Repairing without Retraining: Avoiding Disparate Impact with Counterfactual Distributions

Hao Wang [1] **Berk Ustun** [1] **Flavio P. Calmon** [1]

Abstract

When the performance of a machine learning model varies over groups defined by sensitive attributes (e.g., gender or ethnicity), the performance disparity can be expressed in terms of the probability distributions of the input and output variables over each group. In this paper, we exploit this fact to reduce the disparate impact of a fixed classification model over a population of interest. Given a black-box classifier, we aim to eliminate the performance gap by perturbing the distribution of input variables for the disadvantaged group. We refer to the perturbed distribution as a *counterfactual distribution*, and characterize its properties for common fairness criteria. We introduce a descent algorithm to learn a counterfactual distribution from data. We then discuss how the estimated distribution can be used to build a data preprocessor that can reduce disparate impact without training a new model. We validate our approach through experiments on real-world datasets, showing that it can repair different forms of disparity without a significant drop in accuracy.

1. Introduction

A machine learning model has *disparate impact* when its performance changes across groups defined by *sensitive attributes* such as race or gender (Barocas & Selbst, 2016). Recent work has shown that models can exhibit significant performance disparities between groups (see e.g. Angwin et al., 2016; Dastin, 2018). Such disparities have led to a plethora of research on fair machine learning, focusing on how disparate impact arises (Chen et al., 2018; Datta et al., 2016), how it can be measured (Žliobaitė, 2017; Pierson et al., 2017; Simoiu et al., 2017; Kilbertus et al., 2017; Kusner et al., 2017; Galhotra et al., 2017), and how it can be mitigated (Feldman et al., 2015; Corbett-Davies et al., 2017; Zafar et al., 2017; Calmon et al., 2017; Menon & Williamson, 2018; Canetti et al., 2019). Despite these developments, disparate impact remains difficult to avoid in a large class of real-world applications where:

- Models are procured from a third-party vendor who has the data or technical expertise required for model development (Guszcza et al., 2018).
- Models are deployed on a population where the data distribution differs from the distribution of the training data (i.e., due to dataset shift, Sugiyama et al., 2017).

In such settings, disparate impact is challenging to address, let alone understand. Users typically have black-box access to the model (e.g., via a prediction API), may not have access to the training data (e.g., due to privacy concerns or intellectual property rights), and may not be able to draw conclusions from the training data (e.g., due to distributional shifts in deployment).

In this paper, we aim to mitigate disparate impact in such settings. Our object of interest is a hypothetical distribution of input variables that minimizes disparate impact in the model's *deployment population*. We refer to this distribution as a *counterfactual distribution*. As we will show, an information-theoretic analysis of counterfactual distributions has much to offer. Given a fixed classifier, disparate impact can be expressed in terms the distributions of input and output variables between groups (cf. Figure 1). In turn, a counterfactual distribution can be obtained by repeatedly perturbing the distribution of input variables until a specific measure of disparity is minimized over the deployment population. The counterfactual distribution can then be used to repair the model so that it no longer exhibits disparate impact in deployment.

Contributions The main contributions of this paper are:

1. We introduce a theoretical framework to mitigate performance disparities for a black-box classifiers in deployment.
2. We develop machinery to learn counterfactual distributions from data. Our tools recover a counterfactual

[1]Harvard University, MA, USA. Correspondence to: Hao Wang <hao_wang@g.harvard.edu>, Berk Ustun <berk@seas.harvard.edu>, Flavio P. Calmon <flavio@seas.harvard.edu>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

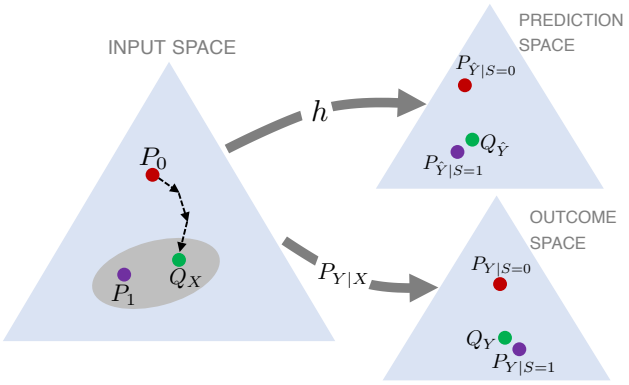

Figure 1: Illustration of probability distributions affecting the disparate impact of a fixed classification model h. Here, P_0 and P_1 denote the distributions of input variables for groups where $S = 0$ and $S = 1$, respectively. Disparate impact is a function of the distributions of predicted outcomes ($P_{\hat{Y}|S=0}$, $P_{\hat{Y}|S=1}$) and true outcomes ($P_{Y|S=0}$, $P_{Y|S=1}$). A *counterfactual distribution* Q_X is a perturbation of P_0 that minimizes a specific measure of disparity. If disparate impact persists under Q_X, there may be irreconcilable differences between the groups (i.e., $P_{Y|X,S=0} \neq P_{Y|X,S=1}$, see Prop. 1). The counterfactual distribution may not be unique, as illustrated by the shaded ellipse.

distribution using a descent procedure in the simplex of probability distributions. We prove that influence functions can be used to compute a gradient in this setting, and derive closed-form estimators that enable efficient computation of influence functions for several (group) fairness criteria.

3. We design pre-processing methods that use counterfactual distributions to repair a black-box classifier in a deployment population without the need to train a new model. The proposed method only affects individuals in a specific group in a way that improves their outcomes (on average).

4. We validate our procedure by repairing classifiers trained with real-world datasets. Our results demonstrate how counterfactual distributions can help mitigate disparate impact in real-world applications.

Use Cases Our approach provides a way to repair classifiers in domains where treatment disparity is legal and ethical[1]. Illustrative use cases include:

- A hospital purchases a readmissions prediction model that satisfies equal opportunity for different genders in the training data but violates it in deployment;
- A rural clinic purchases a classification model to detect bone fractures in x-rays and discovers that patients with a certain physical trait have high FPR;
- A bank enters a new market and discovers its credit score underperforms on customers over 60 years of age.

The tools in this paper will be able to scrutinize and repair performance disparities in all three settings — regardless of whether the model directly uses the sensitive attribute. More importantly, they will be able to achieve parity-based notions of fairness in a way that: (i) only affects one group; (ii) benefits the group it affects (on average); and (iii) incentivizes individuals to reveal their sensitive attributes at prediction time. The latter two points (i.e., do-no-harm and opt-in) are important elements of ethical treatment disparity (see e.g., Lipton et al., 2018; Ustun et al., 2019a).

