Achieving Fair Treatment in Algorithmic Classification

Andrew Morgan
Cornell University
asmorgan@cs.cornell.edu

Rafael Pass[*]
Cornell Tech
rafael@cornell.edu

September 25, 2018

Abstract

Fairness in classification has become an increasingly relevant and controversial issue as computers replace humans in many of today's classification tasks. In particular, a subject of much recent debate is that of finding, and subsequently achieving, suitable definitions of fairness in an algorithmic context. In this work, following the work of Hardt et al. (NIPS'16), we consider and formalize the task of *sanitizing* an unfair classifier $\mathcal{C}$ into a classifier $\mathcal{C}'$ satisfying an approximate notion of "equalized odds" or *fair treatment*. Our main result shows how to take any (possibly unfair) classifier $\mathcal{C}$ over a *finite outcome space*, and transform it—by just *perturbing* the output of $\mathcal{C}$—according to some distribution learned by just having black-box access to samples of labeled, and previously classified, data, to produce a classifier $\mathcal{C}'$ that satisfies fair treatment; we additionally show that our derived classifier is near-optimal in terms of accuracy. We also experimentally evaluate the performance of our method.

1 Introduction

As algorithmic decision-making becomes ever more popular and widely-used in today's society, concerns are being raised about whether, and to what extent, algorithms have the potential to discriminate, either as a result of malicious designers or perhaps from learning biases inherent in previous decisions on which an algorithm could be trained. In a well-known recent example, the COMPAS recidivism analysis tool, one of an increasingly popular set of algorithmic criminal "risk assessments" which are being used nationwide in sentencing and other decisions pertaining to defendants in the criminal justice system, was shown to exhibit highly disparate treatment between different races; a study by ProPublica [1,2] showed that African-American defendants who ultimately did not recidivate were almost twice as likely as white defendants to receive a high risk score from the algorithm.

As a result of these concerns, there has been extensive research in computer science and other fields pertaining to how *fairness*, or *non-discrimination*, should be defined in the context of a classification scenario. In this work, we will formalize and study one such definition, *fair treatment*, which is an approximate and distribution-based version of the notion of *equalized odds* [6] or *balance* [8].

[*] Supported in part by NSF Award CNS-1561209, NSF Award CNS-1217821, AFOSR Award FA9550-15-1-0262, a Microsoft Faculty Fellowship, and a Google Faculty Research Award.

Fair Treatment (a.k.a. approximate equalized odds). The originally proposed notion of fairness in classification is that of *statistical parity* [5] (which is essentially identical to the notion of causal effect [10]), which captures non-discrimination between groups. Given a classifier $\mathcal{C}$ which assigns to *individuals* σ from some distribution $\mathcal{D}$—each of which has some subset of *observable features* $O(\sigma)$—an *outcome* $\mathcal{C}(O(\sigma))$ (e.g., a risk score), and given a function $f(\sigma)$ representing an individual's *actual class* (e.g., whether they will recidivate), statistical parity simply requires that the output of the classifier be independent (or almost independent) of the group of the individual; that is, for any two groups X and Y, the distributions $\{\mathcal{C}(O(\boldsymbol{\sigma}_X))\}$ and $\{\mathcal{C}(O(\boldsymbol{\sigma}_Y))\}$ are ϵ-close in statistical distance. This is a very strong notion of fairness, and in many contexts it may not make sense. In particular, if the *base rates* (e.g., the base percentages of people from each race who actually recidivate) are different, we should perhaps not expect the output distribution of the classifier to be the same across groups. Indeed, as the ProPublica article points out, in the COMPAS example, the overall recidivism probability among African-American defendants was 56%, whereas it was 42% among white defendants. Thus, in such situations, one would reasonably expect a classifier to *on average* output a higher risk score for African-American defendants, which would violate statistical parity. Indeed, the issue raised by ProPublica authors was that, even after taking this base difference into account (more precisely, even after conditioning on individuals that did not recidivate), there was a significant difference in how the classifier treated the two races.

The notion of *equalized odds* due to Hardt et al. [6] formalizes the desiderata articulated by the authors of the ProPublica study (for the case of recidivism) in a general setting by requiring the output of the classifier to be independent of the group of the individuals, *after conditioning on the class of the individuals.* Very similar notions of fairness appear also in works such as [3, 8] using different names; for instance, Kleinberg et al. [8] consider a notion of "balance" which is an approximate version of equalized odds, albeit one which is tailored to scoring-based classifiers over a binary class space and only requires the conditioned expectation of the outcome (i.e., the score) to be close between groups. We here consider a more general approximate version of this notion which applies to all classifiers with a finite outcome space, which we refer to as *ϵ-fair treatment.* This requires that, for any two groups X and Y and any class c, the distributions

- $\{\mathcal{C}(O(\boldsymbol{\sigma}_X)) \mid f(\boldsymbol{\sigma}_X) = c\}$
- $\{\mathcal{C}(O(\boldsymbol{\sigma}_Y)) \mid f(\boldsymbol{\sigma}_Y) = c\}$

are ϵ-close with respect to some appropriate distance metric to be defined shortly. That is, in the COMPAS example, if we restrict to individuals that actually do not recidivate (respectively, those that do), the output of the classifier ought to be essentially independent of the group of the individual (just as intuitively desired by the authors of the ProPublica study, and as explictly put forward in [6]).

We will effectively use the notion of *max-divergence* to determine the "distance" between distributions; this notion, often found in areas such as differential privacy (see [4]), represents this distance as (the logarithm of) the *maximum multiplicative gap* between the probabilities of some element in the respective distributions. We argue that using such a multiplicative distance is important to ensure fairness between groups that may be under-represented in the data (see Section 3.1). Furthermore, as we note in the same appendix, such a notion is closed under "post-processing": if a classifier $\mathcal{C}$ satisfies ϵ-fair treatment with respect to a context $\mathcal{P} = (\mathcal{D}, f, g, O)$, then for any (possibly probabilistic) function $\mathcal{M}$, $\mathcal{C}'(\cdot) = \mathcal{M}(\mathcal{C}(\cdot))$ will also satisfy ϵ-fair treatment with respect to

$\mathcal{P}$. Closure under post-processing is important as we ultimately want the output of any subsequent classifier that uses only the output of a prior fair classifier to be fair as well.[1]

Can we sanitize an "unfair" classifier? As shown in the ProPublica study, the COMPAS classifier has a considerably large error in balance between races and hence also has a large error in the stronger notion of fair treatment. A natural question, then, would be whether we can "post-process" the output of this unfair classifier (or others) to satisfy some notion of balance or fair treatment. Indeed, there is a considerable amount of research devoted to *achieving* various definitions of fairness in practice. This is a highly non-trivial problem, in fact; early naïve approaches, such as just removing protected attributes from the feature set, fail due to redundant encodings for such features in the data (as discussed in [5]).

This question was more recently addressed in the work of Hardt et al. [6], who examine various methods by which a potentially unfair classifier can be post-processed into a fair *binary* classifier. They formalized the notion of a *$\mathcal{C}$-derived classifier*: namely a classifier $\mathcal{C}'$ obtained from $\mathcal{C}$ by first running $\mathcal{C}$, and then "perturbing" the output of $\mathcal{C}$. More precisely, such a $\mathcal{C}$ derived classifier may be specified by a "perturbation matrix" P where entry $P_{i,j}$ indicates with what probability output i gets perturbed into output j. Hardt et al. showed that for classifiers $\mathcal{C}$ over a binary outcome spaces, we can construct non-trivial $\mathcal{C}$-derived classifiers that satisfy their notion of equalized odds (in our terminology "perfect" fair treatment). Subsequent work [13] using this method showed that, for a binary version of the COMPAS classifier (which only attempts to predict recidivism and not output a risk score), it can produce a perfectly fair classifier with only an overall loss in accuracy of roughly 1.5%. Their method, however, requires "perfect" knowledge of the distribution $\mathcal{D}$ as well as of the classifier $\mathcal{C}$ in order to demonstrate optimality; additionally, as mentioned, it only applies to binary outcomes (and as such, does not directly apply to a risk assessment setting such as COMPAS).[2]

Thus, the literature leaves open the questions of (1) whether we can *efficiently* find a $\mathcal{C}$-derived classifier (without having perfect knowledge of $\mathcal{D}$ and $\mathcal{C}$), and (2) whether sanitization can be done for non-binary outputs.

Towards addressing this problem, we first formalize the notion of *black-box sanitization*: how to efficiently find a $\mathcal{C}$-derived classifier given just black-box access to a "sampling oracle" which samples random individuals $\sigma \leftarrow \mathcal{D}$ and outputs $(O(\sigma), f(\sigma), \mathcal{C}(O(\sigma)), g(\sigma))$ (that is, the individual's observable features, prior classification $\mathcal{C}$, actual class, and group, which is essentially the data used by the ProPublica authors to investigate the fairness of COMPAS).

Definition 1 (Informally stated.)**.** We call an algorithm $\mathcal{B}$ a **black-box sanitizer** if, given a distribution $\mathcal{D}$ and a sequence of prior classifiers $\{\mathcal{C}_n\}$ such that $\mathcal{C}_n$ takes as input n-bit descriptions $O_n(\sigma)$ of individuals' features[3], then, for each n, it:

- runs in time polynomial in n, and

[1] Remarking once again on the earlier definition of Kleinberg et al., we note that while it is equivalent to our definition for the case of binary outcomes, it is weaker for non-binary outcomes (as in the case of the COMPAS classifier). Furthermore, as with most expectation-based definitions, it is not closed under post-processing.

[2] Hardt et al [6] also presented a method for sanitizing a classifier outputting a risk-score (just as COMPAS), but the final, derived, classifier again would only output a single bit.

[3] Here we consider a sequence of classifiers for the sake of defining "computational efficiency" of a sanitizer; in particular, we would like the running time of our sanitizer to be polynomial in the feature length n.

- outputs some $\mathcal{C}_n$-derived classifier $\mathcal{C}'_n$ which, with overwhelming probability $1-\nu(n)$ for some $\nu(\cdot)$ negligible[4] in n, satisfies approximate fair treatment (with some small error $\epsilon(n)$) for individuals $\sigma \leftarrow \mathcal{D}$,

while only making "black-box" queries to the prior classifier. (That is, $\mathcal{B}$ cannot use any information about $\mathcal{D}$ or $\mathcal{C}_n$ aside from querying random samples $(O_n(\sigma'), f(\sigma'), \mathcal{C}_n(O_n(\sigma')), g(\sigma'))$ for $\sigma' \leftarrow \mathcal{D}$.)

Our key result is the construction of an efficient (i.e., polynomial-time in n) black-box sanitizer $\mathcal{B}$ that works for any distribution $\mathcal{D}$ and prior classifier sequence $\{\mathcal{C}_n\}$ *over a fixed outcome space*, and produces a classifier which not only satisfies approximate fair treatment but also can be shown to be near-optimal in terms of prediction accuracy (though the same also holds for a more general class of linear loss functions, which are formalized in the main statement of the result):

Theorem 1 (Informally stated)**.** For any fixed outcome space Ω, group space $\mathbb{G}$, and inverse polynomial $\epsilon(n)$, there exists a black-box sanitizer $\mathcal{B}$ with fair treatment error $\epsilon(n)$ such that, with probability at least $1-\nu(n)$ over $\mathcal{B}$'s queries for some inverse-exponential $\nu(\cdot)$, the accuracy loss of the classifier $\mathcal{C}'$ output by $\mathcal{B}$ (compared to the optimal $\mathcal{C}$-derived classifier over the same $\mathcal{D}$, f, and $\mathcal{C}$) is bounded by $|\Omega|(\epsilon(n)+\epsilon(n)^4|\mathbb{G}|/32)$.

We note that while Hardt et al. demonstrate a classifier satisfying *errorless* fair treatment, our derived classifier only satisfies ϵ-approximate fair treatment for some small ϵ, but this is unavoidable as we do not assume knowledge of the distribution $\mathcal{D}$. In contrast, we show how this classifier can be *efficiently* found without this knowledge of $\mathcal{D}$; additionally, our method applies to classifiers over any finite outcome space, as opposed to just binary outcomes.

We also experimentally evaluate the accuracy of our post-processing technique using a data set from the COMPAS recidivism analysis tool [1]. We investigate the fair treatment rates of the original data set and subsequently use the above technique to create classifiers satisfying fair treatment with varying errors while optimizing three different loss functions, amounting to overall accuracy (when considering a binary version of the classifier where scores 0-5 get mapped to a 0, and 6-10 get mapped to 1) and two notions of the similarity of the derived classification to the original classification. We find that our method is able to produce derived classifiers satisfying fair treatment with a relatively small amount of loss (with respect to this experimental data).

1.1 Proof Outline for Theorem 1

We show our sanitization theorem in three steps. First, we consider an arbitrary $\mathcal{C}$-derived classifier, and we demonstrate constraints for a linear program that can be used to efficiently find the optimal such classifier $\mathcal{C}'$ satisfying fair treatment. We note that these constraints are precisely a generalized version of those which Hardt et al. [6] demonstrate for *binary* classifiers $\mathcal{C}$ (though they also consider $\mathcal{C}$ with larger outcome spaces); we, however, also leverage our approximate definition to create constraints for *approximate* fair treatment. We further note that solving this linear program will require time polynomial in the number of possible outcomes $|\mathcal{O}|$.

Of course, our linear constraints, as well as the loss function we wish to optimize, may in general depend on features of $\mathcal{D}$ and $\mathcal{C}$ that we may in this model only approximate with black-box queries. So, towards approximating this optimal classifier in a black-box setting, we show that it suffices to use experimental probabilities derived from these queries rather than actual probabilities to

[4] That is, asymptotically smaller than any inverse polynomial $1/p(n)$.

build the linear program, since over sufficiently many queries, and as long as real probabilities are sufficiently large, it is overwhelmingly likely by a simple Chernoff bound that the experimental probabilities will be very close to accurate. To deal with the case when real probabilities may be quite small (and prone to large multiplicative error in estimation due to variance in samples), we additionally add a very small amount of random noise to the classifier in order to smooth out the multiplicative distance between real and experimental probabilities, effectively by increasing the minimum possible probability of events (noting that the noise is optional when the probabilities we wish to calculate experimentally are reasonably large). By solving this approximate version of the linear program, we may obtain a near-optimal derived classifier satisfying approximate fair treatment with respect to a given loss function.

However, the loss function we wish to minimize in the linear program is also potentially dependent on certain probabilities of events over $\mathcal{C}$ and $\mathcal{D}$ which require non-black-box knowledge to derive exactly; to overcome this, we show that the constructed sanitizer can in fact estimate these accurately using black-box queries by the same argument as that for the linear program's coefficients, and so, given enough samples, an approximate loss function derived from experimental probabilities is overwhelmingly likely to be close to the real loss function. Of course, while the approximation of the loss function is close, it is unclear as to whether the optimum of the approximate loss function is necessarily close to optimal over the real loss function; we show, through leveraging properties of the loss function and the space over which it is defined, that in fact this is the case for accuracy (and other loss functions, including natural classes of loss functions that are linear in the probabilities $\Pr\left[\sigma \leftarrow \mathcal{D} : f(\sigma) = i \wedge \mathcal{C}(O(\sigma)) = j\right]$), which completes our argument of near-optimality.

