A Reductions Approach to Fair Classification

Alekh Agarwal [1] **Alina Beygelzimer** [2] **Miroslav Dudík** [1] **John Langford** [1] **Hanna Wallach** [1]

Abstract

We present a systematic approach for achieving fairness in a binary classification setting. While we focus on two well-known quantitative definitions of fairness, our approach encompasses many other previously studied definitions as special cases. The key idea is to reduce fair classification to a sequence of cost-sensitive classification problems, whose solutions yield a randomized classifier with the lowest (empirical) error subject to the desired constraints. We introduce two reductions that work for any representation of the cost-sensitive classifier and compare favorably to prior baselines on a variety of data sets, while overcoming several of their disadvantages.

1. Introduction

Over the past few years, the media have paid considerable attention to machine learning systems and their ability to inadvertently discriminate against minorities, historically disadvantaged populations, and other protected groups when allocating resources (e.g., loans) or opportunities (e.g., jobs). In response to this scrutiny—and driven by ongoing debates and collaborations with lawyers, policy-makers, social scientists, and others (e.g., Barocas & Selbst, 2016)—machine learning researchers have begun to turn their attention to the topic of "fairness in machine learning," and, in particular, to the design of fair classification and regression algorithms.

In this paper we study the task of binary classification subject to fairness constraints with respect to a pre-defined protected attribute, such as race or sex. Previous work in this area can be divided into two broad groups of approaches.

The first group of approaches incorporate specific quantitative definitions of fairness into existing machine learning methods, often by relaxing the desired definitions of fairness, and only enforcing weaker constraints, such as lack of correlation (e.g., Woodworth et al., 2017; Zafar et al., 2017; Johnson et al., 2016; Kamishima et al., 2011; Donini et al., 2018). The resulting fairness guarantees typically only hold under strong distributional assumptions, and the approaches are tied to specific families of classifiers, such as SVMs.

The second group of approaches eliminate the restriction to specific classifier families and treat the underlying classification method as a "black box," while implementing a wrapper that either works by pre-processing the data or post-processing the classifier's predictions (e.g., Kamiran & Calders, 2012; Feldman et al., 2015; Hardt et al., 2016; Calmon et al., 2017). Existing pre-processing approaches are specific to particular definitions of fairness and typically seek to come up with a single transformed data set that will work across all learning algorithms, which, in practice, leads to classifiers that still exhibit substantial unfairness (see our evaluation in Section 4). In contrast, post-processing allows a wider range of fairness definitions and results in provable fairness guarantees. However, it is not guaranteed to find the most accurate fair classifier, and requires test-time access to the protected attribute, which might not be available.

We present a general-purpose approach that has the key advantage of this second group of approaches—i.e., the underlying classification method is treated as a black box—but without the noted disadvantages. Our approach encompasses a wide range of fairness definitions, is guaranteed to yield the most accurate fair classifier, and does not require test-time access to the protected attribute. Specifically, our approach allows any definition of fairness that can be formalized via linear inequalities on conditional moments, such as *demographic parity* or *equalized odds* (see Section 2.1). We show how binary classification subject to these constraints can be reduced to a sequence of cost-sensitive classification problems. We require only black-box access to a cost-sensitive classification algorithm, which does not need to have any knowledge of the desired definition of fairness or protected attribute. We show that the solutions to our sequence of cost-sensitive classification problems yield a randomized classifier with the lowest (empirical) error subject to the desired fairness constraints.

Corbett-Davies et al. (2017) and Menon & Williamson

[1]Microsoft Research, New York [2]Yahoo! Research, New York. Correspondence to: A. Agarwal <alekha@microsoft.com>, A. Beygelzimer <beygel@gmail.com>, M. Dudík <mdudik@microsoft.com>, J. Langford <jcl@microsoft.com>, H. Wallach <wallach@microsoft.com>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

(2018) begin with a similar goal to ours, but they analyze the Bayes optimal classifier under fairness constraints in the limit of infinite data. In contrast, our focus is algorithmic, our approach applies to any classifier family, and we obtain finite-sample guarantees. Dwork et al. (2018) also begin with a similar goal to ours. Their approach partitions the training examples into subsets according to protected attribute values and then leverages transfer learning to jointly learn from these separate data sets. Our approach avoids partitioning the data and assumes access only to a classification algorithm rather than a transfer learning algorithm.