Related Work We develop a theoretical framework that is used to design methods to determine counterfactual distributions in practice. We then use counterfactual distributions to design optimal transport-based pre-processing methods for ensuring fairness. In this regard, the closest work to ours are those of Feldman et al. (2015); Johndrow & Lum (2017); Del Barrio et al. (2019), which propose methods to control specific disparate impact metrics via optimal transport. These methods differ from ours in that they (i) focus on reducing measures of disparity related to predicted outcomes; (ii) map the input variable distributions across *all* sensitive groups to a common distribution. More broadly, our approach differs from other model-agnostic approaches to mitigate disparate impact (e.g., pre-processing methods such as Kamiran & Calders, 2012; Calmon et al., 2017) in that it does not require access to the training data, and does not require training a new model.

The term "counterfactual distribution" is often used to describe different kinds of hypothetical effects. In the statistics and economics literature (see e.g., Balke & Pearl, 1994; DiNardo et al., 1995; Rubin, 2005; Fortin et al., 2011; Chernozhukov et al., 2013; Johansson et al., 2016; Peters et al., 2017; Fisher & Kennedy, 2018), a counterfactual distribution refers to a hypothetical distribution of an *outcome variable* given a specific distribution of input variables (e.g., the distribution of wages (outcome variable) for young workers if young workers had the same qualifications as older workers). The counterfactual distribution in this work describes a different kind of effect — i.e., a distribution of input variables to minimize disparate impact — and, consequently, must be derived using a different set of tools.

Additional Resources We provide a software implementation of our tools at http://github.com/ustunb/ctfdist. This paper extends work that was first presented at the NeurIPS WESGAI Workshop (Wang et al., 2018b).

[1] Treatment disparity is not illegal nor unethical in all applications of machine learning where fairness is important. In medicine, for example, it is acceptable to use sensitive attributes in prediction. In lending, the Equal Credit Opportunity Act permits the use of age in credit scoring models.

2. Framework

In this section, we formally define counterfactual distributions and discuss their properties.

Preliminaries We consider a standard classification task where the goal is to predict a binary outcome variable $Y \in \{0,1\}$ using a vector of input variables $X = (X_1, \ldots, X_d) \in \mathcal{X}$ drawn from the probability distribution P_X. We are given a black-box classifier $h : \mathcal{X} \to [0,1]$. We assume that $h(\boldsymbol{x}) \in \{0,1\}$ if the classifier outputs a predicted outcome (e.g., SVM) and $h(\boldsymbol{x}) \in [0,1]$ if it outputs a predicted probability (e.g., logistic regression).

We evaluate differences in the performance of the classifier with respect to a *sensitive attribute* $S \in \{0,1\}$ with distribution P_S. We refer to the subset of individuals with $S = 0$ and $S = 1$ as the *target* and *baseline* groups, respectively. We denote their distributions of input variables as $P_0 \triangleq P_{X|S=0}$, and $P_1 \triangleq P_{X|S=1}$. Likewise, we let $P_{\hat{Y}|S=0}(1) \triangleq \mathbb{E}\left[h(X)|S=0\right]$ and $P_{\hat{Y}|S=1}(1) \triangleq \mathbb{E}\left[h(X)|S=1\right]$. For the sake of clarity, we assume that S is not an input variable to the model. Note, however, that our tools and results can be extended to settings where S an input variable for the classifier.

Disparity Metrics We measure the performance disparity between groups in terms of a *disparity metric*. Formally, a disparity metric is a mapping $\mathsf{M} : \mathcal{P} \to \mathbb{R}$ where $\mathcal{P}$ is the set of probability distributions over $\mathcal{X}$. We provide examples of $\mathsf{M}(P_0)$ for common fairness criteria in Table 1. Note that we write disparity metrics as $\mathsf{M}(P_0)$ since they can be expressed as a function P_0 once the classifier and the distributions $P_{Y|X,S}$, P_1, and P_S are fixed.[2]

Counterfactual Distributions A *counterfactual distribution* is a hypothetical probability distribution of input variables for the target group that minimizes a specific disparity metric.

Definition 1. A counterfactual distribution Q_X is a distribution of input variables for the target group such that:

$$Q_X \in \operatorname*{argmin}_{Q'_X \in \mathcal{P}} \left|\mathsf{M}(Q'_X)\right|, \tag{1}$$

where $\mathsf{M}(\cdot)$ is a given disparity metric and $\mathcal{P}$ is the set of probability distributions over $\mathcal{X}$.

There exist several ways to resolve the performance disparity of a fixed classifier by perturbing the distributions of input variables of sensitive groups. For example, one could simultaneously perturb the input distributions for all groups to a "midpoint" distribution (see e.g., the distributions considered by Feldman et al., 2015; Johndrow & Lum, 2017; Del Barrio et al., 2019, to achieve statistical parity).

While our tools could recover such distributions, we will purposely consider a counterfactual distribution that alters the input variables for a sensitive group that attains the less favorable performance (i.e., the target group $S = 0$). This choice reflects our desire to resolve the performance disparity by having the target group perform better, rather than having the baseline group perform worse. As we discuss later, this choice reduces the data requirements to estimate the counterfactual distribution and the individuals who are affected by the repair (i.e., this approach only produces a preprocessor that affects individuals where $S = 0$).

At this point, an observant reader may wonder why a counterfactual distribution for the target group is not simply the distribution of input variables over the baseline group (i.e., $Q_X \equiv P_1$). In fact, the distribution of input variables for the baseline group P_1 *is not necessarily* a counterfactual distribution when $P_{Y|X,S=0} \neq P_{Y|X,S=1}$. We illustrate this point with the following example.