2 Preliminaries and Definitions

2.1 Notation

Conditional probabilities. Given some random variable X and some event E, we let $\Pr[p(X) \mid E]$ denote the probability of a predicate $p(X)$ holding when conditioning the probability space on the event E. If the probability of E is 0, we slightly abuse notation and simply define $\Pr[p(X) \mid E] = 0$.

Multiplicative distance. The following definition of multiplicative distance will be useful to us. We let the **multiplicative distance** $\mu(x, y)$ between two real numbers $x, y \geq 0$ be defined as

$$\mu(x,y) = \begin{cases} \ln\left(\max\left(\frac{x}{y}, \frac{y}{x}\right)\right) & \text{if } x > 0, y > 0 \\ 0 & \text{if } x = y = 0 \\ \infty & \text{otherwise} \end{cases}$$

2.2 Classification Contexts

We start by defining classification contexts and classifiers.

Definition 2. A **classification context** $\mathcal{P}$ is denoted by a tuple $(\mathcal{D}, f, g, O)$ such that:

- $\mathcal{D}$ is a probability distribution with some finite support $\Sigma_{\mathcal{P}}$ (the set of all possible **individuals** to classify).

- $f : \Sigma_{\mathcal{P}} \to \Psi_{\mathcal{P}}$ is a surjective function that maps each individual to their **class** in a set $\Psi_{\mathcal{P}}$.
- $g : \Sigma_{\mathcal{P}} \to \mathbb{G}_{\mathcal{P}}$ is a surjective function that maps each individual to their **group** in a set $\mathbb{G}_{\mathcal{P}}$.
- $O : \Sigma_{\mathcal{P}} \to \{0,1\}^* \times \mathbb{G}_{\mathcal{P}}$ is a function that maps each individual σ to their **observable features** $(O'(\sigma), g(\sigma))$; note that we by default assume that an individual's group can be observed.

We note that f and g are deterministic; this is without loss of generality as we can encode any probabilistic features that f and g may depend on into σ as "unobservable features" of the individual.

Given such a classification context $\mathcal{P}$, we let $\Psi_{\mathcal{P}}$ denote the range of f, and $\mathbb{G}_{\mathcal{P}}$ denote the range of g. Whenever the classification context $\mathcal{P}$ is clear from context, we drop the subscript; additionally, whenever the distribution $\mathcal{D}$ and group function g are clear from context, we use $\boldsymbol{\sigma}$ to denote a random variable that is distributed according to $\mathcal{D}$, and $\boldsymbol{\sigma}_X$ to denote the random variable distributed according to $\mathcal{D}$ conditioned on $g(\sigma) = X$.

2.3 Classifiers

A **classifier** $\mathcal{C}$ for a classification context $\mathcal{P} = (\mathcal{D}, f, g, O)$ is simply a (possibly randomized) algorithm that acts on the support of O (the observable description of an individual). We let $\Omega_{\mathcal{P}}^{\mathcal{C}}$ denote the support of the distribution $\{\mathcal{C}(O(\boldsymbol{\sigma}))\}$.

We also must formalize what it means for a classifier to be "derived" from another classifier; hence, we define the following notion of a classifier $\mathcal{C}'$ that "perturbs" the output of some original classifier $\mathcal{C}$. Given an individual σ, $\mathcal{C}'$ will run $\mathcal{C}$ and then "post-process" the output according only to the output $\mathcal{C}(O(\sigma))$ and σ's group.

Definition 3. [6] Given a classifier $\mathcal{C}$, we say that a classifier $\mathcal{C}'$ is a **$\mathcal{C}$-derived** classifier if, in any context $\mathcal{P} = (\mathcal{D}, f, g, O)$, the outcome $\mathcal{C}'$ is only dependent on $\mathcal{C}(O(\sigma))$ and σ's group $g(\sigma)$. (Equivalently, $\mathcal{C}'$ is a classifier over the context $\mathcal{P}' = (\mathcal{D}, f, g, (\mathcal{C}(O(\cdot)), g(\cdot))).$)

Formally, we can represent this as a $|\Omega_{\mathcal{P}}^{\mathcal{C}}| \times |\Omega_{\mathcal{P}}^{\mathcal{C}}| \times |\mathbb{G}_{\mathcal{P}}|$ vector $\vec{P}_{\mathcal{C}'}$ of the probabilities

$$P_{i,j}^{g} = \Pr\left[\mathcal{C}'(\mathcal{C}(O(\boldsymbol{\sigma}_g)), g) = j | \mathcal{C}(O(\boldsymbol{\sigma}_g)) = i\right]$$

and let $\mathcal{C}'$ be a classifier that, given an individual σ, runs $\mathcal{C}$ on that individual, observes its outcome $i = \mathcal{C}(O(\sigma))$ and group $g(\sigma)$, and assigns that individual the distribution of outcomes $\{j \text{ with pr. } P_{i,j}^{g}\}$.

3 Defining Fair Treatment

Next, we define the notion of *fair treatment* for a classifier $\mathcal{C}$, which is an approximate version of the notion of "equalized odds" from Hardt et al. [6] (which in turn was derived from notions implicit in the ProPublica study [2]).

Definition 4. *(Fair treatment, a.k.a. approximate equalized odds [6].)* We say that a classifier $\mathcal{C}$ satisfies **ϵ-fair treatment** with respect to a context $\mathcal{P} = (\mathcal{D}, f, g, O)$ if, for any groups $X, Y \in \mathbb{G}_{\mathcal{P}}$, any class $c \in \Psi_{\mathcal{P}}$, and any outcome $o \in \Omega_{\mathcal{P}}^{\mathcal{C}}$, we have that

$$\mu(\Pr[\mathcal{C}(O(\boldsymbol{\sigma}_X)) = o \mid f(\boldsymbol{\sigma}_X) = c], \Pr[\mathcal{C}(O(\boldsymbol{\sigma}_Y)) = o \mid f(\boldsymbol{\sigma}_Y) = c]) \leq \epsilon$$

For the case of binary classification tasks and binary classifiers (i.e., when $\Psi_{\mathcal{P}} = \Omega_{\mathcal{P}}^{\mathcal{C}} = \{0, 1\}$), fair treatment is equivalent to requiring "similar" false positive and false negative rates [8].

3.1 On the Use of Multiplicative Distance

As defined here, fair treatment essentially requires that the *max-divergence* between the conditional distributions of outcomes is small between groups. Max-divergence is a distance measure often found in areas such as differential privacy (see [4]); we stress here, through two arguments following very similar logic to differential privacy, that using such a multiplicative distance is important to ensure fairness between groups that may be under-represented in the data, and also that fair treatment defined using multiplicative distance exhibits desirable properties that other distance metrics may not.

First, to motivate our statement that multiplicative distances are important for parity between under-represented groups, consider as an example a classifier used to determine whether to search people for weapons. Assume such a classifier determined to search 1% of minorities at random, but *only* the minorities (and no others). Such a classifier would still have a fair treatment error of 0.01 if we used standard statistical distance, while the max-divergence would in fact be infinite (and indeed, such a classification would be blatantly discriminatory).

Our use of max-divergence between distributions for our definitions is reflective of the fact that, in cases where we have such small probabilities, discrimination should be measured multiplicatively, rather than additively. In addition, when we may have a large number of possible classes, the use of max-divergence (in particular, the *maximum* of the log-probability ratios) means that we always look at the class with the *most* disparity to determine how discriminatory a classification is, rather than potentially amortizing this disparity over a large number of classes.

3.2 Closure under Post-Processing

We also remark that our definition of fair treatment is closed under "post-processing". If a classifier $\mathcal{C}$ satisfies ϵ-fair treatment with respect to a context $\mathcal{P} = (\mathcal{D}, f, g, O)$, then any $\mathcal{C}$-derived classifier *which acts independently of an individual's group* (i.e., whose decision is based only on the outcome of $\mathcal{C}$) will also satisfy ϵ-fair treatment with respect to $\mathcal{P}$.

Theorem 2. Let $\mathcal{C}_1$ be a classifier satisfying ϵ-fair treatment with respect to a context $\mathcal{P} = (\mathcal{D}, f, g, O)$. Let $\mathcal{C}_2$ be any classifier whose output for an individual σ is strictly a (possibly probabilistic) function of $\mathcal{C}_1(O(\sigma))$. Then $\mathcal{C}_2$ satisfies ϵ-fair treatment with respect to $\mathcal{P}$.

Proof. Let $\mathcal{C}_1$ be a classifier satisfying ϵ-fair treatment w.r.t. some context $\mathcal{P}$. Consider some groups $X, Y \in \mathbb{G}_{\mathcal{P}}$, some class $c \in \Psi_{\mathcal{P}}$, and some outcome $o \in \Omega_{\mathcal{P}}^{\mathcal{C}'}$; we need to show that

$$\mu(\Pr[\mathcal{C}_2(\mathcal{C}_1(O(\boldsymbol{\sigma}_X))) = o \mid f(\boldsymbol{\sigma}_X) = c], \Pr[\mathcal{C}_2(\mathcal{C}_1(O(\boldsymbol{\sigma}_Y))) = o \mid f(\boldsymbol{\sigma}_Y) = c]) \leq \epsilon$$

Towards doing this, note that

$$\begin{aligned} &\Pr[\mathcal{C}_2(\mathcal{C}_1(O(\boldsymbol{\sigma}_X))) = o \mid f(\boldsymbol{\sigma}_X) = c] \\ = \sum_{o_1 \in \Omega_{\mathcal{P}}^{\mathcal{C}_1}} &\Pr[\mathcal{C}_2(o_1) = o \mid f(\boldsymbol{\sigma}_X) = c, \mathcal{C}_1(O(\boldsymbol{\sigma}_X)) = o_1] \Pr[\mathcal{C}_1(O(\boldsymbol{\sigma}_X)) = o_1 \mid f(\boldsymbol{\sigma}_X) = c] \\ &= \sum_{o_1 \in \Omega_{\mathcal{P}}^{\mathcal{C}_1}} \Pr[\mathcal{C}_2(o_1) = o] \Pr[\mathcal{C}_1(O(\boldsymbol{\sigma}_X)) = o_1 \mid f(\boldsymbol{\sigma}_X) = c] \end{aligned}$$

where the last step follows from the fact that $\mathcal{C}_2$ depends only on $\mathcal{C}_1$. By the same argument applied to Y, we also have that:

$$\begin{aligned} &\Pr[\mathcal{C}_2(\mathcal{C}_1(O(\boldsymbol{\sigma}_Y))) = o \mid f(\boldsymbol{\sigma}_Y) = c] \\ = \sum_{o_1 \in \Omega_{\mathcal{P}}^{\mathcal{C}_1}} &\Pr[\mathcal{C}_2(o_1) = o] \Pr[\mathcal{C}_1(O(\boldsymbol{\sigma}_Y)) = o_1 \mid f(\boldsymbol{\sigma}_Y) = c] \end{aligned}$$

These two probabilities are ϵ-close since, by fair treatment, $\Pr[\mathcal{C}_1(O(\boldsymbol{\sigma}_X)) = o_1 \mid f(\boldsymbol{\sigma}_X) = c]$ and $\Pr[\mathcal{C}_1(O(\boldsymbol{\sigma}_Y)) = o_1 \mid f(\boldsymbol{\sigma}_Y) = c]$ are ϵ-close, and furthermore multiplicative distance is preserved under linear operations[5]. This proves the theorem. □

We also remark that, in general, earlier "expectation-based" definitions of fair treatment are not preserved under post-processing.

4 Black-Box Sanitization

Next, we provide a novel definition of the type of sanitizer we shall construct in our main theorem.

For the purposes of defining a "computationally efficient" sanitizer, let us define a notion of an "ensemble" of classification contexts, wherein we assume a parameter n (similar to the idea of a security parameter in cryptography) so that each individual's observable features can be represented in n bits. In particular, this means that, for some setting of n there may be up to 2^n distinct descriptions of individuals in a distribution $\mathcal{D}$, and so a *computationally efficient* black-box classifier which runs in polynomial time with respect to n could not, for instance, query every possible feature description.

Definition 5. Let a **classification context ensemble** Π be given by a sequence of classification contexts $\{\mathcal{P}_n\}_{n\in\mathbb{N}} = \{(\mathcal{D}, f, g, O_n)\}_{n\in\mathbb{N}}$ (note that $(\mathcal{D}, f, g)$ remains the same as n varies), such that, whenever $2^n \geq |\mathbb{G}_{\mathcal{P}_n}|$ (i.e., n is sufficiently large to describe $g(\sigma)$), O_n maps the space $\Sigma_{\mathcal{P}_n}$ of individuals to $\{0,1\}^n$, the space of n-bit descriptions.

Notably, the contexts are effectively describing the same distribution of individuals, but using different feature lengths for each context in the ensemble. Also, because $\mathcal{D}$, f, and g are the same throughout, this implies that the space of individuals $\Sigma_{\mathcal{P}_n}$ and the class and group spaces $\Psi_{\mathcal{P}_n}$ and $\mathbb{G}_{\mathcal{P}_n}$ are likewise the same for every n.

In our proofs, we will also consider deriving our classifier from a *sequence* of prior classifiers $\chi = \{\mathcal{C}_n\}_{n\in\mathbb{N}}$, where the classifier $\mathcal{C}_i$ is used to classify individuals in the context $\mathcal{P}_i$ (that is, individuals having feature length i).

Lastly, we wish to represent the fact that a sanitizer may, given a prior classifier sequence χ over a distribution ensemble Π, wish to make black-box queries to a distribution of labeled "training data" representing individuals' observable features, classes, groups, and prior classifications. We shall denote this distribution for a specific parameter n by

$$\tau_{\chi,\Pi}(1^n) \triangleq \{\sigma \leftarrow \mathcal{D} : (O_n(\sigma), f(\sigma), \mathcal{C}_n(O_n(\sigma)), g(\sigma))\}$$

[5] That is, if $\mu(a,b) \leq \epsilon$ and $\mu(a',b') \leq \epsilon$ then $\mu(\alpha a + \beta a', \alpha b + \beta b') \leq \epsilon$.

Notationally, let $\mathcal{B}^{\tau_{\chi,\Pi}}(1^n)$[6] denote that a sanitizer $\mathcal{B}$ may make black-box queries to the distribution $\tau_{\chi,\Pi}(1^n)$ for some parameter n. Finally, we are able to formalize the notion of a "black-box sanitizer" given the above:

Definition 6. We say that an algorithm $\mathcal{B}^{(\cdot)}$ is an $\epsilon(\cdot)$-**black-box sanitizer** if it is:

- *Efficient:* there exists a polynomial $p(\cdot,\cdot)$ such that, for any $m \in \mathbb{N}$, and for any context ensemble $\Pi = \{\mathcal{P}_n\}_{n\in\mathbb{N}}$ and sequence $\chi = \{\mathcal{C}_n\}_{n\in\mathbb{N}}$ of classifiers for which $|\Psi_{\mathcal{P}_n}| \leq m$, $|\mathbb{G}_{\mathcal{P}_n}| \leq m$, and $|\Omega_{\mathcal{P}_n}^{\mathcal{C}_n}| \leq m$ (i.e., the class, group, and output spaces have size bounded by m), $\mathcal{B}^{\tau_{\chi,\Pi}}(1^n)$ runs in time at most $p(m,n)$ for all $n \in \mathbb{N}$.
- *Fair:* for any context ensemble $\Pi = \{\mathcal{P}_n\}_{n\in\mathbb{N}}$ and any sequence $\chi = \{\mathcal{C}_n\}_{n\in\mathbb{N}}$ of classifiers, there exists negligible $\nu(\cdot)$[7] such that, for all $n \in \mathbb{N}$, with probability at least $(1-\nu(n))$ over the samples it queries from $\tau_{\chi,\Pi}(1^n)$, $\mathcal{B}^{\tau_{\chi,\Pi}}(1^n)$ outputs a $\mathcal{C}_n$-derived classifier $\mathcal{C}'$[8] which satisfies $\epsilon(n)$-fair treatment with respect to $\mathcal{P}_n$.

