prediction on ENEM-50k-5C. FairProjection is `FairProjection-CE` with LR base classifier.

6 Final remarks and limitations

We only consider group-fairness and it would be interesting to try to incorporate other fairness notions (e.g., individual fairness) into our formulation. We assume that $\boldsymbol{h}^{\text{base}}$ is a pre-trained accurate (and potentially unfair) classifier. One future research direction is understanding how the accuracy of $\boldsymbol{h}^{\text{base}}$ influences the performance of the projected classifier. A rigorous transferability result from near-optimality guarantees for the dual problem (8) to the primal problem (1) (with an f-divergence objective) is also an important future line of work. Finally, the performance of `FairProjection` is inherently constrained by data availability. Performance may degrade with intersectional increases of the number of groups, the number of labels, and the number of fairness constraints.

[8]Base accuracy for `FairProjection`$= 0.336$, `Adversarial`$= 0.307$. Random guessing accuracy$= 0.2$.

References

[AAB+15] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[AAW+20] Wael Alghamdi, Shahab Asoodeh, Hao Wang, Flavio P. Calmon, Dennis Wei, and Karthikeyan Natesan Ramamurthy. Model projection: Theory and applications to fair machine learning. In *2020 IEEE International Symposium on Information Theory (ISIT)*, pages 2711–2716, 2020.

[ABD+18] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. In *International Conference on Machine Learning*, pages 60–69. PMLR, 2018.

[ALMK16] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *ProPublica*, 2016.

[BDH+18] Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, et al. Ai fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. *arXiv preprint arXiv:1810.01943*, 2018.

[BPC+11] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, jan 2011.

[BZZ+21] Michelle Bao, Angela Zhou, Samantha A Zottola, Brian Brubach, Sarah Desmarais, Aaron Seth Horowitz, Kristian Lum, and Suresh Venkatasubramanian. It's COMPASlicated: The messy relationship between RAI datasets and algorithmic fairness benchmarks. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.

[CDH+19] Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, Luca Oneto, and Massimiliano Pontil. Leveraging labeled and unlabeled data for consistent fair binary classification. *Advances in Neural Information Processing Systems*, 32, 2019.

[CDPF+17] Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 797–806, 2017.

[CHKV19] L Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K Vishnoi. Classification with fairness constraints: A meta-algorithm with provable guarantees. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 319–328, 2019.

[Cho17] Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163, 2017.

[CJG+19] Andrew Cotter, Heinrich Jiang, Maya R Gupta, Serena Wang, Taman Narayan, Seungil You, and Karthik Sridharan. Optimization with non-differentiable constraints with applications to fairness, recall, churn, and other goals. *J. Mach. Learn. Res.*, 20(172):1–59, 2019.

[Csi75] Imre Csiszár. I-divergence geometry of probability distributions and minimization problems. *The annals of probability*, pages 146–158, 1975.

[Csi95] Imre Csiszár. Generalized projections for non-negative functions. In *Proceedings of 1995 IEEE International Symposium on Information Theory*, page 6. IEEE, 1995.

[DHMS21] Frances Ding, Moritz Hardt, John Miller, and Ludwig Schmidt. Retiring adult: New datasets for fair machine learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 6478–6490. Curran Associates, Inc., 2021.

[FFM+15] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 259–268, 2015.

[FSV+19] Sorelle A Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P Hamilton, and Derek Roth. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 329–338, 2019.

[GMV+21] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92, 2021.

[HPS16] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29:3315–3323, 2016.

[INE20] INEP. Instituto nacional de estudos e pesquisas educaionais anísio teixeira, microdados do ENEM. https://www.gov.br/inep/pt-br/acesso-a-informacao/dados-abertos/microdados/enem, 2020. Accessed: 2022-05-23.

[IPH+11] Steven J Ingels, Daniel J Pratt, Deborah R Herget, Laura J Burns, Jill A Dever, Randolph Ottem, James E Rogers, Ying Jin, and Steve Leinwand. High school longitudinal study of 2009 (hsls: 09): Base-year data file documentation. nces 2011-328. *National Center for Education Statistics*, 2011.

[JN20] Heinrich Jiang and Ofir Nachum. Identifying and correcting label bias in machine learning. In *International Conference on Artificial Intelligence and Statistics*, pages 702–712. PMLR, 2020.

[JWC22] Haewon Jeong, Hao Wang, and Flavio Calmon. Fairness without imputation: A decision tree approach for fair prediction with missing values. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022.

[KCT20] Joon Sik Kim, Jiahao Chen, and Ameet Talwalkar. Fact: A diagnostic for group fairness trade-offs. In *International Conference on Machine Learning*, pages 5264–5274. PMLR, 2020.

[KKZ12] F. Kamiran, A. Karim, and X. Zhang. Decision theory for discrimination-aware classification. In *2012 IEEE 12th International Conference on Data Mining*, pages 924–929, Dec 2012.

[KS15] M Ashok Kumar and Rajesh Sundaresan. Minimization problems based on relative α-entropy i: Forward projection. *IEEE Transactions on Information Theory*, 61(9):5063–5080, 2015.

[KS16] M Ashok Kumar and Igal Sason. Projection theorems for the rényi divergence on α-convex sets. *IEEE Transactions on Information Theory*, 62(9):4924–4935, 2016.

[Lic13] M. Lichman. UCI machine learning repository, 2013.

[MW18] Aditya Krishna Menon and Robert C Williamson. The cost of fairness in binary classification. In *Conference on Fairness, Accountability and Transparency*, pages 107–118. PMLR, 2018.

[PRW+17] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On fairness and calibration. *arXiv preprint arXiv:1709.02012*, 2017.

[PVG+11] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[WRC20] Dennis Wei, Karthikeyan Natesan Ramamurthy, and Flavio P Calmon. Optimized score transformation for fair classification. In *23rd International Conference on Artificial Intelligence and Statistics*, 2020.

[WRC21] Dennis Wei, Karthikeyan Natesan Ramamurthy, and Flavio P Calmon. Optimized score transformation for consistent fair classification. *Journal of Machine Learning Research*, 22(258):1–78, 2021.

[YCK20] Forest Yang, Mouhamadou Cisse, and Oluwasanmi O Koyejo. Fairness with overlapping groups; a probabilistic perspective. *Advances in Neural Information Processing Systems*, 33, 2020.

[ZLM18] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340, 2018.

[ZVRG17] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial Intelligence and Statistics*, pages 962–970, 2017.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Sections 3–4 for theoretical results, and Section 5 for experimental results.
   (b) Did you describe the limitations of your work? [Yes] See the '**Final remarks and limitations**' paragraph at the end of Section 5, and also the end of the '**Group-fairness constraints**' paragraph in Section 2.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See '**Final remarks and limitations**' paragraph at the end of Section 5, e.g., lack of samples could negatively impact performance.
   (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] See Assumption 1 and Remark 1.
   (b) Did you include complete proofs of all theoretical results? [Yes] See SM.
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] See, see SM.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See, see SM and main text.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See, all results employ cross validation.
   (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)? [Yes] See, see numerical results section.
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]
   (b) Did you mention the license of the assets? [Yes] Yes, details (including licenses) are in the SM.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Yes, see SM.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] Yes, only publicly and freely available code was used.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] Yes. In particular, discussion regarding the ENEM dataset is available in the SM.
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]