Example 1. Consider a classification task where the input variables $X = (X_1, X_2) \in \{0,1\}^2$ are drawn from distributions such that $P_{X|S=s} = P_{X_1|S=s} \cdot P_{X_2|S=s}$ for $s \in \{0,1\}$ where:

$$\Pr(X_1 = 1|S=0) = 0.9,\ \Pr(X_2 = 1|S=0) = 0.2,$$
$$\Pr(X_1 = 1|S=1) = 0.1,\ \Pr(X_2 = 1|S=1) = 0.5.$$

Assume that the true outcome variables Y are drawn from the conditional distributions:

$$\begin{aligned} P_{Y|X,S=0}(1|\boldsymbol{x}) &= \mathsf{logistic}(2x_1 - 2x_2), \\ P_{Y|X,S=1}(1|\boldsymbol{x}) &= \mathsf{logistic}(2x_1 + 4x_2 - 3). \end{aligned} \tag{2}$$

In this case, the Bayes optimal classifier for $S = 1$ is $h(\boldsymbol{x}) = \mathbb{I}[x_2 = 1]$. Using the difference in FPR as the disparity metric, h achieves $\mathsf{M}(P_0) = 25.1\%$. In this case, setting $P_0 \leftarrow P_1$ would achieve a disparity of $\mathsf{M}(P_1) = 43.6\%$. In contrast, we can achieve a disparity metric of $\mathsf{M}(Q_X) = 0.0\%$ for a counterfactual distribution such that

$$\begin{aligned} Q_X(0,0) &= 0.50, & Q_X(0,1) &= 0.09, \\ Q_X(1,0) &= 0.41, & Q_X(1,1) &= 0.00. \end{aligned}$$

Example 1 shows that counterfactual distributions may be non-trivial when the conditional distributions of Y given X differ across groups (i.e., $P_{Y|X,S=0} \neq P_{Y|X,S=1}$). In particular, the condition $P_{Y|X,S=0} \neq P_{Y|X,S=1}$ will always hold whenever counterfactual distributions do not completely eliminate the disparity between groups. We formalize this statement in the next proposition.

[2] As we show in Appendix A.1, the disparity metric can be expressed as a function of P_0 once the following objects are fixed: h, the classifier; $P_{Y|X,S}$ the distribution of the true outcomes given input variables and sensitive attribute; P_1, the distribution of input variables over the baseline group; and P_S the distribution of the sensitive attribute.

PERFORMANCE METRIC	ACRONYM	DISPARITY METRIC
Statistical Parity	SP	$\Pr(\hat{Y}=0\|S=0) - \Pr(\hat{Y}=0\|S=1)$
False Discovery Rate	FDR	$\Pr(Y=0\|\hat{Y}=1,S=0) - \Pr(Y=0\|\hat{Y}=1,S=1)$
False Negative Rate	FNR	$\Pr(\hat{Y}=0\|Y=1,S=0) - \Pr(\hat{Y}=0\|Y=1,S=1)$
False Positive Rate	FPR	$\Pr(\hat{Y}=1\|Y=0,S=0) - \Pr(\hat{Y}=1\|Y=0,S=1)$
Distribution Alignment	DA_λ	$\text{D}_{\text{KL}}(P_{\hat{Y}\|S=0} \| P_{\hat{Y}\|S=1}) + \lambda \text{D}_{\text{KL}}(P_0 \| P_1)$

Table 1: Disparity metrics $\mathsf{M}(P_0)$ for common fairness criteria (see e.g., Romei & Ruggieri, 2014, for a list). We assume that $S=0$ attains the less favorable value of performance so that $\mathsf{M}(P_0) \geq 0$. Our tools can be used for any fairness criterion that can be expressed as a convex combination of these metrics (e.g., equalized odds; see Proposition 3). *Distribution Alignment* is a metric proposed in Wang et al. (2018a) that measures the disparity of predicted outcomes via the KL-divergence.

Proposition 1. *If $\mathsf{M}(Q_X) > 0$ where Q_X is a counterfactual distribution for a disparity metric in Table 1, then $P_{Y|X,S=0} \neq P_{Y|X,S=1}$.*

Proposition 1 illustrates how a counterfactual distribution can be used to detect cases where a classifier exhibits an irreconcilable performance disparity between groups – i.e., a disparity that cannot be resolved by perturbing the distributions of input variables for the target group. The result complements various impossibility results on inevitable trade-offs between groups (see e.g., Chouldechova, 2017; Kleinberg et al., 2016; Pleiss et al., 2017). It also provides a sufficient condition that can inform the need for treatment disparity (see e.g., of Kleinberg et al., 2018; Dwork et al., 2018; Lipton et al., 2018; Ustun et al., 2019a).

3. Methodology

In this section, we present information-theoretic tools to learn counterfactual distributions from data. We first demonstrate how influence functions provide a natural "descent direction" in this setting. Next, we derive closed-form expressions for the influence functions of the disparity metrics in Table 1. Lastly, we present a descent procedure that combines these results to learn a counterfactual distribution for the deployment population.

3.1. Measuring the Descent Direction

In what follows, we describe how to reduce the value of a disparity metric by perturbing the distribution of input variables over the target group P_0.

We start by formally defining the local perturbation of an input distribution.

Definition 2. The perturbed distribution $\widetilde{P}_0$ over the target group ($S=0$) is given by

$$\widetilde{P}_0(\boldsymbol{x}) \triangleq P_0(\boldsymbol{x})(1+\epsilon f(\boldsymbol{x})), \quad \forall \boldsymbol{x} \in \mathcal{X} \tag{3}$$

where $f : \mathcal{X} \to \mathbb{R}$ is a perturbation function from the class of all functions with zero mean and unit variance w.r.t. P_0, and $\epsilon > 0$ is a positive scaling constant chosen so that $\widetilde{P}_0$ is a valid probability distribution.

Here, $f(\boldsymbol{x})$ represents a direction in the probability simplex while ϵ represents the magnitude of perturbation (see e.g., Borade & Zheng, 2008; Anantharam et al., 2013; Huang et al., 2014, for other applications of local perturbations of measures in information theory).

As we will see shortly, the direction of steepest descent for disparate impact can be measured using an *influence function* (see e.g., Huber, 2011; Koh & Liang, 2017, for other uses in machine learning).

Definition 3. For a disparity metric $\mathsf{M}(\cdot)$, the influence function $\psi : \mathcal{X} \to \mathbb{R}$ is given by

$$\psi(\boldsymbol{x}) \triangleq \lim_{\epsilon \to 0} \frac{\mathsf{M}\left((1-\epsilon)P_0 + \epsilon \delta_{\boldsymbol{x}}\right) - \mathsf{M}(P_0)}{\epsilon} \tag{4}$$

where $\delta_{\boldsymbol{x}}(\boldsymbol{z}) = \mathbb{I}[\boldsymbol{z} = \boldsymbol{x}]$ is the delta function at $\boldsymbol{x}$.