4.1 Loss Functions

Lastly, we need to define "optimality" for derived classifiers in this context. In particular, we assume some loss function $\ell(\cdot)$ bounded in $[0,1]$ which may either be fixed or based on $\mathcal{D}$, f, and $\mathcal{C}$ (in which case we write $\ell_{\mathcal{D},f,\mathcal{C}}(\cdot)$ for clarity). Intuitively, $\ell(\mathcal{C}')$ represents the "loss" in utility incurred by classifying an individual σ with outcome $\mathcal{C}'(O(\sigma))$ when their actual class is $f(\sigma)$. As a concrete example, if we consider classifiers which attempt to classify each individual according to their correct class $f(\sigma) \in \Psi$, one might consider the *overall inaccuracy* as a loss function, which is given by:

$$\ell_{\mathcal{D},f,\mathcal{C}}(\mathcal{C}') = 1 - \Pr\left[\mathcal{C}'(O(\boldsymbol{\sigma})) = f(\boldsymbol{\sigma})\right]$$

We can define the *error* of a derived classifier to be its loss compared to the optimal *perfectly fair* derived classifier, as follows:

Definition 7. For some context $\mathcal{P} = (\mathcal{D}, f, g, O)$ and prior classifier $\mathcal{C}$, given some loss function $\ell_{\mathcal{D},f,\mathcal{C}}$ that maps any classifier to its loss in $[0,1]$, letting $\mathcal{S}$ be the set of all $\mathcal{C}$-derived classifiers satisfying (errorless) 0-fair treatment, then we define the **error** of some $\mathcal{C}$-derived classifier $\mathcal{C}'$ with respect to $\ell_{\mathcal{D},f,\mathcal{C}}$ to be

$$\Delta_{\ell,\mathcal{D},f,\mathcal{C}}(\mathcal{C}') = \ \max_{\mathcal{C}^*\in\mathcal{S}}(\ell_{\mathcal{D},f,\mathcal{C}}(\mathcal{C}') - \ell_{\mathcal{D},f,\mathcal{C}}(\mathcal{C}^*))$$

We note that, because we compare a classifier (which may be only approximately fair) to the optimal *perfectly* fair classifier, certain particularly good classifiers may in fact have a negative loss. We could, when considering ϵ-approximately fair classifiers, generalize this notion to consider the loss over all $f(\epsilon)$-fair classifiers for some $f(\epsilon) < \epsilon$ and derive a similar optimality result to what we prove here, but for simplicity and consistency over different parameters ϵ we consider the case when $f(\epsilon) = 0$.

[6] The input of 1^n, or a string of n ones, is provided simply as a cryptographic convention, so that we can assert that the running time of $\mathcal{B}$ is polynomial in its input length. When implicit or clear from context, we shall for notational simplicity omit this input.

[7] That is, $\nu(n) < 1/p(n)$ for every polynomial $p(\cdot)$ and sufficiently large n.

[8] That is, $\mathcal{B}$ outputs the probabilities $\vec{P}_{\mathcal{C}'}$ corresponding to the derived classifier $\mathcal{C}'$.

Linear loss functions. Furthermore, with respect to derived classifiers, we consider the class of loss functions $\ell_{\mathcal{D},f,\mathcal{C}}$ which are *linear* in the probabilities $P^g_{i,j}$ constituting the derived classifier—that is:

Definition 8. We say that a loss function $\ell_{\mathcal{D},f,\mathcal{C}}(\cdot)$ is a **linear** loss function for a context $\mathcal{P} = (\mathcal{D}, f, g, O)$ and prior classifier $\mathcal{C}$ if it can be represented as some $|\Omega^{\mathcal{C}}_{\mathcal{P}}| \times |\Omega^{\mathcal{C}}_{\mathcal{P}}| \times |\mathbb{G}_{\mathcal{P}}|$ vector $\vec{\ell}_{\mathcal{D},f,\mathcal{C}}$ so that the loss of a derived classifier $\mathcal{C}'$ is given as the inner product

$$\langle \vec{\ell}_{\mathcal{D},f,\mathcal{C}}, \vec{P}_{\mathcal{C}'} \rangle = \sum_{i,j \in \Omega^{\mathcal{C}}_{\mathcal{P}}, g \in \mathbb{G}_{\mathcal{P}}} (\vec{\ell}_{\mathcal{D},f,\mathcal{C}})^g_{i,j} P^g_{i,j}$$

of this vector with the probabilities constituting the derived classifier $\mathcal{C}'$.

We can define error slightly more specifically for linear loss functions using the vector form:

$$\Delta_{\vec{\ell},\mathcal{D},f,\mathcal{C}}(\mathcal{C}') = \ \max_{\mathcal{C}^* \in \mathcal{S}}(\langle \vec{\ell}_{\mathcal{D},f,\mathcal{C}}, \vec{P}_{\mathcal{C}'} \rangle - \langle \vec{\ell}_{\mathcal{D},f,\mathcal{C}}, \vec{P}_{\mathcal{C}^*} \rangle)$$

We will focus on the specific subclass of linear loss functions whose coefficients (the coefficients of $P^g_{i,j}$) can either be constant or up to d^{th}-degree polynomials in probabilities $\Pr[g(\boldsymbol{\sigma}) = \gamma]$ and $\Pr[f(\boldsymbol{\sigma}_g) = i \wedge \mathcal{C}(O(\boldsymbol{\sigma}_g)) = j]$, which can be formalized as follows:

Definition 9. We shall define a **linear loss function with t-term coefficients of degree** d as one that can be represented as

$$\ell_{\mathcal{D},f,\mathcal{C}}(\mathcal{C}') = \sum_{i,j,g} q^g_{i,j}(\rho) P^g_{i,j}$$

or equivalently

$$(\vec{\ell}_{\mathcal{D},f,\mathcal{C}})^g_{i,j} = q^g_{i,j}(\rho)$$

where ρ denotes the set of all variables $\Pr[g(\boldsymbol{\sigma}) = \gamma]$ and $\Pr[f(\boldsymbol{\sigma}_g) = x \wedge \mathcal{C}(O(\boldsymbol{\sigma}_g)) = y]$ (for any γ, x, y), $P^g_{i,j}$ is the vector representation of $\mathcal{C}'$, and each $q^g_{i,j}(\cdot)$ is a d^{th}-degree polynomial in the variables of ρ which contains at most t monomials which themselves are bounded in $[0,1]$ whenever the variables in ρ are likewise bounded.

We note that overall inaccuracy as described above is in fact a linear loss function with $(|\Omega^{\mathcal{C}}_{\mathcal{P}}|-1)$-term coefficients of degree 2, as we shall shortly demonstrate; furthermore, a wide variety of other useful loss functions are also linear with degree-2 coefficients. Returning to the example of COMPAS from the introduction, for instance, we see that the space of outcomes is a "risk score" from 1 to 10, while the space of classes is binary (either recidivating or not), so rather than overall accuracy (which as noted above requires the spaces to be identical) we will need another notion of loss. We exhibit three useful loss functions for this scenario in Section 6, all of which will have degree-2 coefficients, which we will use to evaluate the quality of the fair classifiers we derive from COMPAS. Returning to investigating the notion of overall inaccuracy:

Claim 1. For a context $\mathcal{P}$ and for any classifier with $\Omega^{\mathcal{C}}_{\mathcal{P}} = \Psi_{\mathcal{P}} = \mathcal{O}$, the overall inaccuracy loss function

$$\ell_{\mathcal{D},f,\mathcal{C}}(\mathcal{C}') = 1 - \Pr\left[\mathcal{C}'(O(\boldsymbol{\sigma})) = f(\boldsymbol{\sigma})\right]$$

is a linear loss function with $(|\Omega^{\mathcal{C}}_{\mathcal{P}}| - 1)$-term coefficients of degree 2.

Proof. The inaccuracy of a classifier, conditioning on a group g, can be expressed as a linear function in $P^g_{i,j}$ if $\mathcal{D}, f, \mathcal{C}$ are fixed:

$$\Pr\left[f(\boldsymbol{\sigma}_g) \neq \mathcal{C}'(O(\boldsymbol{\sigma}_g))\right] = 1 - \Pr\left[f(\boldsymbol{\sigma}_g) = \mathcal{C}'(O(\boldsymbol{\sigma}_g))\right]$$

$$= 1 - \sum_{j \in \mathcal{O}} \Pr\left[f(\boldsymbol{\sigma}_g) = j \wedge \mathcal{C}'(O(\boldsymbol{\sigma}_g)) = j\right]$$

$$= 1 - \sum_{i,j \in \mathcal{O}} \Pr\left[f(\boldsymbol{\sigma}_g) = j \wedge \mathcal{C}(O(\boldsymbol{\sigma}_g)) = i \wedge \mathcal{C}'(O(\boldsymbol{\sigma}_g)) = j\right]$$

$$= 1 - \sum_{i,j \in \mathcal{O}} \Pr\left[f(\boldsymbol{\sigma}_g) = j \wedge \mathcal{C}(O(\boldsymbol{\sigma}_g)) = i\right] \Pr\left[\mathcal{C}'(O(\boldsymbol{\sigma}_g)) = j | f(\boldsymbol{\sigma}_g) = j \wedge \mathcal{C}(O(\boldsymbol{\sigma}_g)) = i\right]$$

Recalling that the output of $\mathcal{C}'$ is based only on an individual's group and the output of $\mathcal{C}$:

$$= 1 - \sum_{i,j \in \mathcal{O}} \Pr\left[f(\boldsymbol{\sigma}_g) = j \wedge \mathcal{C}(O(\boldsymbol{\sigma}_g)) = i\right] \Pr\left[\mathcal{C}'(O(\boldsymbol{\sigma}_g)) = j | \mathcal{C}(O(\boldsymbol{\sigma}_g)) = i\right]$$

$$= 1 - \sum_{i,j \in \mathcal{O}} \Pr\left[f(\boldsymbol{\sigma}_g) = j \wedge \mathcal{C}(O(\boldsymbol{\sigma}_g)) = i\right] P^g_{i,j}$$

This can be expanded into the overall inaccuracy of $\mathcal{C}'$ if we sum over groups, i.e.,

$$1 - \sum_{i,j \in \mathcal{O}; \gamma \in \mathbb{G}_{\mathcal{P}}} \Pr\left[g(\boldsymbol{\sigma}) = \gamma\right] \Pr\left[f(\boldsymbol{\sigma}_\gamma) = j \wedge \mathcal{C}(O(\boldsymbol{\sigma}_\gamma)) = i\right] P^\gamma_{i,j}$$

or, equivalently,

$$\sum_{i,j \in \mathcal{O}; \gamma \in \mathbb{G}_{\mathcal{P}}} \Pr\left[g(\boldsymbol{\sigma}) = \gamma\right] \sum_{k \neq j} \Pr\left[f(\boldsymbol{\sigma}_\gamma) = k \wedge \mathcal{C}(O(\boldsymbol{\sigma}_\gamma)) = i\right] P^\gamma_{i,j}$$

This suggests that we can, as previously described, write this loss function as a vector $\vec{\ell}_{\mathcal{D},f,\mathcal{C}}$ over the space of probabilities $P^\gamma_{i,j}$, in particular such that

$$(\vec{\ell}_{\mathcal{D},f,\mathcal{C}})^\gamma_{i,j} = \Pr\left[g(\boldsymbol{\sigma}) = \gamma\right] \sum_{k \neq j} \Pr\left[f(\boldsymbol{\sigma}_\gamma) = k \wedge \mathcal{C}(O(\boldsymbol{\sigma}_\gamma)) = i\right]$$

Notably, each of these coefficients has $\mathcal{O} - 1 = |\Omega^{\mathcal{C}}_{\mathcal{P}}| - 1$ monomials bounded in $[0, 1]$ which are degree 2 in the probabilities of the form $\Pr\left[g(\boldsymbol{\sigma}) = \gamma\right]$ and $\Pr\left[f(\boldsymbol{\sigma}_g) = x \wedge \mathcal{C}(O(\boldsymbol{\sigma}_g)) = y\right]$, as desired. □

5 Theorem: Achieving Fair Treatment by Post-Processing

We now show that it is possible to achieve fair treatment, even in non-binary classification scenarios, by post-processing starting from a prior classification that may be unfair. We note that, though our theorems only state existence, we provide our concrete construction of the black-box sanitizer in the body of the proof. Focusing first on the specific example above where we use inaccuracy as a loss function, we show the following positive result:

Theorem 3. For any fixed outcome space Ω, any polynomial $q(n)$, and any $\epsilon(n) \in [\frac{1}{q(n)}, 1)$, there exists an $\epsilon(\cdot)$-black-box sanitizer $\mathcal{B}$ which, given any context ensemble $\Pi = \{\mathcal{P}_n\}_{n\in\mathbb{N}}$ (such that $|\mathbb{G}_{\mathcal{P}_n}| = m$) and any classifier sequence χ such that $\Psi_{\mathcal{P}_n} = \Omega_{\mathcal{P}_n}^{\mathcal{C}_n} = \Omega$, there exists negligible $\nu(\cdot)$ such that, with probability $1-\nu(n)$ over the samples it queries from $\tau_{\chi,\Pi}(1^n)$, $\mathcal{B}$ outputs a classifier $\mathcal{C}''$ which both satisfies $\epsilon(n)$-fair treatment and has error

$$\Delta_{\ell,\mathcal{D},f,\mathcal{C}}(\mathcal{C}'') \leq |\Omega|(\epsilon(n) + m(|\Omega|-1)\epsilon(n)^4/32)$$

with respect to the overall inaccuracy loss function

$$\ell_{\mathcal{D},f,\mathcal{C}}(\mathcal{C}'') = 1 - \Pr\left[\mathcal{C}''(O(\boldsymbol{\sigma})) = f(\boldsymbol{\sigma})\right]$$

This is in fact implied directly by the following more general result, which we shall prove in its stead:

Theorem 4. For any fixed outcome space Ω, any polynomial $q(n)$, and any $\epsilon(n) \in [\frac{1}{q(n)}, 1)$, there exists an $\epsilon(\cdot)$-black-box sanitizer $\mathcal{B}$ which, given any context ensemble $\Pi = \{\mathcal{P}_n\}_{n\in\mathbb{N}}$ and any classifier sequence χ, there exists negligible $\nu(\cdot)$ such that, with probability $1-\nu(n)$ over the samples it queries from $\tau_{\chi,\Pi}(1^n)$, $\mathcal{B}$ outputs a classifier $\mathcal{C}''$ which both satisfies $\epsilon(n)$-fair treatment and has error

$$\Delta_{\ell,\mathcal{D},f,\mathcal{C}}(\mathcal{C}'') \leq |\Omega_{\mathcal{P}_n}^{\mathcal{C}_n}|(\epsilon(n) + |\mathbb{G}_{\mathcal{P}_n}|\epsilon(n)^4 dt/64)$$

with respect to any linear loss function with t-term coefficients of degree d.

In the example above where we consider overall inaccuracy, we have (by Claim 1) $d = 2$ and $t = |\Omega| - 1$, directly implying Theorem 3. Next, we outline the proof of Theorem 4:

Achieving fair treatment with distributional knowledge. We begin with the simplifying assumption that the sanitizer we construct does have perfect knowledge of the context Π and classifier $\chi = \{\mathcal{C}_n\}_{n\in\mathbb{N}}$, and we show (Claim 2) that for each n we can use the probabilities of events in those distributions to construct a set of linear constraints for fair treatment over the probabilities $P_{i,j}^g = \Pr[\mathcal{C}'(O(\boldsymbol{\sigma}_g)) = j | \mathcal{C}_n(O(\boldsymbol{\sigma}_g)) = i]$. Then, given a loss function which is also linear in $P_{i,j}^g$, we can construct a linear program (Corollary 1) to efficiently minimize loss subject to the constraints for fair treatment. Since, by construction, any $\mathcal{C}_n$-derived $\mathcal{C}'(\sigma)$ which satisfies fair treatment will lie within the region determined by our constraints, we have shown that it is possible to efficiently determine the optimal fair $\mathcal{C}_n$-derived classifier (with respect to any linear loss function) in a non-black-box setting.

Black-box approximation. Next, we work towards discarding the assumption of non-black-box knowledge of Π and χ. In particular, we use a Chernoff-type bound to show (Lemma 3) that, given a sufficiently large (yet still polynomial in n) number of labeled and classified samples from $\tau_{\chi,\Pi}(1^n)$, with very high probability (i.e., probability $1-\nu(n)$) all of the experimental probabilities relevant to our linear program will be close enough to their actual counterparts so that any solution to the linear program formulated from the experimental probabilities will also satisfy approximate fair treatment with respect to the actual probabilities. However, we note that the Chernoff bound will only apply when the real probabilities of the events in question are sufficiently large; if we are not guaranteed that this is the case, we additionally add a very small amount of noise to the classifier $\mathcal{C}'$ to deal with the possibility that events with very small real probability are measured to have a wildly different experimental probability due to sampling variance. This random noise will ensure that these events are accounted for when approximating the linear program while adding only a minimal error to the approximation. So, combined with the previous step, this suggests the approach that we will use to construct the final sanitizer $\mathcal{B}$; specifically, we can do as follows:

- Use a sufficiently large (yet polynomial in n) number of samples from $\tau_{\chi,\Pi}(1^n)$ to estimate the parameters of the linear constraints from the previous step, in particular using a fairness error significantly smaller than $\epsilon(n)$ in order to account for variance in samples and random noise that will be added, yet one large enough to not rule out optimal classifiers that may not be perfectly fair. Also use the samples to estimate any distributionally dependent parameters of the loss function.

- Use standard linear programming techniques to optimize the derived loss function over the derived constraint region in polynomial time, and take the optimal solution as the "transformation parameters" of a derived classifier $\mathcal{C}'$ (i.e., the probabilities $P^g_{i,j}$).

- Output the (slightly noisy) classifier $\mathcal{C}''$ which, except with a small probability, applies the transformation given by the above solution to the output of the prior classifier; the rest of the time, it returns a random outcome.

If parameterized correctly, this classifier will still satisfy ϵ-approximate fairness whenever all of the above Chernoff bounds hold; furthermore, as we subsequently show, the output will also not incur much loss due to estimating parameters and adding noise when these bounds hold.

Showing near-optimality. In particular, we must account both for the noise added to the solution $\mathcal{C}'$ to the linear program and for the fact that the loss function over which $\mathcal{B}$ optimizes may be imprecise, as we have remarked that loss functions such as accuracy are in general dependent on features of the context or the classifier (which our sanitizer must estimate using samples). However, once again, we show (Claim 5) that this can be overcome by using another Chernoff-type bound (Lemma 4) to show that, with high probability, the experimentally derived coefficients of the loss function are very close to the corresponding coefficients of the actual loss function. Then we demonstrate that a slightly noisy variant of the optimal $\mathcal{C}_n$-derived classifier is always derivable by $\mathcal{B}$ when the bounds hold, and furthermore use linearity to show that, in that case, the actual loss of the output $\mathcal{C}''$ must not differ by much from that of the optimal $\mathcal{C}_n$-derived classifier (in particular, the possible degree of difference depends on the degree and number of terms of the loss function's coefficients and the number of variables, i.e., the number of groups and outcomes possible), even when the intermediate classifier $\mathcal{C}'$ itself might differ from this classifier due to the optimum over the approximate loss function being different from the optimum over the actual loss function.

Notation. For brevity and notational simplicity, in the body of the proof we will abbreviate the probability $\Pr[E(\boldsymbol{\sigma}_g)]$ (i.e., the probability of some event E holding for σ drawn from group g) as $\Pr_g[E(\sigma)]$, and the probability $\Pr[g(\boldsymbol{\sigma}) = \gamma]$ as $\Pr[\gamma]$.

Furthermore, we abbreviate the event $f(\sigma) = i$ as f_i, and similarly for any classifier $\mathcal{C}$ abbreviate $\mathcal{C}(O(\sigma)) = i$ as $\mathcal{C}_i$.

5.1 Step 1: Achieving Fair Treatment

For our first step, we prove the following result, showing that an optimal derived classifier can always be found efficiently given "perfect" knowledge of a context and a prior classifier:

Claim 2. Let $\mathcal{C}$ be an arbitrary classifier over context $\mathcal{P} = (\mathcal{D}, f, g, O)$. Then there exists a set of polynomially many (in $|\Psi_{\mathcal{P}}|$, $|\mathbb{G}_{\mathcal{P}}|$, and $|\Omega_{\mathcal{P}}^{\mathcal{C}}|$) satisfiable linear constraints in the variables $P_{i,j}^{g} = \Pr_g\left[\mathcal{C}'(\sigma) = j|\mathcal{C}(\sigma) = i\right]$ that define the set of $\mathcal{C}$-derived classifiers $\mathcal{C}'$ which satisfy ϵ-fair treatment with respect to $\mathcal{P}$.

Corollary 1. Let $\mathcal{C}$ be an arbitrary classifier over context $\mathcal{P} = (\mathcal{D}, f, g, O)$, and let $\ell_{\mathcal{D},f,\mathcal{C}}$ be a loss function which is linear over the probabilities $P_{i,j}^{g}$ as defined above. Then the $\mathcal{C}$-derived $\mathcal{C}'$ which minimizes $\ell_{\mathcal{D},f,\mathcal{C}}(\cdot)$ while satisfying ϵ-fair treatment with respect to $\mathcal{P}$ can be found efficiently (i.e., in time polynomial in $|\Psi_{\mathcal{P}}|$, $|\mathbb{G}_{\mathcal{P}}|$, and $|\Omega_{\mathcal{P}}^{\mathcal{C}}|$).[9]

The corollary will follow immediately from Claim 2 by the efficiency of solving linear programs (that is, the well-known fact that a linear program with a polynomial number of variables and constraints may be solved in polynomial time). We now prove Claim 2:

Proof. Assume we have a discrete classifier $\mathcal{C}$ that classifies individuals from a context $\mathcal{P} = (\mathcal{D}, f, g, O)$, and we wish to produce $\mathcal{C}'$ that satisfies ϵ-fair treatment with respect to $\mathcal{P}$. Consider the $\mathcal{C}$-derived classifier defined by the set of $|\mathbb{G}_{\mathcal{P}}||\Omega_{\mathcal{P}}^{\mathcal{C}}|^2$ variables

$$P_{i,j}^{g} = \Pr_g\left[\mathcal{C}'_j|\mathcal{C}_i\right]$$

for $i, j \in \Omega_{\mathcal{P}}^{\mathcal{C}}$ and $g \in \mathbb{G}_{\mathcal{P}}$.

Next, we directly translate the definition of fair treatment into a set of constraints, which represents the space of all possible derived classifiers satisfying ϵ-fair treatment:

$$\begin{aligned} \forall i, j \in \Omega_{\mathcal{P}}^{\mathcal{C}}, \forall g \in \mathbb{G}_{\mathcal{P}} : P_{i,j}^{g} \in [0,1] \\ \forall i \in \Omega_{\mathcal{P}}^{\mathcal{C}}, \forall g \in \mathbb{G}_{\mathcal{P}} : \sum_{j \in \Omega_{\mathcal{P}}^{\mathcal{C}}} P_{i,j}^{g} = 1 \\ \forall j \in \Omega_{\mathcal{P}}^{\mathcal{C}}, \forall k \in \Psi_{\mathcal{P}}, \forall X, Y \in \mathbb{G}_{\mathcal{P}} : \Pr_X\left[\mathcal{C}'_j|f_k\right] \leq e^{\epsilon}\Pr_Y\left[\mathcal{C}'_j|f_k\right] \end{aligned}$$

Notice, however, that:

$$\Pr_g\left[\mathcal{C}'_j|f_k\right] = \frac{1}{\Pr_g\left[f_k\right]}\left(\Pr_g\left[f_k \wedge \mathcal{C}'_j\right]\right) = \frac{1}{\Pr_g\left[f_k\right]}\left(\sum_{i \in \Omega_{\mathcal{P}}^{\mathcal{C}}} \Pr_g\left[f_k \wedge \mathcal{C}'_j \wedge \mathcal{C}_i\right]\right)$$

As observed earlier (see the proof of Claim 1), because we assign outcomes in $\mathcal{C}'$ based only on $\mathcal{C}$ and $g(\sigma)$, it must be the case that $\Pr_g\left[\mathcal{C}'_j|\mathcal{C}_i\right] = \Pr_g\left[\mathcal{C}'_j|\mathcal{C}_i \wedge f_k\right]$, or, expanding using conditional probability,

$$\frac{\Pr_g\left[\mathcal{C}'_j \wedge \mathcal{C}_i\right]}{\Pr_g\left[\mathcal{C}_i\right]} = \frac{\Pr_g\left[f_k \wedge \mathcal{C}'_j \wedge \mathcal{C}_i\right]}{\Pr_g\left[f_k \wedge \mathcal{C}_i\right]}$$

[9] If $\ell_{\mathcal{D},f,\mathcal{C}}(\cdot)$ is not linear, it is of course findable, but not necessarily efficiently, as we no longer have a linear program.

which implies

$$\mathrm{Pr}_g\left[f_k \wedge \mathcal{C}'_j \wedge \mathcal{C}_i\right] = \frac{\mathrm{Pr}_g\left[f_k \wedge \mathcal{C}_i\right]\mathrm{Pr}_g\left[\mathcal{C}'_j \wedge \mathcal{C}_i\right]}{\mathrm{Pr}_g\left[\mathcal{C}_i\right]} = \mathrm{Pr}_g\left[f_k \wedge \mathcal{C}_i\right]\mathrm{Pr}_g\left[\mathcal{C}'_j|\mathcal{C}_i\right] = \mathrm{Pr}_g\left[f_k \wedge \mathcal{C}_i\right]P^g_{i,j}$$

So our conditions of the form $\mathrm{Pr}_X\left[\mathcal{C}'_j|f_k\right] \leq e^\epsilon \mathrm{Pr}_Y\left[\mathcal{C}'_j|f_k\right]$ can be rewritten (after substituting and multiplying through) as

$$\mathrm{Pr}_Y\left[f_k\right]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathrm{Pr}_X\left[f_k \wedge \mathcal{C}_i\right]P^X_{i,j}\right) \leq e^\epsilon \mathrm{Pr}_X\left[f_k\right]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathrm{Pr}_Y\left[f_k \wedge \mathcal{C}_i\right]P^Y_{i,j}\right)$$

We can also reformat the second set of conditions into inequality constraints by selecting $j^* \in \Omega^{\mathcal{C}}_{\mathcal{P}}$, replacing each P^g_{i,j^*} with $1 - \sum_{j\in\Omega^{\mathcal{C}}_{\mathcal{P}}\setminus j^*} P^g_{i,j}$, and requiring $\sum_{j\in\Omega^{\mathcal{C}}_{\mathcal{P}}\setminus j^*} P^g_{i,j} \leq 1$. Then our final set of constraints becomes:

$$\begin{aligned}
\forall i \in \Omega^{\mathcal{C}}_{\mathcal{P}}, \forall j \in \Omega^{\mathcal{C}}_{\mathcal{P}} \setminus j^*, \forall g \in \mathbb{G}_{\mathcal{P}} : P^g_{i,j} \geq 0, P^g_{i,j} \leq 1 \\
\forall i \in \Omega^{\mathcal{C}}_{\mathcal{P}}, \forall g \in \mathbb{G}_{\mathcal{P}} : \sum_{j\in\Omega^{\mathcal{C}}_{\mathcal{P}}\setminus j^*} P^g_{i,j} \leq 1 \\
\forall j \in \Omega^{\mathcal{C}}_{\mathcal{P}} \setminus j^*, \forall k \in \Psi_{\mathcal{P}}, \forall X, Y \in \mathbb{G}_{\mathcal{P}} : \\
\mathrm{Pr}_Y\left[f_k\right]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathrm{Pr}_X\left[f_k \wedge \mathcal{C}_i\right]P^X_{i,j}\right) \leq e^\epsilon \mathrm{Pr}_X\left[f_k\right]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathrm{Pr}_Y\left[f_k \wedge \mathcal{C}_i\right]P^Y_{i,j}\right) \\
\forall k \in \Psi_{\mathcal{P}}, \forall X, Y \in \mathbb{G}_{\mathcal{P}} : \\
\mathrm{Pr}_Y\left[f_k\right]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathrm{Pr}_X\left[f_k \wedge \mathcal{C}_i\right]\left(1 - \sum_{j\in\Omega^{\mathcal{C}}_{\mathcal{P}}\setminus j^*} P^X_{i,j}\right)\right) \leq e^\epsilon \mathrm{Pr}_X\left[f_k\right]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathrm{Pr}_Y\left[f_k \wedge \mathcal{C}_i\right]\left(1 - \sum_{j\in\Omega^{\mathcal{C}}_{\mathcal{P}}\setminus j^*} P^Y_{i,j}\right)\right)
\end{aligned}$$

which is a system of $2|\mathbb{G}_{\mathcal{P}}||\Omega^{\mathcal{C}}_{\mathcal{P}}|^2 + |\mathbb{G}_{\mathcal{P}}|^2|\Omega^{\mathcal{C}}_{\mathcal{P}}||\Psi_{\mathcal{P}}|$ equations in $|\mathbb{G}_{\mathcal{P}}||\Omega^{\mathcal{C}}_{\mathcal{P}}|(|\Omega^{\mathcal{C}}_{\mathcal{P}}| - 1)$ variables.