Intuitively, given a sufficiently large dataset from the deployment population, the influence function approximates the change in a disparity metric when a sample $\boldsymbol{x} \in \mathcal{X}$ from the target group is removed (or added) to the dataset.

In Proposition 2, we show that perturbing the distribution P_0 along the direction defined by $-\psi(\boldsymbol{x})$ produces the largest local decrease of the disparity metric. That is, $-\psi(\boldsymbol{x})$ reflects the direction of steepest descent in disparate impact.

Proposition 2. *Given a disparity metric $\mathsf{M}(\cdot)$, we have that*

$$\operatorname*{argmin}_{f(\boldsymbol{x})} \lim_{\epsilon \to 0} \frac{\mathsf{M}(\widetilde{P}_0) - \mathsf{M}(P_0)}{\epsilon} = \frac{-\psi(\boldsymbol{x})}{\sqrt{\mathbb{E}\left[\psi(X)^2 | S=0\right]}}, \tag{5}$$

for any influence function $\psi : \mathcal{X} \to \mathbb{R}$ such that $\mathbb{E}\left[\psi(X)^2 | S=0\right] \neq 0$.

Proposition 3 shows that when disparity is measured using a linear combination of metrics, the influence function for the compound metric can be expressed as a linear combination of the influence functions for its components.

Proposition 3. *Given any convex combination of K disparity metrics $\mathsf{M}(P_0) = \sum_{i=1}^{K} \lambda_i \mathsf{M}_i(P_0)$, the influence function of the compound disparity metric $\mathsf{M}(P_0)$ has the form:*

$$\psi(\boldsymbol{x}) = \sum_{i=1}^{K} \lambda_i \psi_i(\boldsymbol{x}). \tag{6}$$

Proposition 3 allows us to consider a larger class of disparity measures than those shown in Table 1. For instance, one can recover a counterfactual distribution to achieve equalized odds (Hardt et al., 2016) by using a convex combination of influence functions for FPR and FNR.

3.2. Computing Influence Functions

We now present closed-form expressions for the influence functions of disparity metrics shown in Table 1. The expressions will be cast in terms of three classifiers:

- $h(\boldsymbol{x})$: the black-box classifier that we aim to repair;
- $\hat{s}(\boldsymbol{x})$: a classifier that uses the same input variables as h, but aims to predict the probability that an individual belongs to the baseline group, $P_{S|X}(1|\boldsymbol{x})$.
- $\hat{y}_0(\boldsymbol{x})$: a classifier that uses the same input variables as h, but aims to predict the true outcome *for individuals in the target group*, $P_{Y|X,S=0}(1|\boldsymbol{x})$.

Given $h(\boldsymbol{x})$, we would train $\hat{s}(\boldsymbol{x})$ and $\hat{y}_0(\boldsymbol{x})$ using an *auditing dataset* $\mathcal{D}^{\text{audit}} = \{(\boldsymbol{x}_i, y_i, s_i)\}_{i=1}^{n}$ drawn from the deployment population. With these three models in hand, we can then compute influence functions using closed-form expressions shown in the following proposition.

Proposition 4. *The influence functions for the disparity metrics in Table 1 can be expressed as*

$$\psi^{SP}(\boldsymbol{x}) = -h(\boldsymbol{x}) + \hat{\mu}_0,$$

$$\psi^{FDR}(\boldsymbol{x}) = \frac{h(\boldsymbol{x})(1-\hat{y}_0(\boldsymbol{x})) - \nu_{0,1} h(\boldsymbol{x})}{\hat{\mu}_0},$$

$$\psi^{FNR}(\boldsymbol{x}) = \frac{(1-h(\boldsymbol{x}))\hat{y}_0(\boldsymbol{x}) - \gamma_{0,1}\hat{y}_0(\boldsymbol{x})}{\mu_0},$$

$$\psi^{FPR}(\boldsymbol{x}) = \frac{h(\boldsymbol{x})(1-\hat{y}_0(\boldsymbol{x})) - \gamma_{1,0}(1-\hat{y}_0(\boldsymbol{x}))}{(1-\mu_0)},$$

$$\psi^{DA}(\boldsymbol{x}) = \log\frac{\hat{\mu}_0(1-\hat{\mu}_1)}{(1-\hat{\mu}_0)\hat{\mu}_1} h(\boldsymbol{x}) + \lambda \log\frac{1-\hat{s}(\boldsymbol{x})}{\hat{s}(\boldsymbol{x})} - \hat{\mu}_0 \log\frac{\hat{\mu}_0(1-\hat{\mu}_1)}{(1-\hat{\mu}_0)\hat{\mu}_1} - \lambda \mathbb{E}\left[\log\frac{1-\hat{s}(X)}{\hat{s}(X)} \middle| S=0\right],$$

where μ_s, $\hat{\mu}_s$, $\gamma_{a,b}$, and $\nu_{a,b}$ are constants such that

$$\mu_s \triangleq \Pr(Y=1|S=s),$$
$$\hat{\mu}_s \triangleq \Pr(\hat{Y}=1|S=s),$$
$$\gamma_{a,b} \triangleq \Pr(\hat{Y}=a|Y=b,S=0),$$
$$\nu_{a,b} \triangleq \Pr(Y=a|\hat{Y}=b,S=0).$$

3.3. Learning Counterfactual Distributions from Data

So far we have shown that influence functions can be used to evaluate the direction of steepest descent of a disparity metric (Proposition 2), and that the value of an influence function can be estimated using data from the deployment population (Proposition 4).

Considering these results, one would expect that disparity could be minimized by repeatedly (i) perturbing the distribution in the direction of steepest descent (5), and (ii) estimating the influence function at the new, perturbed distribution. Repeating these steps, we would recover an approximate solution to (1) – i.e., an approximate counterfactual distribution.

In Algorithm 1, we formalize this intuition by presenting a descent procedure to recover a counterfactual distribution for a given disparity metric $\mathsf{M}(\cdot)$. The procedure is analogous to stochastic gradient descent in the space of distributions over $\mathcal{X}$, where the resampling at each iteration corresponds to a gradient step.

Algorithm 1 Distributional Descent.