Furthermore, we know that this system necessarily has a solution on its domain, since taking $P^g_{i,j} = 1/|\Omega^{\mathcal{C}}_{\mathcal{P}}|$ for each i, j, and g corresponds to a classifier $\mathcal{C}'$ where all individuals are offered a uniform distribution over outcomes; this classifier trivially satisfies fair treatment (and indeed, one can easily verify that it satisfies the above conditions for any $\mathcal{C}$ and $\mathcal{P}$). □

Thus, finding assignments for $P_{i,j}$ in order to construct a classifier $\mathcal{C}'$ satisfying fair treatment with respect to $\mathcal{C}$ becomes a linear optimization problem—that is, to find an assignment that satisfies the sets of conditions above while minimizing some linear loss function.

5.2 Step 2: Approximate Fairness From Experimental Probabilities

Of course, we have only established so far that $\mathcal{C}'$ constructed in such a manner satisfies fair treatment if we already know the exact probabilities $\mathrm{Pr}_g\left[f_k\right]$ and $\mathrm{Pr}_g\left[f_k \wedge \mathcal{C}_i\right]$ for each group g. This of course requires non-black-box knowledge of $\mathcal{P}$ and $\mathcal{C}$; however, we will now show by a Chernoff bound that, assuming $\mathcal{B}$ is given *experimental* probabilities $\mathrm{Pr}_g\left[f_k\right]$ and $\mathrm{Pr}_g\left[f_k \wedge \mathcal{C}_i\right]$ from a sufficiently large "training set" of individuals randomly drawn from the distribution $\tau_{\chi,\Pi}(1^n)$, $\mathcal{C}'$ constructed according to the above linear program, and with a small amount of random noise added to prevent interference due to experimental variance in observing extremely rare events, will still satisfy ϵ-*approximate* fair treatment with overwhelming probability. Specifically, it can be proven that the probability of $\mathcal{C}'$ *not* satisfying approximate fair treatment is extremely small given a sufficiently large number of random samples (i.e., a number inversely polynomial in the desired fair treatment error ϵ).

To formalize what we mean by adding "a small amount of random noise", given some derived classifier $\mathcal{C}'$ (which we recall can be expressed as an $|\Omega_{\mathcal{P}}^{\mathcal{C}}| \times |\Omega_{\mathcal{P}}^{\mathcal{C}}|$ perturbation matrix), and letting $(\mathbf{1})_{m \times n}$ be an $m \times n$ matrix of all ones, we shall let

$$Q_r(\mathcal{C}') \triangleq \frac{r}{|\Omega_{\mathcal{P}}^{\mathcal{C}}|}(\mathbf{1})_{|\Omega_{\mathcal{P}}^{\mathcal{C}}| \times |\Omega_{\mathcal{P}}^{\mathcal{C}}|} + (1-r)\mathcal{C}'$$

be the derived classifier that with probability r outputs a random outcome and otherwise outputs an outcome according to the classifier $\mathcal{C}'$. (Hence, $Q_r(\mathcal{C}')(\sigma)$ is identical to $\mathcal{C}'(\sigma)$ with probability $1-r$.)

We will herein make use of the following well-known bound (for ease of notation, we denote $\exp(x) = e^x$):

Lemma 1. (Hoeffding Bound.) Let $X_1, \ldots, X_N$ be independent binary random variables (i.e., $X_i \in \{0, 1\}$). Let m be the expected value of their average and X^* their actual average. Then, for any $\delta \in (0, 1)$:

$$\Pr\left[|X^* - m| \geq \delta\right] \leq 2 \exp\left(-2\delta^2 N\right)$$

In particular, when δ and m are fixed, this probability is inversely exponential (i.e., *negligible*) in the number of random variables N. To take advantage of this, consider our scenario where we have some classifier $\mathcal{C}$ trained using some number of individuals drawn (independently) from the distribution from the distribution $\tau_{\chi,\Pi}(1^n)$, and we wish to measure the probability of some event E_1 occurring conditioned on a subgroup g. Notationally, we will henceforth denote by $\mathsf{Ex}[E]$ the experimental probability of an event E over a set of random samples—i.e., letting $\mathcal{S}$ be the set of samples and $\mathbf{1}_{E(s)}$ the indicator variable which is 1 if E is true for a sample s and 0 if not:

$$\frac{1}{|\mathcal{S}|}\sum_{s \in \mathcal{S}} \mathbf{1}_{E(s)}$$

We will denote by $\mathsf{Ex}_g[E]$ the experimental probability of E conditioned on a group g, or $\mathsf{Ex}[E \wedge g]/\mathsf{Ex}[g]$. Then we prove the following lemma:

Lemma 2. Given a distribution $\mathcal{D}$, event E, and group g, then, letting Ex denote the experimental probability as derived from N independent samples from the distribution $\tau_{\chi,\Pi}(1^n)$, for any $\delta \in (0, 1)$, with probability at least $1 - 4 \exp\left(-2\left(\frac{\delta \Pr[g]}{3}\right)^2 N\right)$ over the samples, the following two conditions hold:

1. $|\mathsf{Ex}_g[E] - \Pr_g[E]| < \delta$
2. $|\mathsf{Ex}[g] - \Pr[g]| < \delta$

Specifically, this states that the probability of the experimental and real probabilities diverging for some fixed event E is inverse-exponential in the size of $\mathcal{C}$'s training set.

Proof. First we prove the following claim:

Claim 3. Given positive real numbers a, b, c, d, ϵ such that $|a-b| < \epsilon$ and $|c-d| < \epsilon$, then

$$\left|\frac{a}{c} - \frac{b}{d}\right| < \frac{(a+c)\epsilon}{c(c-\epsilon)}$$

Proof. The following three facts suffice:

$$\left|\frac{a}{c} - \frac{b}{d}\right| = \frac{1}{cd}|ad - bc|$$

$$\frac{1}{cd} < \frac{1}{c(c-\epsilon)}$$

$$|ad - bc| < |a(c+\epsilon) - (a-\epsilon)c| = \epsilon(a+c)$$

□

So, as long as $|\mathsf{Ex}[g] - \Pr[g]| < \delta$ and $|\mathsf{Ex}[E \wedge g] - \Pr[E \wedge g]| < \delta$, then

$$|\mathsf{Ex}_g[E] - \Pr_g[E]| = \left|\frac{\Pr[E \wedge g]}{\Pr[g]} - \frac{\mathsf{Ex}[E \wedge g]}{\mathsf{Ex}[g]}\right| < \frac{(\Pr[E \wedge g] + \Pr[g])\delta}{\Pr[g](\Pr[g] - \delta)}$$

which means that, by Lemma 1,

$$\Pr\left[|\mathsf{Ex}_g[E] - \Pr_g[E]| \geq \frac{(\Pr[E \wedge g] + \Pr[g])\delta}{\Pr[g](\Pr[g] - \delta)}\right]$$

$$\leq \Pr[|\mathsf{Ex}[g] - \Pr[g]| \geq \delta] + \Pr[|\mathsf{Ex}[E \wedge g] - \Pr[E \wedge g]| \geq \delta] \leq 4\exp\left(-2\delta^2 N\right)$$

This follows because, for each of the (unconditioned) probabilities in question, we can use a Chernoff bound with N variables $X_1, \ldots, X_n$ equal to 1 if the respective event occurs for a sampled individual and 0 otherwise; then X^* is equal to the experimental probability of the event and m (its expectation) is by definition equal to the actual probability.

Finally, let

$$\delta' = \frac{(\Pr[E \wedge g] + \Pr[g])\delta}{\Pr[g](\Pr[g] - \delta)} = \frac{(\Pr_g[E] + 1)\delta}{\Pr[g] - \delta}$$

Then

$$\delta'(\Pr[g] - \delta) = (\Pr_g[E] + 1)\delta$$

$$\delta'\Pr[g] = (\Pr_g[E] + 1 + \delta')\delta$$

$$\frac{\delta'\Pr[g]}{\Pr_g[E] + 1 + \delta'} = \delta$$

And so

$$\Pr\left[|\mathsf{Ex}_g[E] - \Pr_g[E]| \geq \delta'\right] \leq 4\exp\left(-2\delta^2 N\right) = 4\exp\left(-2\left(\frac{\delta'\Pr[g]}{\Pr_g[E] + 1 + \delta'}\right)^2 N\right)$$

$$\leq 4\exp\left(-2\left(\frac{\delta'\Pr[g]}{3}\right)^2 N\right)$$

since $\delta' < 1$ by assumption and $\Pr_g[E] \leq 1$ trivially. Furthermore, when we show that $|\mathsf{Ex}_g[E] - \Pr_g[E]| < \delta'$, we do so by showing that

$$|\Pr[g] - \mathsf{Ex}[g]| \leq [\delta =] \frac{\delta' \Pr[g]}{\Pr_g[E] + 1 + \delta'} \leq \delta'$$

which completes the other part of the argument. □

Now we can prove our key lemmas using this consequence.

Lemma 3. Given context $\mathcal{P} = (\mathcal{D}, f, g, O)$ and $\epsilon \in (0,1)$, let $\mathcal{C}'$ be a $\mathcal{C}$-derived classifier satisfying a modification of the linear constraints in Corollary 1 for $(\epsilon^2/4)$-fair treatment where the coefficients are determined by the *experimental* (rather than actual) probabilities of the respective events given N random samples $(O(\sigma), f(\sigma), \mathcal{C}(\sigma), g(\sigma))$ from the distribution $\tau_{\mathcal{X},\Pi}(1^n)$. Then the classifier $Q_{2\epsilon|\Omega^{\mathcal{C}}_{\mathcal{P}}|/3}(\mathcal{C}')$ satisfies ϵ-approximate fair treatment with respect to $\mathcal{P}$ except with probability negligible in N over the selection of samples—in particular, with probability $1 - O(e^{(-c\epsilon^4 N)})$ for some constant c dependent only on $\mathcal{D}$.

Proof. Let $c = \frac{2}{144^2} \min_g \Pr[g]^2$. Notice that c is not dependent on n or, for that matter, on anything besides the (fixed) distribution $\mathcal{D}$.

First let us consider the classifier $\mathcal{C}'$ before noise is added. Because $\mathcal{C}'$ is derived from $\mathcal{C}$ according to Corollary 1, we have, by the respective constraints for fair treatment for each $j \in \Omega^{\mathcal{C}}_{\mathcal{P}}$, $X, Y \in \mathbb{G}_{\mathcal{P}}$, and $k \in \Psi_{\mathcal{P}}$:

$$\mu\left(\mathsf{Ex}_Y[f_k]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathsf{Ex}_X[f_k \wedge \mathcal{C}_i] P^X_{i,j}\right), \mathsf{Ex}_X[f_k]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathsf{Ex}_Y[f_k \wedge \mathcal{C}_i] P^Y_{i,j}\right)\right) \leq \frac{\epsilon^2}{4}$$

which, since both sides are at most 1 and thus can differ additively by at most $1 - e^{-\epsilon^2/4} \leq \epsilon^2/4$, implies:

$$\left|\mathsf{Ex}_Y[f_k]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathsf{Ex}_X[f_k \wedge \mathcal{C}_i] P^X_{i,j}\right) - \mathsf{Ex}_X[f_k]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathsf{Ex}_Y[f_k \wedge \mathcal{C}_i] P^Y_{i,j}\right)\right| \leq \frac{\epsilon^2}{4}$$

where $P^X_{i,j}$ and $P^Y_{i,j}$ are derived from solving the constraints. Applying Lemma 2 (1) once for each $k \in \Psi_{\mathcal{P}}$ to the event f_k and group Y (with $\delta = \epsilon^2/48$) then gives us that

$$\left|\Pr_Y[f_k]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathsf{Ex}_X[f_k \wedge \mathcal{C}_i] P^X_{i,j}\right) - \mathsf{Ex}_X[f_k]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathsf{Ex}_Y[f_k \wedge \mathcal{C}_i] P^Y_{i,j}\right)\right| \leq \frac{\epsilon^2}{4} + \frac{\epsilon^2}{48}$$

except with probability no greater than

$$4 \exp\left(-2\left(\frac{(\epsilon^2/48)\Pr[Y]}{3}\right)^2 N\right) \leq 4 \exp\left(-2\left(\frac{\epsilon^2(\min_g \Pr[g])}{144}\right)^2 N\right)$$

$$= 4 \exp\left(-\left(\frac{2\epsilon^4(\min_g \Pr[g])^2}{144^2}\right) N\right) = 4(\exp(-c\epsilon^4 N))$$

for each choice of k, or, over all of the $|\Psi_{\mathcal{P}}|$ choices of k, no greater than $4|\Psi_{\mathcal{P}}|(\exp(-c\epsilon^4 N))$ by the union bound. Symmetrically for each event f_k and group X:

$$\left|\mathrm{Pr}_Y\left[f_k\right]\left(\sum_{i\in\Omega_{\mathcal{P}}^{\mathcal{C}}}\mathsf{Ex}_X[f_k\wedge\mathcal{C}_i]P_{i,j}^X\right)-\mathrm{Pr}_X\left[f_k\right]\left(\sum_{i\in\Omega_{\mathcal{P}}^{\mathcal{C}}}\mathsf{Ex}_Y[f_k\wedge\mathcal{C}_i]P_{i,j}^Y\right)\right|\leq\frac{\epsilon^2}{4}+\frac{\epsilon^2}{24}$$

except with the same failure probability. We then do the same for the events $f_k \wedge \mathcal{C}_i$ (for each of the $|\Psi_{\mathcal{P}}|$ choices of k) conditioned on X and Y to obtain that

$$\left|\mathrm{Pr}_Y\left[f_k\right]\left(\sum_{i\in\Omega_{\mathcal{P}}^{\mathcal{C}}}\mathrm{Pr}_X\left[f_k\wedge\mathcal{C}_i\right]P_{i,j}^X\right)-\mathrm{Pr}_X\left[f_k\right]\left(\sum_{i\in\Omega_{\mathcal{P}}^{\mathcal{C}}}\mathrm{Pr}_Y\left[f_k\wedge\mathcal{C}_i\right]P_{i,j}^Y\right)\right|\leq\frac{\epsilon^2}{4}+\frac{\epsilon^2}{12}=\frac{\epsilon^2}{3}$$

except with probability $16(\exp(-c\epsilon^4 N))$ for each choice of k (or $16|\Psi_{\mathcal{P}}|(\exp(-c\epsilon^4 N))$ overall). By the union bound over all classes $k \in \Psi_{\mathcal{P}}$ and over all (fewer than $|\mathbb{G}_{\mathcal{P}}|^2$) pairs of groups X and Y, the total failure probability from applying these bounds to all of the constraints is at most $16|\mathbb{G}_{\mathcal{P}}|^2|\Psi_{\mathcal{P}}|(\exp(-c\epsilon^4 N)) = O(\exp(-c\epsilon^4 N))$, which is of course negligible in the number of samples N. So, with probability at least $1 - O(\exp(-c\epsilon^4 N))$ over the drawn samples, all of the above constraints will hold.