Input:
$h : \mathcal{X} \to [0,1]$ ▷ classification model
$\mathcal{D} = \{(\boldsymbol{x}_i, y_i, s_i)\}_{i=1}^{n}$ ▷ data from deployment population
$\mathsf{M}(\cdot)$ ▷ disparity metric
$\epsilon > 0$ ▷ step size
Initialize
$I_0 \leftarrow \{i = 1, \ldots, n \mid s_i = 0\}$
$\mathcal{D}_0 \leftarrow (\boldsymbol{x}_i, y_i)$ for $i \in I_0$ ▷ samples where $s_i = 0$
$\mathcal{D}_1 \leftarrow (\boldsymbol{x}_i, y_i)$ for $i \notin I_0$ ▷ samples where $s_i \neq 0$
$\boldsymbol{w}_0 \leftarrow [w_i]_{i \in I_0}$ where $w_i = 1.0$ ▷ initialize weights
$M \leftarrow \mathsf{M}(\mathcal{D}_0 \cup \mathcal{D}_1)$ ▷ evaluate disparity metric
repeat
  $M^{\text{old}} \leftarrow M$
  $\psi_i \leftarrow \psi(\boldsymbol{x}_i)$ for $i \in I_0$ ▷ compute $\psi(\boldsymbol{x}_i)$ for points in $\mathcal{D}_0$
  $w_i \leftarrow (1 - \epsilon\psi_i) \cdot w_i$ for $i \in I_0$
  $\widetilde{\mathcal{D}}_0 \leftarrow \textsc{Resample}(\mathcal{D}_0, \boldsymbol{w}_0)$
  $M \leftarrow \mathsf{M}(\widetilde{\mathcal{D}}_0 \cup \mathcal{D}_1)$ ▷ evaluate disparity metric
until $M \geq M^{\text{old}}$
return: $\boldsymbol{w}_0, \widetilde{\mathcal{D}}_0$ ▷ samples from counterfactual distribution

procedure RESAMPLE($\mathcal{D}$, $\boldsymbol{w}$)
  return: $|\mathcal{D}|$ points sampled from $\mathcal{D}$ using the weights $\boldsymbol{w}$
end procedure

Given a classifier h and a dataset $\{(\boldsymbol{x}_i, y_i, s_i)\}_{i=1}^{n}$ from the distribution $P_{X,Y,S}$, the procedure outputs a dataset drawn from a counterfactual distribution.

The procedure pairs each point with a *sampling weight* w_i, which is initialized as $w_i = 1.0$. At each iteration, it first computes the value of the influence function $\psi(\boldsymbol{x})$. Next, it updates the values of each sampling weight for each point in the target group as $(1 - \epsilon\psi(\boldsymbol{x}_i)) \cdot w_i$, where ϵ is a

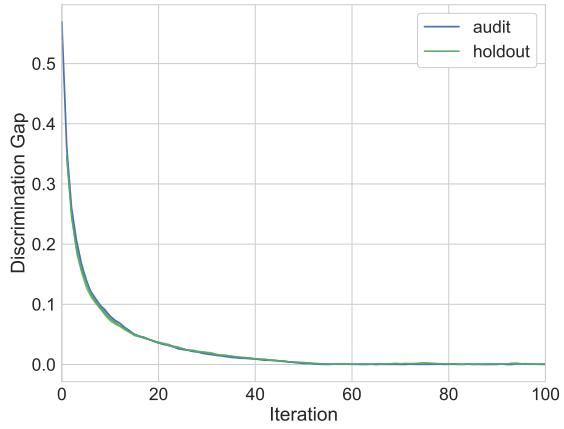

Figure 2: Values of FPR for auditing dataset (blue) and holdout dataset (green), respectively, with each iteration in distributional descent for a synthetic dataset. Here, the procedure converges to a counterfactual distribution in 50 iterations. We show additional steps for the sake of illustration.

user-specified step size parameter. The updated sampling weights represent the direction in which the distribution for the target group should be perturbed to reduce $\mathsf{M}(\cdot)$. The data points from the target group are then resampled with their sampling weights. The set of resampled points mimics one drawn from the perturbed distribution.

The procedure determines if the classifier still has disparate impact at the end of each iteration by computing the value of $\mathsf{M}(\cdot)$ on the set of resampled points. These steps are repeated until $\mathsf{M}(\cdot)$ ceases to decrease. Once the procedure stops, it outputs: (i) dataset drawn from a counterfactual distribution; (ii) a set of sampling weights for each point from the target group, which can be used to draw samples from the counterfactual distribution.

In Figure 2, we show the progress (and convergence) of Algorithm 1 when recovering a counterfactual distribution for a synthetic dataset described in Appendix B.1.

4. Model Repair

In this section, we describe how to use counterfactual distributions to repair classifiers that exhibit disparate impact.

Preprocessor Given a classifier $h(\boldsymbol{x})$, we aim to mitigate disparate impact by constructing a *preprocessor* $T : \mathcal{X} \to \mathcal{X}$ that alters the features of the target group. Thus, the *repaired classifier* $\widetilde{h}(\boldsymbol{x})$ will operate as:

$$\widetilde{h}(\boldsymbol{x}) = \begin{cases} h(T(\boldsymbol{x})) & \text{if } s = 0, \\ h(\boldsymbol{x}) & \text{otherwise.} \end{cases} \tag{7}$$

The preprocessor is a (potentially randomized) mapping that transforms the distribution of samples over the target population into the counterfactual distribution, i.e., given a random variable X drawn from the target population distribution, the distribution of $T(X)$ will approximate counterfactual distribution.

Optimal Transport We produce the preprocessor by solving an optimal transport problem. To this end, we require the following inputs:

- $\mathcal{D}_0$, which represents the original samples for the target group. We assume $\mathcal{D}_0$ contains n_0 samples, of which m are distinct: $\{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_m\}$.
- $\widetilde{\mathcal{D}}_0$, which represents the samples drawn from the counterfactual distribution (i.e., the data produced via resampling in Algorithm 1). We assume that $\widetilde{\mathcal{D}}_0$ contains $\widetilde{n}_0$ samples, of which $\widetilde{m}$ are distinct: $\{\widetilde{\boldsymbol{x}}_1, \cdots, \widetilde{\boldsymbol{x}}_{\widetilde{m}}\}$.