This is not quite identical to the statement

$$\mu\left(\mathrm{Pr}_X\left[\mathcal{C}'_j|f_k\right],\mathrm{Pr}_Y\left[\mathcal{C}'_j|f_k\right]\right)\leq\epsilon$$

(i.e., fair treatment for $\mathcal{C}'$); particularly, if the probability of some outcome is very small, then a bound on the *additive* distance between real and experimental probabilities has no impact on whether the *multiplicative* distance is bounded. To overcome this issue, we will consider the classifier $Q_{2|\Omega_{\mathcal{P}}^{\mathcal{C}}|\epsilon/3}(\mathcal{C}')$ that, as defined above, runs $\mathcal{C}'$ and outputs the result *except* with probability $2|\Omega_{\mathcal{P}}^{\mathcal{C}}|\epsilon/3$, in which case it will pick an output uniformly at random. This guarantees that the probability of any outcome occurring (even conditioned on any group) must be at least $2\epsilon/3$; hence, except with the aforementioned failure probability, the multiplicative distance between the real and experimental probabilities for any such conditional outcome can be at most either

$$\ln\left(\frac{2\epsilon/3+\epsilon^2/3}{2\epsilon/3}\right)=\ln\left(1+\epsilon/2\right)\leq\epsilon$$

or

$$\ln\left(\frac{2\epsilon/3}{2\epsilon/3-\epsilon^2/3}\right)=\ln\left(\frac{1}{1-\epsilon/2}\right)\leq\epsilon$$

for all $\epsilon < 1$. □

Remark. While it may seem counterintuitive for the classifier output by our sanitizer to output a uniformly random class with small probability, in fact this "random noise" is only necessary due to the possibility of arbitrarily small probabilities $\mathrm{Pr}_g\left[f_k \wedge \mathcal{C}_i\right]$ occurring in the distribution $\mathcal{D}$; specifically, if some such event occurs with small enough probability, it would likely be measured to have probability 0, potentially causing an unbounded multiplicative fairness error in the derived classifier. If there instead exists a constant lower bound for these probabilities (or even, once parameterized, an asymptotic lower bound of $\epsilon(n)$), then we can directly obtain the result above without having to add noise to the outcome of the derived classifier, as we do in our experimental results (Section 6).

Importantly, we can also apply Lemma 3 in reverse, transforming from the exact conditions to the modified conditions with experimental probabilities, under precisely the same conditions. This will be useful to demonstrate optimality (i.e., that the optimal fair classifier is derivable by $\mathcal{B}$ as it is overwhelmingly likely to satisfy approximate versions of the constraints) in the following section.

Lemma 4. Given context $\mathcal{P} = (\mathcal{D}, f, g, O)$, let $\mathcal{C}'$ be a $\mathcal{C}$-derived classifier satisfying the conditions in Corollary 1 for 0-fair treatment with respect to $\mathcal{P}$. Then, for any $\epsilon \in (0,1)$, the classifier $Q_{\epsilon^2|\Omega^{\mathcal{C}}_{\mathcal{P}}|/4}(\mathcal{C}')$, with probability $1 - O(e^{(-c\epsilon^8 N)})$ (for some constant c dependent only on $\mathcal{D}$) over N random samples $(O(\sigma), f(\sigma), \mathcal{C}(\sigma), g(\sigma))$ from the distribution $\tau_{\chi,\Pi}(1^n)$, satisfies the modification of the linear constraints in Corollary 1 for $(\epsilon^2/4)$-fair treatment where the coefficients are determined by the *experimental* (rather than actual) probabilities of the respective events given the random samples.

Proof. We proceed very similarly to Lemma 3, except changing the error parameter ϵ and reversing $\mathsf{Ex}[\ldots]$ with $\Pr[\ldots]$. Since we know that $\mathcal{C}'$ satisfies perfect fair treatment, we have, this time with respect to the real probabilities:

$$\left|\Pr_Y[f_k]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \Pr_X[f_k \wedge \mathcal{C}_i]\, P^X_{i,j}\right) - \Pr_X[f_k]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \Pr_Y[f_k \wedge \mathcal{C}_i]\, P^Y_{i,j}\right)\right| = 0$$

Next we apply Lemma 2 (1) with $\delta = \epsilon^4/128$ to all events f_k and $f_k \wedge \mathcal{C}_i$ for groups X and Y just as in Lemma 3, obtaining that

$$\left|\mathsf{Ex}_Y[f_k]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathsf{Ex}_X[f_k \wedge \mathcal{C}_i]P^X_{i,j}\right) - \mathsf{Ex}_X[f_k]\left(\sum_{i\in\Omega^{\mathcal{C}}_{\mathcal{P}}} \mathsf{Ex}_Y[f_k \wedge \mathcal{C}_i]P^Y_{i,j}\right)\right| \le 4\left(\frac{\epsilon^4}{128}\right) = \frac{\epsilon^4}{32}$$

except with probability $O(e^{(-c\epsilon^8 N)})$ over the N samples taken (for some small constant $\mathcal{C}$ dependent only on $\mathcal{D}$). To convert this into multiplicative distance, we use the classifier $Q_{\epsilon^2|\Omega^{\mathcal{C}}_{\mathcal{P}}|/4}(\mathcal{C}')$ so that the probability of any outcome is at least $\epsilon^2/4$. Then, as long as the conditions of Lemma 2 are true, the multiplicative distance between the real and experimental probabilities for any such conditional outcome can be at most either

$$\ln\left(\frac{\epsilon^2/4 + \epsilon^4/32}{\epsilon^2/4}\right) = \ln\left(1 + \epsilon^2/8\right) \le \epsilon^2/4$$

or

$$\ln\left(\frac{\epsilon^2/4}{\epsilon^2/4 - \epsilon^4/32}\right) = \ln\left(\frac{1}{1-\epsilon^2/8}\right) \le \epsilon^2/4$$

for all $\epsilon < 1$. □

5.3 Step 3: Optimality over Derived Classifiers

Now we can construct an $\epsilon(\cdot)$-black box sanitizer for any inverse polynomial $\epsilon(n)$ using Corollary 1 and Lemma 3. In particular, given some context ensemble $\Pi = \{(\mathcal{D}, f, g, O_n)\}_{n\in\mathbb{N}}$ and a sequence of classifiers $\chi = \{\mathcal{C}_n\}_{n\in\mathbb{N}}$, if, for any n, we use Corollary 1 on experimental probabilities (given enough samples from $\tau_{\chi,\Pi}(1^n)$) to produce a $\mathcal{C}_n$-derived classifier which is fair with respect to those probabilities, Lemma 3 allows us to assert that a slightly noisy version of the resulting classifier is still approximately fair, even though we only have black-box access to the training data set $\tau_{\chi,\Pi}(1^n)$ (whereas notably our original formulation in Corollary 1 requires non-black-box access to determine the exact probabilities $\Pr_g[f_k]$ and $\Pr_g[f_k \wedge \mathcal{C}_i]$ for the optimization constraints). We propose the following construction and subsequently prove its correctness as a black-box sanitizer, amounting to the first part (existence) of the proof of Theorem 4:

Constructing the Black-Box Sanitizer. Consider the following algorithm for $\mathcal{B}_{\tau_{\chi,\Pi}}$ on input 1^n, where we assume some farness parameter $\epsilon(n) \geq \frac{1}{q(n)}$ for polynomial $q(\cdot)$ and some loss function $\ell_{\mathcal{D},f,\mathcal{C}}(\cdot)$ which is linear in the probabilities $P^g_{i,j}$ but may depend on probabilities observed in $\mathcal{D}$, f, and $\mathcal{C}$:

- *(Estimating constraints by sampling.)* Use queries to $\tau_{\chi,\Pi}(1^n)$ to produce (for some $\epsilon' > 0$ and polynomial $p(n) = \Omega(q(n)^{8+\epsilon'})$) $N = p(n)$ samples $(O_n(\sigma'), f(\sigma'), \mathcal{C}_n(O_n(\sigma')), g(\sigma'))$ for $\sigma' \leftarrow \mathcal{D}$, so that the failure probabilities described in both Lemmas 3 and 4 are negligible in n. (In particular, this failure probability will be at most $O(e^{-cp(n)/q(n)^8}) = O(e^{-cn^{\epsilon'}})$, which is negligible since c depends only on the fixed distribution $\mathcal{D}$.)[10]
- *(Estimating the loss function.)* Furthermore, use the experimental probabilities of the samples to estimate any distributionally-dependent parameters of the loss function ℓ. Call the approximate loss function $\ell'(\cdot)$.
- *(Solving the derived constraints.)* Use Corollary 1 to produce probabilities $P^g_{i,j}$ for a $\mathcal{C}_n$-derived classifier which minimizes $\ell'(\cdot)$ with respect to the constraints for $(\epsilon(n)^2/4)$-fair treatment generated from the experimental probabilities $\mathsf{Ex}_g[f_k]$ and $\mathsf{Ex}_g[f_k \wedge \mathcal{C}_i]$ over the N samples.
- *(Adding noise and producing the derived classifier.)* Output the $\mathcal{C}_n$-derived classifier $\mathcal{C}'' = Q_{2\epsilon(n)|\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|/3}(\mathcal{C}')$ (which with probability $2\epsilon(n)|\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|/3$ outputs a uniformly random element of $\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}$, and which otherwise uses the probabilities $P^g_{i,j}$ found from the optimization to classify σ according to $\mathcal{C}_n(O_n(\sigma))$ and σ's group $g(\sigma)$—i.e., draws from the distribution $\{j \text{ with pr. } P^g_{\mathcal{C}_n(\sigma),j}\}$).

Claim 4. For any $\epsilon(n) \geq \frac{1}{q(n)}$ for polynomial $q(\cdot)$, the above construction of $\mathcal{B}_{(\cdot)}$ is an $\epsilon(\cdot)$-black-box sanitizer.

Proof. By Lemma 3, the classifier $\mathcal{C}'' = Q_{2\epsilon(n)|\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|/3}(\mathcal{C}')$ output by $\mathcal{B}$ satisfies $\epsilon(n)$-fair treatment with probability at least $1 - \nu(n)$ (where $\nu(\cdot)$ is negligible) for any given n.

Furthermore, we note that the algorithm for $\mathcal{B}$ is efficient; for any context ensemble Π and classifier sequence χ such that $|\mathbb{G}_{\mathcal{P}_n}| \leq m$, $|\Psi_{\mathcal{P}_n}| \leq m$, and $|\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}| \leq m$, it runs in time polynomial in m and polynomial in n. The former bound comes from the running time of the linear program in Corollary 1, and the latter bound comes from Lemma 3 and the fact that we make $N = p(n)$ oracle queries to gather "training data". Hence $\mathcal{B}_{(\cdot)}$ must be an $\epsilon(\cdot)$-black-box sanitizer. □

[10] We use $\omega(q(n)^8)$ samples so that we can later assert that Lemma 4 holds with all-but-negligible probability in the optimality step. For the current step, only $\omega(q(n)^4)$ samples are necessary.

Notably, the running time of this algorithm is proportional to $\frac{1}{\epsilon(n)^8}$, which is natural in that, to derive a more accurate approximation of the real probabilities with training data, more samples are required. (In fact, as we shall show, decreasing ϵ and/or respectively increasing the number of samples will reduce both the fairness and optimality errors.)

Finally, we remark on the loss function $\ell_{\mathcal{D},f,\mathcal{C}}(\cdot)$ and the optimality of our construction. Of course, the entries of $\vec{\ell}_{\mathcal{D},f,\mathcal{C}}$—that is, the probabilities $\Pr[g]$ and $\Pr_g[f(\sigma) = k \wedge \mathcal{C}(\sigma) = i]$—are in general unknown to the black-box sanitizer $\mathcal{B}$, and this is why our construction uses its training samples to also calculate the experimental probabilities needed to approximate the loss function. Now we will show that using the experimentally derived loss function (naturally) increases the error bound of $\mathcal{C}''$, but only slightly (albeit dependent on the degree and number of terms of the coefficients of $P^g_{i,j}$ in ℓ). The following claim essentially states that, as the optimum of a linear loss function changes at most minimally if the coefficients change minimally, the loss of the classifier output by $\mathcal{B}$ over the predicted loss function will not be much worse than the loss over the correct loss function. This fact, combined with the fact that (a slightly noisy version of) the optimal perfectly fair classifier can always be derived by $\mathcal{B}$ if it knows the correct loss function, suffices to show that the classifier actually derived by $\mathcal{B}$ will not be much worse than the optimal fair classifier, hence proving the final part of Theorem 4.

Claim 5. With probability at least $1 - \nu(n)$ (for negligible $\nu(\cdot)$) over $\mathcal{B}$'s queries, the $\mathcal{C}''$ output by $\mathcal{B}_{\tau_{\chi,\Pi}}(1^n)$ constructed above has error

$$\Delta_{\ell,\mathcal{D},f,\mathcal{C}}(\mathcal{C}'') \leq |\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|(\epsilon(n) + |\mathbb{G}_{\mathcal{P}_n}|\epsilon(n)^4 dt/64)$$

with respect to any linear loss function with t-term coefficients of degree d given by $\ell_{\mathcal{D},f,\mathcal{C}}(\mathcal{C}'')$.

Proof. Herein we shall for consistency refer to the loss function optimized by $\mathcal{B}$ by deriving from the experimental probabilities as $\ell'(\cdot)$, and the "true" loss function as $\ell(\cdot)$.

Let $\mathcal{C}^*$ be the optimal perfectly fair $\mathcal{C}_n$-derived classifier, let $\mathcal{C}^{**} \triangleq Q_{\epsilon(n)^2|\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|/4}(\mathcal{C}^*)$ be a noisy version of $\mathcal{C}^*$, and, as in the construction of $\mathcal{B}$, let $\mathcal{C}'$ be the classifier that optimizes ℓ' over the experimentally derived constraints and $\mathcal{C}'' = Q_{2\epsilon(n)|\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|/3}(\mathcal{C}')$ the noisy version of $\mathcal{C}'$. Towards bounding the quantity $\ell(\mathcal{C}'') - \ell(\mathcal{C}^*)$ and thus the error, we bound the difference in loss between successive pairs of classifiers:

- Beginning with $\mathcal{C}''$, the actual output, we notice that the difference in loss between $\mathcal{C}'$ and $\mathcal{C}''$ must be small because $\mathcal{C}''$ is by definition identical to $\mathcal{C}'$ except with small probability.

- Next, we can bound the difference in loss between $\mathcal{C}^{**}$ and $\mathcal{C}'$ by noticing that Lemma 4 provides that $\mathcal{C}^{**}$ with high probability satisfies $(\epsilon^2/4)$-fair treatment with respect to the experimentally derived constraints and can thus be derived by $\mathcal{B}$. So this means that $\mathcal{B}$ must find a classifier which is as good as $\mathcal{C}^{**}$ or better with respect to ℓ'; by analyzing the similarity between ℓ and ℓ' we can also conclude that $\mathcal{C}^{**}$ is not much better than $\mathcal{C}'$ in terms of the true loss function ℓ.

- Finally, the difference in loss between $\mathcal{C}^*$ and $\mathcal{C}^{**}$ is once again bounded by the fact that $\mathcal{C}^{**}$ is nearly identical to $\mathcal{C}^*$.

Formally, we present the following subclaims:

Subclaim 1. $\ell(\mathcal{C}'') - \ell(\mathcal{C}') \leq 2\epsilon(n)|\Omega_{\mathcal{P}_n}^{\mathcal{C}_n}|/3$ with probability 1.