With these samples at hand, we formulate an optimal transport problem of the form:

$$\min_{\gamma_{ij} \in \mathbb{R}^+} \quad \sum_{i=1}^{m} \sum_{j=1}^{\widetilde{m}} C_{ij} \gamma_{ij} \tag{8a}$$

$$\text{s.t.} \quad \sum_{j=1}^{\widetilde{m}} \gamma_{ij} = p_i \quad i = 1, \cdots, m \tag{8b}$$

$$\sum_{i=1}^{m} \gamma_{ij} = q_j \quad j = 1, \cdots, \widetilde{m}. \tag{8c}$$

Here, C_{ij} represents the cost of altering the input variables from $\boldsymbol{x}_i$ to $\widetilde{\boldsymbol{x}}_j$ given a user-specified *cost function* that we will discuss shortly; p, q are the empirical estimates of P_0 and Q_X, respectively,

$$p_i = \frac{1}{n_0} \sum_{\boldsymbol{x} \in \mathcal{D}_0} \delta_{\boldsymbol{x}_i}(\boldsymbol{x}), \quad q_j = \frac{1}{\widetilde{n}_0} \sum_{\boldsymbol{x} \in \widetilde{\mathcal{D}}_0} \delta_{\widetilde{\boldsymbol{x}}_j}(\boldsymbol{x});$$

The optimal transport problem in (8) is a standard linear program that aims to find a *coupling* of p and q, γ (see e.g., Peyré & Cuturi, 2017; Villani, 2008). Formally, a coupling is a joint probability distribution with marginal distributions specified by p and q. Given the minimal-cost coupling γ^*, one can construct a (randomized) preprocessor $T(\cdot)$ which takes a sample $\boldsymbol{x}_i$ and returns an altered sample $\widetilde{\boldsymbol{x}}_j$ with probability γ^*_{ij}/p_i.

We note that the linear programming formulation in (8) is designed for settings with discrete input distributions. In settings when the distributions P_0 and Q_X are continuous, an analogous optimal transport problem can be formulated and solved with other approaches (see e.g., Benamou & Brenier, 2000; Angenent et al., 2003).

Choice of Cost Function The cost function C_{ij} controls how samples of the target group are perturbed. By default, one could use a standard distance metric such as the L_2-norm (e.g., Feldman et al., 2015; Johndrow & Lum, 2017; Del Barrio et al., 2019). However, one could also consider additional criteria to fine-tune the mapping specified by $T(\cdot)$. For example, one can specify a cost function that avoids specific kinds of change by setting the value of C_{ij} to a large constant so as to penalize undesirable mappings (e.g., a mapping that would alter immutable attributes such as marital status Ustun et al., 2019b).

Customization via Constraints Users can also fine-tune the behavior of the preprocessor by adding custom constraints to the feasible region of optimal transport problems as in (8). For example, one can impose constraints on *individual fairness* to ensure that the repaired classifier will "treat similar individuals similarly" (see e.g., Dwork et al., 2012). This behavior could be induced by including constraints of the form:

$$\frac{1}{2}\sum_{j=1}^{\widetilde{m}}\left|\frac{\gamma_{ij}}{p_i}-\frac{\gamma_{lj}}{p_l}\right| \leq \mathsf{d}(\boldsymbol{x}_i,\boldsymbol{x}_l) \quad \text{for all } i,l \in [m].$$

Here, the LHS is the total-variation distance (Cover & Thomas, 2012) between the distributions of $T(\boldsymbol{x}_i)$ and $T(\boldsymbol{x}_l)$, and $\mathsf{d}(\cdot,\cdot) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ is a distance metric that reflects the similarity between samples.

5. Experiments

In this section, we demonstrate how counterfactual distributions can be used to avoid disparate impact for classifiers on real-world datasets. We include all datasets and scripts to reproduce our results at http://github.com/ustunb/ctfdist.

Setup We aim to recover counterfactual distributions for different disparity metrics in Table 1. To this end, we consider processed versions of the `adult` dataset (Bache & Lichman, 2013) and the ProPublica `compas` dataset (Angwin et al., 2016).

For each dataset, we use:

- 30% of samples to train a classifier $h(\boldsymbol{x})$ to repair;
- 50% of samples to recover a counterfactual distribution via Algorithm 1;
- 20% of samples as a hold-out set to evaluate the performance of the repaired model.

We use ℓ_2-logistic regression to train a classifier $h(\boldsymbol{x})$ as well as the classifiers $\hat{y}_0(\boldsymbol{x})$ and $\hat{s}(\boldsymbol{x})$ that we use to estimate the influence functions in Algorithm 1. We tune the parameters and estimate the performance of each classifiers using a standard 10-fold CV setup.

Our setup assumes that the data used to train and repair the model are drawn from the same distribution, which may not be the case in settings with dataset shift. Our setup also differs from real-world settings in that we use 70% of the samples in each dataset to estimate the counterfactual distribution. In practice, however, we would use all available samples since we would be given the classifier to repair.

Discussion In Table 2, we show the effectiveness of preprocessors built to mitigate different kinds of disparity for the classifiers trained on the `adult` and `compas` datasets.

We build each preprocessor as follows. We first resample data from the target population according to Algorithm 1. This outputs a dataset of samples drawn from the counterfactual distribution. Next, we use the resampled dataset to produce an empirical estimate of the counterfactual distribution Q_X. This distribution is then used to obtain the preprocessor by solving a version of (8) with the cost function $C_{ij} = \|\boldsymbol{x}_i - \widetilde{\boldsymbol{x}}_j\|_2^2$.

As shown, the approach reduces disparate impact in the target group, while having a minor effect on test accuracy at various decision points across the full ROC curve. Counterfactual distributions provide a way to scrutinize this mapping in greater detail. As shown in Table 3, one can visualize the differences between the observed distribution and counterfactual distribution to understand how the input variable distributions are altered to reduce disparity.

This kind of constrastive analysis may be helpful in understanding the factors that produce performance disparities in the first place. For example, the differences between the observed distribution P_0 and the counterfactual distribution Q_X could be used to identify prototypical samples (see e.g, Bien & Tibshirani, 2011; Kim et al., 2016), or to score features in terms of their ability to produce disparities in the deployment population (Datta et al., 2016; 2017; Adler et al., 2018).

6. Conclusion

We have introduced a new distributional paradigm to understand and mitigate disparate impact. Our framework is based on counterfactual distributions, which can be efficiently computed given a fixed model and data from a population of interest. The tools in this work apply to binary classification models. However, our approach can be extended to handle other kinds of supervised learning models.