Proof. $\mathcal{C}''$ is identical to $\mathcal{C}'$ except with probability $2\epsilon(n)|\Omega_{\mathcal{P}_n}^{\mathcal{C}_n}|/3$ (i.e., no probability $P_{i,j}$ can differ between the two by more than that amount). As such, since the loss function ℓ is bounded in $[0,1]$ by assumption and linear in the probabilities $P_{i,j}^g$, the subclaim follows by linearity. Formally:

$$\ell(Q_r(\mathcal{C}')) = \ell\left(\frac{r}{|\Omega_{\mathcal{P}}^{\mathcal{C}}|}(\mathbf{1})_{|\Omega_{\mathcal{P}}^{\mathcal{C}}|\times|\Omega_{\mathcal{P}}^{\mathcal{C}}|} + (1-r)\mathcal{C}'\right) = r\ell\left(\frac{1}{|\Omega_{\mathcal{P}}^{\mathcal{C}}|}(\mathbf{1})_{|\Omega_{\mathcal{P}}^{\mathcal{C}}|\times|\Omega_{\mathcal{P}}^{\mathcal{C}}|}\right) + (1-r)\ell(\mathcal{C}')$$

and so:

$$\ell(Q_r(\mathcal{C}')) - \ell(\mathcal{C}') = r\ell\left(\frac{1}{|\Omega_{\mathcal{P}}^{\mathcal{C}}|}(\mathbf{1})_{|\Omega_{\mathcal{P}}^{\mathcal{C}}|\times|\Omega_{\mathcal{P}}^{\mathcal{C}}|}\right) - r\ell(\mathcal{C}') \leq r(1-0) = r$$

□

Subclaim 2. $\ell(\mathcal{C}')-\ell(\mathcal{C}^{**}) \leq |\Omega_{\mathcal{P}_n}^{\mathcal{C}_n}||\mathbb{G}_{\mathcal{P}_n}|dt\epsilon(n)^4/64$ with probability at least $1-\nu(n)$ (for negligible $\nu(\cdot)$) over $\mathcal{B}$'s queries.

Proof. We show this through three lemmas.

First, it is important to observe how far the experimental loss function ℓ' might be from the real function ℓ. Denote by $\ell_{i,j}^g$ the entry of $\vec{\ell}$ corresponding to the coefficient of $P_{i,j}^g$ (resp. for $\vec{\ell'}$). Then:

Lemma 5. With probability $1-\nu'(n)$ (for negligible $\nu'(\cdot)$), if ℓ is a linear loss function with t-term coefficients of degree d, then for any i, j, g it is true that $\left|\ell'^g_{i,j} - \ell^g_{i,j}\right| \leq dt\epsilon(n)^4/128$.

Proof. By Lemma 2 for each event $f_k \wedge \mathcal{C}_i$ and each group g with $\delta = \epsilon(n)^4/128$, we have $|\mathsf{Ex}_g[f_k \wedge \mathcal{C}_i] - \Pr_g[f_k \wedge \mathcal{C}_i]| < \epsilon(n)^4/128$ and $|\Pr[g] - \mathsf{Ex}[g]| < \epsilon(n)^4/128$ for any i, k, g except with probability $\nu'(n) = O(e^{(-c\epsilon(n)^8 p(n))})$ (which is negligible in n as $\mathcal{B}$ takes $p(n) = \omega(\epsilon(n)^{-8})$ samples).

As we consider loss functions whose coefficients are polynomial in the above probabilities, we can note the following identity to bound the error between the coefficients in ℓ and ℓ': if we have $x_1, \ldots, x_n, x_1', \ldots, x_n' \in [0,1]$ and $|x_i - x_i'| \leq \epsilon_i$ for each i, then[11]:

$$\left|\prod_i x_i - \prod_i x_i'\right| \leq \sum_i \epsilon_i$$

So, given some coefficient $\ell_{i,j}^g$ in the loss function which is a polynomial in the respective probabilities, the respective additive error between the real and experimental value of any degree-d monomial in that coefficient (which is bounded in $[0,1]$, i.e., does not contain a constant term greater than 1) will be at most $d\epsilon(n)^4/128$; this can be seen by taking $n = d$ in the above identity,

[11] *Proof:* If $x_1x_2 > x_1'x_2'$, then:

$$|x_1x_2 - x_1'x_2'| = x_1x_2 - x_1'x_2' < x_1(x_2' + \epsilon_2) - (x_1 - \epsilon_1)x_2' = \epsilon_2 x_1 + \epsilon_1 x_2' \leq \epsilon_1 + \epsilon_2$$

and otherwise:

$$|x_1x_2 - x_1'x_2'| = x_1'x_2' - x_1x_2 < (x_1 + \epsilon_1)x_2' - x_1(x_2' - \epsilon_2) = \epsilon_1 x_2' + \epsilon_2 x_1 \leq \epsilon_1 + \epsilon_2$$

Applying the same to x_1x_2 and x_3 gives $|(x_1x_2)x_3 - (x_1'x_2')x_3'| \leq (\epsilon_1 + \epsilon_2) + \epsilon_3$, and iteratively repeating to include all i ultimately gives the conclusion.

letting x_i represent a real probability, x'_i the corresponding experimental probability, and noting that as shown above $\epsilon_i \leq \epsilon(n)^4/128$ for each i. In turn, the coefficient itself, or the sum of t of these monomials, cannot have error greater than $dt\epsilon(n)^4/128$ (adding the error bounds from each individual monomial). So, for any variable $P^g_{i,j}$, except with the aforementioned negligible probability:

$$\left|\ell'^g_{i,j} - \ell^g_{i,j}\right| \leq dt\epsilon(n)^4/128$$

as desired. □

Next, we compare the value of the experimental loss function ℓ' between $\mathcal{C}^{**}$ and $\mathcal{C}'$, which is easily done since $\mathcal{B}$ optimizes $\mathcal{C}'$ with respect to ℓ' over a region that we can show includes $\mathcal{C}^{**}$:

Lemma 6. $\ell'(\mathcal{C}') \leq \ell'(\mathcal{C}^{**})$ with probability at least $1 - \nu''(n)$ (for negligible $\nu''(\cdot)$) over $\mathcal{B}$'s queries.

Proof. By Lemma 4, except with some negligible probability $\nu''(n)$ (again negligible since $\mathcal{B}$ takes $p(n) = \omega(\epsilon(n)^{-8})$ samples), $\mathcal{C}^{**} = Q_{\epsilon(n)^2|\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|/4}(\mathcal{C}^*)$ satisfies $\epsilon(n)^2/4$-fair treatment with respect to the *experimental* probabilities derived by $\mathcal{B}$, since $\mathcal{C}^*$ satisfies perfect (errorless) fair treatment with respect to the actual probabilities. However, recall that the $\mathcal{C}'$ recovered by $\mathcal{B}$ can by construction (Corollary 1) lie anywhere within the set of derived classifiers satisfying $\epsilon(n)^2/4$-fair treatment with respect to the same derived experimental probabilities. Since $\mathcal{B}$ optimizes ℓ' over that region, we know that, with all but the above negligible probability:

$$\ell'(\mathcal{C}') \leq \ell'(\mathcal{C}^{**})$$

as desired, because, since $\mathcal{C}^{**}$ is always findable by $\mathcal{B}$, $\mathcal{B}$ can always find either $\mathcal{C}^{**}$ itself or something with a smaller value of ℓ'. □

Finally, let $k \triangleq |\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|$ and recall the L_1-norm $||\vec{a} - \vec{b}||_1 = \sum_i (a_i - b_i)$ between two vectors. Henceforth let $(P_{\mathcal{C}'})^g_{i,j}$ denote the entry of the vector form $\vec{P}_{\mathcal{C}'}$ corresponding to $P^g_{i,j}$ for $\mathcal{C}'$, and respectively for $\mathcal{C}^{**}$. Towards relating $\ell'(\mathcal{C}') - \ell'(\mathcal{C}^{**})$ to $\ell(\mathcal{C}') - \ell(\mathcal{C}^{**})$ (the quantity we wish to bound), we show the following:

Lemma 7. $||\mathcal{C}' - \mathcal{C}^{**}||_1 \leq 2k|\mathbb{G}_{\mathcal{P}_n}|$.

Proof. Consider the $|\mathbb{G}_{\mathcal{P}_n}|k(k-1)$-dimensional space defined by the variables $P^g_{i,j}$, in which we have assumed the loss functions ℓ and ℓ' to be linear.[12] Consider moving between the points in this space which represent $\mathcal{C}^{**}$ and $\mathcal{C}'$. Each of the k sets of coordinates $(P^g_{i,1}, \ldots, P^g_{i,k-1})$ must sum to at most 1, because each set represents a probability distribution; hence, considering that moving from $\mathcal{C}^{**}$ and $\mathcal{C}'$ may decrease some number of coordinates in each such set by up to a total of 1 and correspondingly add up to a total of 1, the L_1-norm between these two points is bounded by:

$$||\mathcal{C}' - \mathcal{C}^{**}||_1 = \sum_{i,j,g} |(P_{\mathcal{C}'})^g_{i,j} - (P_{\mathcal{C}^{**}})^g_{i,j}| \leq \sum_{i,g} |1 + 1| = 2k|\mathbb{G}_{\mathcal{P}_n}|$$

This completes the argument. □

[12] While there are $2k^2$ variables in total, notice that $P^g_{i,k}$ is fully determined by $P^g_{i,1}$ through $P^g_{i,k-1}$.

Since ℓ and ℓ' are linear, we know that

$$\ell'(\mathcal{C}') - \ell'(\mathcal{C}^{**}) = \langle \vec{\ell'}, \vec{P}_{\mathcal{C}'} \rangle - \langle \vec{\ell'}, \vec{P}_{\mathcal{C}^{**}} \rangle = \langle \vec{\ell'}, \vec{P}_{\mathcal{C}'} - \vec{P}_{\mathcal{C}^{**}} \rangle = \sum_{i,j,g} \ell'^{g}_{i,j}((P_{\mathcal{C}'})^{g}_{i,j} - (P_{\mathcal{C}^{**}})^{g}_{i,j})$$

Also, using Lemma 5's bound on the difference between entries of ℓ and ℓ':

$$\ell(\mathcal{C}') - \ell(\mathcal{C}^{**}) = \sum_{i,j,g} \ell^{g}_{i,j}((P_{\mathcal{C}'})^{g}_{i,j} - (P_{\mathcal{C}^{**}})^{g}_{i,j}) \leq \sum_{i,j,g} \left(\ell'^{g}_{i,j} + \frac{dt\epsilon(n)^4}{128} \right) ((P_{\mathcal{C}'})^{g}_{i,j} - (P_{\mathcal{C}^{**}})^{g}_{i,j})$$

$$= \ell'(\mathcal{C}') - \ell'(\mathcal{C}^{**}) + \frac{dt\epsilon(n)^4}{128} \sum_{i,j,g} ((P_{\mathcal{C}'})^{g}_{i,j} - (P_{\mathcal{C}^{**}})^{g}_{i,j}) \leq 0 + \frac{dt\epsilon(n)^4}{128} ||\mathcal{C}' - \mathcal{C}^{**}||_1$$

where the final step follows because, by Lemma 6, (except with negligible probability) $\ell'(\mathcal{C}') \leq \ell'(\mathcal{C}^{**})$, or $\ell'(\mathcal{C}') - \ell'(\mathcal{C}^{**}) \leq 0$. So, using Lemma 7's bound of $2k|\mathbb{G}_{\mathcal{P}_n}|$ on the L_1-norm, we obtain that

$$\ell(\mathcal{C}') - \ell(\mathcal{C}^{**}) \leq 2k|\mathbb{G}_{\mathcal{P}_n}|(dt\epsilon(n)^4/128) = k|\mathbb{G}_{\mathcal{P}_n}|dt\epsilon(n)^4/64 = |\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}||\mathbb{G}_{\mathcal{P}_n}|dt\epsilon(n)^4/64$$

as desired, with all but negligible probability $\nu(n) \triangleq \nu'(n) + \nu''(n)$. □

Subclaim 3. $\ell(\mathcal{C}^{**}) - \ell(\mathcal{C}^*) \leq \epsilon(n)^2 |\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|/4$ with probability 1.

Proof. $\mathcal{C}^{**} = Q_{\epsilon(n)^2|\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|/4}(\mathcal{C}^*)$, so this follows by linearity, similarly to Subclaim 1. □

So, adding the differences from the above subclaims (and recalling $\epsilon(n) \leq 1$), the error of $\mathcal{C}''$ is at most:

$$\ell(\mathcal{C}'') - \ell(\mathcal{C}^*) = (\ell(\mathcal{C}'') - \ell(\mathcal{C}')) + (\ell(\mathcal{C}') - \ell(\mathcal{C}^{**})) + (\ell(\mathcal{C}^{**}) - \ell(\mathcal{C}^*))$$

$$\leq 2\epsilon(n)|\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|/3 + |\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}||\mathbb{G}_{\mathcal{P}_n}|dt\epsilon(n)^4/64 + \epsilon(n)^2|\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|/4 \leq |\Omega^{\mathcal{C}_n}_{\mathcal{P}_n}|(\epsilon(n) + |\mathbb{G}_{\mathcal{P}_n}|\epsilon(n)^4 dt/64)$$

with probability at least $1 - \nu(n)$ (as given in Subclaim 2) over $\mathcal{B}$'s queries, as desired. □

Claims 4 and 5 taken together suffice to prove Theorem 4.

6 Experimental Results

Finally, we investigated the accuracy of our methods for producing derived classifiers using data collected from the COMPAS recidivism prediction tool [1]. This data was compiled by ProPublica and used in their analysis of the fairness of the COMPAS algorithm [2]; it contains information about criminals in Broward County, Florida who were screened using this tool in 2013 and 2014. Specifically, we use the same subset of this data as the analysis in [2] (and subsequent analysis in [13]), taking the records of 5,278 individuals filtered according to the following criteria:

- Individuals whose race is either Caucasian or African-American (i.e., the two groups we wish to compare with respect to fairness).
- Individuals who have clear information as to the risk score assigned and whether they recidivated or not in the following two years.

	Black		White	
Score	R	NR	R	NR
1	93	272	139	466
2	121	225	110	211
3	140	158	88	150
4	171	166	107	136
5	171	152	94	106
6	200	118	96	64
7	222	121	73	40
8	225	76	75	21
9	237	80	56	21
10	193	34	36	14
Total	1773	1402	874	1229

	Black		White	
Score	R	NR	R	NR
1	5.25%	19.40%	15.90%	37.92%
2	6.82%	16.05%	12.59%	17.17%
3	7.90%	11.27%	10.07%	12.21%
4	9.64%	11.84%	12.24%	11.07%
5	9.64%	10.84%	10.76%	8.62%
6	11.28%	8.42%	10.98%	5.21%
7	12.52%	8.63%	8.35%	3.25%
8	12.69%	5.42%	8.58%	1.71%
9	13.37%	5.71%	6.41%	1.71%
10	10.89%	2.43%	4.12%	1.14%
Total	100%	100%	100%	100%

Figure 1: Conditional treatment rates for the COMPAS data. These tables represent the distributions of the scores (1-10) of individuals in each race conditioned on their actual class (recidivated or not), in (left) absolute numbers and (right) probabilities. In this case, the most treatment disparity occurs among non-recidivators given a score of 9, and so the fair treatment error is equal to $\ln\left(\frac{80}{1402}/\frac{21}{1229}\right) \approx 1.2058$.