Limitations Our approach requires collecting data on sensitive attribute, which may infringe privacy (though this is difficult to avoid as discussed in Žliobaitė & Custers, 2016).

			Original Model			Repaired Model		Target Group AUC	
Dataset	Metric	Target Group	Baseline Group	Target Group	**Disc. Gap**	Target Group	**Disc. Gap**	Before Repair	After Repair
adult	SP	Female	0.696	0.874	**0.178**	0.688	**-0.007**	0.895	0.758
adult	FNR	Female	0.478	0.639	**0.161**	0.483	**0.004**	0.895	0.880
adult	FPR	Male	0.021	0.119	**0.098**	0.023	**0.002**	0.829	0.714
compas	SP	White	0.514	0.594	**0.079**	0.533	**0.018**	0.704	0.667
compas	FNR	White	0.350	0.487	**0.137**	0.439	**0.088**	0.704	0.699
compas	FPR	Non-white	0.190	0.278	**0.087**	0.160	**-0.029**	0.732	0.680

Table 2: Change in disparate impact for classification models for `adult` and `compas` when paired with a randomized preprocessor built to mitigate different kinds of disparity. Each row shows the value of a specific performance metric for the classifier over the target and baseline groups (e.g., SP, FNR, and FPR). The target group is defined as the group that attains the less favorable value of the performance metric. The preprocessor aims to reduce to difference in performance metric by randomly perturbing the input variables for individuals in the target group. We also include AUC to show the change in performance due to the randomized preprocessor. All values are computed using a hold-out sample that is not used to train the model or build the preprocessor.

	Observed		Counterfactual		
	Female	Male	SP Female	FNR Female	FPR Male
Married	18%	63%	39%	23%	54%
Immigrant	10%	11%	11%	11%	12%
HighestDegree_is_HS	32%	32%	24%	28%	37%
HighestDegree_is_AS	7%	8%	9%	9%	6%
HighestDegree_is_BS	15%	18%	21%	17%	13%
HighestDegree_is_MSorPhD	6%	7%	13%	8%	5%
AnyCapitalLoss	3%	5%	8%	5%	4%
Age ≤ 30	39%	29%	29%	38%	35%
WorkHrsPerWeek<40	38%	17%	33%	37%	19%
JobType_is_WhiteCollar	34%	19%	36%	35%	15%
JobType_is_BlueCollar	5%	34%	4%	5%	39%
JobType_is_Specialized	23%	21%	29%	23%	20%
JobType_is_ArmedOrProtective	1%	2%	1%	1%	3%
Industry_is_Private	73%	69%	64%	69%	70%
Industry_is_Government	15%	12%	22%	17%	12%
Industry_is_SelfEmployed	5%	15%	8%	6%	13%

Table 3: Counterfactual distributions produced using Algorithm 1 for a classifier on `adult`. We observe that different metrics produce different counterfactual distributions. By comparing the distribution of the target group with the counterfactual distribution, we can evaluate how the repaired classifier will perturb their features to reduce disparity.

Our approach also aims to mitigate performance using a randomized preprocessor. While randomization is a common technique to reduce disparate impact in the literature (see e.g., Hardt et al., 2016; Agarwal et al., 2018), it may not be practical in applications such as loan approval since an applicant could achieve a different predicted outcome by applying multiple times. Some effects of randomization can be mitigated by heuristic strategies. Given that we have considered counterfactual distributions that improve the performance for the target group, however, our approach does have a benefit in that randomization will only apply to individuals in the target group and only be applied in a way that will improve their outcomes.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. CCF-18-45852.

References

Adler, P., Falk, C., Friedler, S. A., Nix, T., Rybeck, G., Scheidegger, C., Smith, B., and Venkatasubramanian, S. Auditing black-box models for indirect influence. *Knowledge and Information Systems*, 54(1):95–122, 2018.

Agarwal, A., Beygelzimer, A., Dudík, M., Langford, J., and Wallach, H. A reductions approach to fair classification. *arXiv preprint arXiv:1803.02453*, 2018.

Anantharam, V., Gohari, A., Kamath, S., and Nair, C. On maximal correlation, hypercontractivity, and the data processing inequality studied by Erkip and Cover. *arXiv preprint arXiv:1304.6133*, 2013.

Angenent, S., Haker, S., and Tannenbaum, A. Minimizing flows for the monge–kantorovich problem. *SIAM journal on mathematical analysis*, 35(1):61–97, 2003.

Angwin, J., Larson, J., Mattu, S., and Kirchner, L. Machine bias. *ProPublica*, 2016.

Bache, K. and Lichman, M. UCI Machine Learning Repository, 2013.

Balke, A. and Pearl, J. Counterfactual probabilities: Computational methods, bounds and applications. In *Proceedings of the Tenth international conference on Uncertainty in artificial intelligence*, pp. 46–54. Morgan Kaufmann Publishers Inc., 1994.

Barocas, S. and Selbst, A. D. Big data's disparate impact. *Cal. L. Rev.*, 104:671, 2016.

Benamou, J.-D. and Brenier, Y. A computational fluid mechanics solution to the monge-kantorovich mass transfer problem. *Numerische Mathematik*, 84(3):375–393, 2000.

Bien, J. and Tibshirani, R. Prototype selection for interpretable classification. *The Annals of Applied Statistics*, pp. 2403–2424, 2011.

Borade, S. and Zheng, L. Euclidean information theory. In *IEEE International Zurich Seminar on Communications*, pp. 14–17. IEEE, 2008.

Calmon, F., Wei, D., Vinzamuri, B., Ramamurthy, K. N., and Varshney, K. R. Optimized pre-processing for discrimination prevention. In *Advances in Neural Information Processing Systems*, pp. 3992–4001, 2017.

Canetti, R., Cohen, A., Dikkala, N., Ramnarayan, G., Scheffler, S., and Smith, A. From soft classifiers to hard decisions: How fair can we be? In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 309–318. ACM, 2019.

Chen, I., Johansson, F. D., and Sontag, D. Why is my classifier discriminatory? *arXiv preprint arXiv:1805.12002*, 2018.

Chernozhukov, V., Fernández-Val, I., and Melly, B. Inference on counterfactual distributions. *Econometrica*, 81(6):2205–2268, 2013.

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

Corbett-Davies, S., Pierson, E., Feller, A., Goel, S., and Huq, A. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 797–806. ACM, 2017.