- Individuals for whom their screening and arrest were separated by at most 30 days (otherwise, this indicates that they may be unrelated).
- Individuals screened for offenses resulting in jail time (e.g., not traffic violations).

For the COMPAS assessment, an individual's outcome is denoted by the risk score (1-10) assigned to them by the algorithm, and their actual class is denoted by whether or not they recidivated within two years of being assessed. Notably, the fact that the classifier is non-binary (even if the space of actual classes is binary) lends itself well to the techniques we present here.

6.1 Existing Biases

First, we examined the fairness errors present in the COMPAS analysis data according to our definitions. Our findings are consistent with those of prior investigations of COMPAS [2, 3, 11, 12], which suggest that the risk scores are far more consistent with an alternate (yet purportedly incorrect [6, 7]) notion of fairness known as *predictive parity*—which requires that the distributions of *classes* be close between groups conditioned on an *outcome*—than with fair treatment, which we recall requires the distributions of outcomes between groups to be close conditioned on any particular class. (Note that several results [3, 8, 9] exist indicating that predictive parity and fair treatment are incompatible, and so it is unsurprising that the COMPAS data displays a large error in fair treatment.) We present here the conditional treatment (Figure 1) and predictivity (Figure 2) rates for the COMPAS data set, along with our derivations of the errors for the respective fairness conditions.

We first note that, indeed, there is considerable disparity in the base rates of recidivism between the two groups; in particular, among those classified, the base rate for Caucasians is $874/2103 \approx$

	Score	1	2	3	4	5	6	7	8	9	10
Black	R	25.48%	34.97%	46.98%	50.74%	52.94%	62.89%	64.72%	74.75%	74.76%	85.02%
	NR	74.52%	65.03%	53.02%	49.26%	47.06%	37.11%	35.28%	25.25%	25.24%	14.98%
	Total	365	346	298	337	323	318	343	301	317	227
White	R	22.98%	34.27%	36.97%	44.03%	47.00%	60.00%	64.60%	78.13%	72.73%	72.00%
	NR	77.02%	65.73%	63.03%	55.97%	53.00%	40.00%	35.40%	21.88%	27.27%	28.00%
	Total	605	321	238	243	200	160	113	96	77	50

Figure 2: Conditional predictivity rates for the COMPAS data. These tables represent the distributions of the actual classes of individuals conditioned on their risk scores; the numbers of individuals are identical to those in Figure 1. In this case, the most predictive disparity occurs among non-recidivators given a score of 10, and so the predictive parity error is equal to $\ln\left(\frac{14}{50}/\frac{34}{227}\right) \approx 0.6256$. We note, however, that the second-worst predictive disparity occurs for recidivators with score 3, and has a log-ratio of 0.2395.

41.56%, while the base rate for African-Americans is $1773/3175 \approx 55.84\%$. In particular, this means, by impossibility results such as that of [9], that predictive parity and fair treatment cannot be simultaneously achieved, even in approximation.[13]

Observing Figure 1, we can see that, as observed by Chouldechova [3] and the ProPublica analysis [2], there is a considerable amount of disparate treatment between the two races; indeed, the distributions of individuals' scores conditioned on recidivism are heavily skewed towards lower scores for Caucasians than for African-Americans, and the distributions conditioned on non-recidivism are skewed likewise. The max-divergence between the distributions, or our error for fair treatment, is indeed quite large (approximately 1.2058), as might be expected given the observation that the COMPAS algorithm is designed to preserve predictive parity.

However, in that light, the results shown for predictive parity in Figure 2 may at first glance seem surprising—the error we derive is still relatively large (a max-divergence of roughly 0.6256). This, however, is mostly an unfortunate consequence of the small size of this data set; as observed in Section 5.2, the accuracy of experimental probabilities is highly dependent on the probability of certain classifications occurring for a particular group. In this case, a score of 10 is very rarely assigned to Caucasians, and so the disparity we see here (between non-recidivators assigned a score of 10) is likely due to general inaccuracy in the experimental probabilities of the data set resulting from the rarity of this event. This is substantiated by the fact that most conditional predictivity rates are far closer between groups—indeed, we remark that the second worst log-ratio is only 0.2395. Hence, while the small data set here exhibits relatively poor predictive parity, it seems likely that COMPAS, if run on a considerably larger set of individuals, will in general exhibit reasonably strong predictive parity. (Of course, this alone does not imply that the algorithm is in any way intuitively fair!)

[13] The best known (and likely a tight) theoretical lower bound for the error of either fair treatment or predictive parity in this particular case—specifically, where every class/outcome combination occurs with non-zero probability—is half the max-divergence between base rates, or roughly 0.1478. This bound is quite easily proven and was discovered in a preliminary version of [9].

ϵ	∞	1.2058	1.0	0.5	0.4	0.3	0.2	0.1
Accuracy	65.915%	65.915%	65.762%	65.420%	65.289%	65.182%	65.064%	64.900%
Similarity	100.00%	100.00%	99.598%	96.648%	95.395%	94.030%	92.290%	89.949%
Closeness	0	0	-0.0153	-0.1820	-0.2521	-0.3323	-0.4160	-0.4992

ϵ	0.05	0.02	0.01	0
Accuracy	64.805%	64.744%	64.573%	64.357%
Similarity	88.500%	87.607%	87.282%	86.951%
Closeness	-0.5427	-0.5689	-0.5782	-0.5881

Figure 3: Optimal utilities measured for the three distinct utility functions (described in Section 6.2) when we vary the error for fair treatment allowed by the constraints of our linear-program formulation.

6.2 Constructing Fair Derived Classifiers

Next, we evaluated the optimum of the linear program we present in Section 5.1 using the experimental probabilities from the COMPAS data set as 'training data". In particular, we combined the constraints for approximate fair treatment from Claim 2 with three different utility functions— one representing overall accuracy and the other two representing closeness to the original classifier—and measured the optimal utility among fair derived classifiers as we vary the allowed fair treatment error ϵ. We used an ordinary linear program solver to evaluate the optima, and efficiency-wise we found that computation took a negligible amount of time on a standard laptop computer. The results are shown in Figure 3.[14]

In particular, the three utility functions that we optimize for are as follows:

- First, we examine the overall *accuracy* of the classifier. We compute this in the same manner as in Section 5.3 (with the exception that we consider the equivalent problem of maximizing utility $1 - \ell$ rather than minimizing ℓ):

  $$\sum_{i,j \in \mathcal{O}; g} \Pr[g] \Pr_g[f(\sigma) = j \wedge \mathcal{C}(\sigma) = i] P^g_{i,j}$$

  This varies between 0 and 1 (although we note that it is trivial to achieve 50% accuracy by simply assigning a random classification). Of course, measuring this is complicated by the fact that we have ten outcomes and two classes; hence, we instead assign the outcomes representing "low" risk scores (1-4) to the non-recidivatory class and those representing "high" risk scores (7-10) to the recidivatory class. (For 5 and 6, we weighted membership in each class equally, at 0.5.)

- We also measured the *similarity* of the derived classifier to the original classification—that is, the fraction of individuals expected to be assigned the same score in the derived classifier as in the original. We compute this as:

  $$\Pr\left[\mathcal{C}'(\sigma) = \mathcal{C}(\sigma)\right] = \sum_{g; i \in \mathcal{O}} \Pr[g] \Pr_g[\mathcal{C}(\sigma) = i] P^g_{i,i}$$

[14] Notably, because the probabilities of events have a constant lower bound, we follow the remark in Section 5.2 and do not add random noise to our derived classifiers.

This will also lie between 0 and 1, and corresponds exactly to the expected fraction of times an individual will receive the same score as their COMPAS score.

- Finally, we measure the *closeness*, which measures the negative of the expected difference between an individual's score between the derived and original classifiers.[15] In particular, we compute this as:

$$-\sum_{g;i,j\in\mathcal{O}} \Pr[g]\Pr_g[\mathcal{C}(\sigma)=i]\,P^g_{i,j}|i-j|$$

  Notably, this is effectively a loss function; it is bounded above by 0, and a larger negative score means that individuals' scores differ more from the COMPAS scores in expectation. It is bounded in $[-10,0]$ rather than $[0,1]$, but to prove near-optimality we can simply scale the result from Claim 5 accordingly (albeit losing the respective constant factor).

From Figure 3, we observe that it is possible to create a perfectly fair derived classifier with relatively small accuracy loss compared to the original COMPAS data. However, using the notion of accuracy as our utility function had an interesting yet predictable effect on the derived classifiers produced by the optimum to the linear program; we found that these classifiers acted as simple *threshold classifiers* over the prior risk score. For instance, observing the accuracy-based optimal derived classifier with $\epsilon = 0.2$, there were clear thresholds in this derived classifier such that everyone in certain prior score ranges (1-5 for African-Americans, 1-3 for Caucasians) would receive a low risk score, everyone in certain other ranges (6-10 and 5-10, respectively) would receive a high risk score, and one "mixed" threshold (where, of Caucasians with prior score 4, 69% received a low score and 31% a high score). This is of course reflective of the fact that there are only two classes; perhaps expectedly, for this measure of utility, our results produce classifiers very similar to the binary post-processing of Hardt et al. [6], which indeed has been found to exhibit a remarkably similar ($\approx 64.5\%$) accuracy with respect to the same data [13].

Hence, we consider other measures of utility not based on a binary class space. Indeed, we note that when considering intuitive fairness it is a desideratum that individuals' scores in the derived classifier should be very close to their scores from the original assessment, and equal as often as possible. (For instance, giving 10% of individuals originally scored as 1 a score of 10 in the derived classifier would be rather questionable.) This is the basis for our second and third notions of the utility of a classifier; how close can we make the derived classifier to the prior classifier while requiring fair treatment?

As Figure 3 shows, we can derive a classifier that will give individuals the same score as COMPAS roughly 87% of the time while still satisfying errorless fair treatment with respect to our data; it is also possible to derive a classifier satisfying errorless fair treatment that gives individuals a score which is in expectation only about 0.588 points different from their COMPAS score; as this latter classifier exhibits some interesting properties, we reproduce it in Figure 4. In particular, we notice that this classifier almost always gives African-Americans the same score as their COMPAS score, while tending to give Caucasians a score 1-3 points higher to account for the treatment bias in the original data.

[15] This represents the intuition that, in the prior scenario (where we wish to come up with a derived classifier as close to the original as possible), individuals' scores in the derived classifier should be relatively close to their scores in the prior classifier, even if they do not match. In the prior, the closeness is not considered, only whether the two scores match.

Score		1	2	3	4	5	6	7	8	9	10
Black	1	100.0%	0.0%	0.0%	0.0%	0.0%	2.17%	0.0%	26.15%	0.0%	0.0%
	2	0.0%	100.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	3	0.0%	0.0%	100.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	4	0.0%	0.0%	0.0%	100.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	5	0.0%	0.0%	0.0%	0.0%	100.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	6	0.0%	0.0%	0.0%	0.0%	0.0%	97.83%	0.0%	0.0%	0.0%	0.0%
	7	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	100.0%	0.0%	0.0%	0.0%
	8	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	73.85%	0.0%	0.0%
	9	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	97.98%	0.0%
	10	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2.01%	100.0%
White	1	55.39%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	2	41.54%	1.73%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	3	3.07%	58.86%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	4	0.0%	7.80%	86.04%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	5	0.0%	31.61%	13.96%	23.55%	12.81%	0.0%	0.0%	0.0%	0.0%	0.0%
	6	0.0%	0.0%	0.0%	57.05%	0.0%	36.88%	0.0%	0.0%	0.0%	0.0%
	7	0.0%	0.0%	0.0%	19.40%	57.07%	0.0%	48.00%	0.0%	0.0%	0.0%
	8	0.0%	0.0%	0.0%	0.0%	9.98%	21.94%	52.00%	18.01%	0.0%	0.0%
	9	0.0%	0.0%	0.0%	0.0%	20.15%	41.18%	0.0%	0.0%	100.0%	0.0%
	10	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	81.99%	0.0%	100.0%

Figure 4: The derived classifier which maximizes the closeness utility function with zero fair treatment error. Columns represent the original class; rows represent the derived class. While there are a few anomalies (e.g., 26% of blacks with COMPAS score 8 receiving a score of 1), these are likely due to our use of experimental probabilities over a small data set; for the most part, this classifier exhibits behavior relatively consistent with rectifying the biases observed in the original data.

All of the derived classifiers exhibited here, as well as our utility functions, are of course derived using the experimental probabilities of the data set, which are only rough estimates of the actual probabilities (indeed, we suspect that this leads to some variance-related quirks, as observed with the predictive parity rates and the classifier in Figure 4). The actual probabilities over individuals classified by COMPAS are unknown, but we note that, as observed in Section 5, derived classifiers which satisfy fair treatment according to these experimental probabilities are very likely to satisfy approximate fair treatment with respect to the actual probabilities. In addition, noting that bounds analogous to those presented in Claim 5 apply to all three of the utility functions presented here, the utility of our derived classifiers will likely not vary by much between our experimentally derived versions of the utility functions and the utility functions over the real probabilities. Hence, we stress that the experimental results here are accurate not only with respect to our data set, but also with respect to COMPAS classification in general over the distribution of individuals from which this data was collected.

As a final note, we once again remark on the relatively small size of the experimental data set we employed here. We plan to repeat this analysis on larger suitable data sets in the future when we are able to find them, and we expect to obtain similar but higher-quality results compared to what we observed here when we do so.

References

[1] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. How We Analyzed the COMPAS Recidivism Algorithm. *ProPublica*, 2016. https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm.

[2] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine Bias: Risk Assessments in Criminal Sentencing. *ProPublica*, 2016. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing.

[3] Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *FATML*, 2016.

[4] Cynthia Dwork. Differential Privacy. In *Proceedings of the 33rd International Conference on Automata, Languages and Programming - Volume Part II*, ICALP'06, pages 1–12, Berlin, Heidelberg, 2006. Springer-Verlag.

[5] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness Through Awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS '12, pages 214–226, New York, NY, USA, 2012. ACM.

[6] Moritz Hardt, Eric Price, and Nathan Srebro. Equality of Opportunity in Supervised Learning. *NIPS*, 2016.

[7] Niki Kilbertus, Mateo Rojas-Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing, and Bernhard Schölkopf. Avoiding Discrimination through Causal Reasoning. *NIPS*, 2017.

[8] Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent Trade-Offs in the Fair Determination of Risk Scores. *ITCS*, 2017.

[9] Andrew Morgan and Rafael Pass. Paradoxes in Fair Computer-Aided Decision Making. https://arxiv.org/abs/1711.11066, 2017.

[10] Judea Pearl. Direct and Indirect Effects. In *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, UAI'01, pages 411–420, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[11] Jennifer L. Skeem and Christopher T. Lowenkamp. Risk, Race, and Recidivism: Predictive Bias and Disparate Impact. 2016. http://dx.doi.org/10.2139/ssrn.2687339.

[12] Jennifer L. Skeem, John Monahan, and Christopher T. Lowenkamp. Gender, Risk Assessment, and Sanctioning: The Cost of Treating Women Like Men. 2016. http://dx.doi.org/10.2139/ssrn.2718460.

[13] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P. Gummadi. Fairness Beyond Disparate Treatment and Disparate Impact: Learning Classification without Disparate Mistreatment. https://arxiv.org/abs/1610.08452, 2016.