Cover, T. M. and Thomas, J. A. *Elements of information theory*. John Wiley & Sons, 2012.

Dastin, J. Amazon scraps secret ai recruiting tool that showed bias against women. *San Fransico, CA: Reuters. Retrieved on October*, 9:2018, 2018.

Datta, A., Sen, S., and Zick, Y. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *2016 IEEE Symposium on Security and Privacy*, pp. 598–617. IEEE, 2016.

Datta, A., Fredrikson, M., Ko, G., Mardziel, P., and Sen, S. Proxy non-discrimination in data-driven systems. *arXiv preprint arXiv:1707.08120*, 2017.

Del Barrio, E., Gamboa, F., Gordaliza, P., and Loubes, J.-M. Obtaining fairness using optimal transport theory. In *International Conference on Machine Learning*, 2019.

DiNardo, J., Fortin, N. M., and Lemieux, T. Labor market institutions and the distribution of wages, 1973-1992: A semiparametric approach. Technical report, National bureau of economic research, 1995.

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

Dwork, C., Immorlica, N., Kalai, A. T., and Leiserson, M. D. Decoupled classifiers for group-fair and efficient machine learning. In *Conference on Fairness, Accountability and Transparency*, pp. 119–133, 2018.

Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., and Venkatasubramanian, S. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 259–268. ACM, 2015.

Fisher, A. and Kennedy, E. H. Visually communicating and teaching intuition for influence functions. *arXiv preprint arXiv:1810.03260*, 2018.

Fortin, N., Lemieux, T., and Firpo, S. Decomposition methods in economics. In *Handbook of labor economics*, volume 4, pp. 1–102. Elsevier, 2011.

Galhotra, S., Brun, Y., and Meliou, A. Fairness testing: testing software for discrimination. In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*, pp. 498–510. ACM, 2017.

Guszcza, J., Rahwan, I., Bible, W., Cebrian, M., and Katyal, V. Why we need to audit algorithms, 2018. URL `https://hbr.org/2018/11/why-we-need-to-audit-algorithms`.

Hardt, M., Price, E., Srebro, N., et al. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pp. 3315–3323, 2016.

Huang, S.-L., Makur, A., Kozynski, F., and Zheng, L. Efficient statistics: Extracting information from iid observations. In *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 699–706. IEEE, 2014.

Huber, P. J. Robust statistics. In *International Encyclopedia of Statistical Science*, pp. 1248–1251. Springer, 2011.

Johansson, F., Shalit, U., and Sontag, D. Learning representations for counterfactual inference. In *International conference on machine learning*, pp. 3020–3029, 2016.

Johndrow, J. E. and Lum, K. An algorithm for removing sensitive information: application to race-independent recidivism prediction. *arXiv preprint arXiv:1703.04957*, 2017.

Kamiran, F. and Calders, T. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012.

Kilbertus, N., Carulla, M. R., Parascandolo, G., Hardt, M., Janzing, D., and Schölkopf, B. Avoiding discrimination through causal reasoning. In *Advances in Neural Information Processing Systems*, pp. 656–666, 2017.

Kim, B., Khanna, R., and Koyejo, O. O. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in Neural Information Processing Systems*, pp. 2280–2288, 2016.

Kleinberg, J., Mullainathan, S., and Raghavan, M. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*, 2016.

Kleinberg, J., Ludwig, J., Mullainathan, S., and Rambachan, A. Algorithmic fairness. In *AEA Papers and Proceedings*, volume 108, pp. 22–27, 2018.

Koh, P. W. and Liang, P. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning*, pp. 1885–1894, 2017.

Kusner, M. J., Loftus, J., Russell, C., and Silva, R. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, pp. 4069–4079, 2017.

Lipton, Z. C., Chouldechova, A., and McAuley, J. Does mitigating ml's impact disparity require treatment disparity? *arXiv preprint arXiv:1711.07076*, 2018.

Menon, A. K. and Williamson, R. C. The cost of fairness in binary classification. In *Conference on Fairness, Accountability and Transparency*, pp. 107–118, 2018.

Peters, J., Janzing, D., and Schölkopf, B. *Elements of causal inference: foundations and learning algorithms*. MIT press, 2017.

Peyré, G. and Cuturi, M. Computational optimal transport. Technical report, Center for Research in Economics and Statistics, 2017.

Pierson, E., Corbett-Davies, S., and Goel, S. Fast threshold tests for detecting discrimination. *arXiv preprint arXiv:1702.08536*, 2017.

Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., and Weinberger, K. Q. On fairness and calibration. In *Advances in Neural Information Processing Systems*, pp. 5684–5693, 2017.

Romei, A. and Ruggieri, S. A multidisciplinary survey on discrimination analysis. *The Knowledge Engineering Review*, 29(5): 582–638, 2014.

Rubin, D. B. Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469):322–331, 2005.

Simoiu, C., Corbett-Davies, S., Goel, S., et al. The problem of infra-marginality in outcome tests for discrimination. *The Annals of Applied Statistics*, 11(3):1193–1216, 2017.

Sugiyama, M., Lawrence, N. D., Schwaighofer, A., et al. *Dataset shift in machine learning*. The MIT Press, 2017.

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

Ustun, B., Spangher, A., and Liu, Y. Actionable recourse in linear classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 10–19. ACM, 2019b.

Villani, C. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

Wang, H., Ustun, B., and Calmon, F. P. On the direction of discrimination: An information-theoretic analysis of disparate impact in machine learning. In *2018 IEEE International Symposium on Information Theory*, pp. 1216–1220. IEEE, 2018a.

Wang, H., Ustun, B., and Calmon, F. P. Avoiding disparate impact with counterfactual distributions. In *NeurIPS Workshop on Ethical, Social and Governance Issues in AI*, 2018b.

Zafar, M. B., Valera, I., Rodriguez, M., Gummadi, K., and Weller, A. From parity to preference-based notions of fairness in classification. In *Advances in Neural Information Processing Systems*, pp. 229–239, 2017.

Žliobaitė, I. Measuring discrimination in algorithmic decision making. *Data Mining and Knowledge Discovery*, 31(4):1060–1089, 2017.

Žliobaitė, I. and Custers, B. Using sensitive personal data may be necessary for avoiding discrimination in data-driven decision models. *Artificial Intelligence and Law*, 24(2):183–201, 2016.