A preliminary version of this paper appeared at the FAT/ML workshop (Agarwal et al., 2017), and led to extensions with more general optimization objectives (Alabi et al., 2018) and combinatorial protected attributes (Kearns et al., 2018).

In the next section, we formalize our problem. While we focus on two well-known quantitative definitions of fairness, our approach also encompasses many other previously studied definitions of fairness as special cases. In Section 3, we describe our reductions approach to fair classification and its guarantees in detail. The experimental study in Section 4 shows that our reductions compare favorably to three baselines, while overcoming some of their disadvantages and also offering the flexibility of picking a suitable accuracy–fairness tradeoff. Our results demonstrate the utility of having a general-purpose approach for combining machine learning methods and quantitative fairness definitions.

2. Problem Formulation

We consider a binary classification setting where the training examples consist of triples (X, A, Y), where $X \in \mathcal{X}$ is a feature vector, $A \in \mathcal{A}$ is a protected attribute, and $Y \in \{0, 1\}$ is a label. The feature vector X can either contain the protected attribute A as one of the features or contain other features that are arbitrarily indicative of A. For example, if the classification task is to predict whether or not someone will default on a loan, each training example might correspond to a person, where X represents their demographics, income level, past payment history, and loan amount; A represents their race; and Y represents whether or not they defaulted on that loan. Note that X might contain their race as one of the features or, for example, contain their zipcode—a feature that is often correlated with race. Our goal is to learn an accurate classifier $h : \mathcal{X} \to \{0, 1\}$ from some set (i.e., family) of classifiers $\mathcal{H}$, such as linear threshold rules, decision trees, or neural nets, while satisfying some definition of fairness. Note that the classifiers in $\mathcal{H}$ do not explicitly depend on A.

2.1. Fairness Definitions

We focus on two well-known quantitative definitions of fairness that have been considered in previous work on fair classification; however, our approach also encompasses many other previously studied definitions of fairness as special cases, as we explain at the end of this section.

The first definition—*demographic* (or statistical) *parity*—can be thought of as a stronger version of the US Equal Employment Opportunity Commission's "four-fifths rule," which requires that the "selection rate for any race, sex, or ethnic group [must be at least] four-fifths (4/5) (or eighty percent) of the rate for the group with the highest rate."[1]

Definition 1 (Demographic parity—DP)**.** *A classifier h satisfies demographic parity under a distribution over (X, A, Y) if its prediction $h(X)$ is statistically independent of the protected attribute A—that is, if $\mathbb{P}[h(X) = \hat{y} \mid A = a] = \mathbb{P}[h(X) = \hat{y}]$ for all a, $\hat{y}$. Because $\hat{y} \in \{0, 1\}$, this is equivalent to $\mathbb{E}[h(X) \mid A = a] = \mathbb{E}[h(X)]$ for all a.*

The second definition—*equalized odds*—was recently proposed by Hardt et al. (2016) to remedy two previously noted flaws with demographic parity (Dwork et al., 2012). First, demographic parity permits a classifier which accurately classifies data points with one value $A = a$, such as the value a with the most data, but makes random predictions for data points with $A \neq a$ as long as the probabilities of $h(X) = 1$ match. Second, demographic parity rules out perfect classifiers whenever Y is correlated with A. In contrast, equalized odds suffers from neither of these flaws.

Definition 2 (Equalized odds—EO)**.** *A classifier h satisfies equalized odds under a distribution over (X, A, Y) if its prediction $h(X)$ is conditionally independent of the protected attribute A given the label Y—that is, if $\mathbb{P}[h(X) = \hat{y} \mid A = a, Y = y] = \mathbb{P}[h(X) = \hat{y} \mid Y = y]$ for all a, y, and $\hat{y}$. Because $\hat{y} \in \{0, 1\}$, this is equivalent to $\mathbb{E}[h(X) \mid A = a, Y = y] = \mathbb{E}[h(X) \mid Y = y]$ for all a, y.*

We now show how each definition can be viewed as a special case of a general set of linear constraints of the form

$$\mathbf{M}\boldsymbol{\mu}(h) \leq \mathbf{c}, \tag{1}$$

where matrix $\mathbf{M} \in \mathbb{R}^{|\mathcal{K}| \times |\mathcal{J}|}$ and vector $\mathbf{c} \in \mathbb{R}^{|\mathcal{K}|}$ describe the linear constraints, each indexed by $k \in \mathcal{K}$, and $\boldsymbol{\mu}(h) \in \mathbb{R}^{|\mathcal{J}|}$ is a vector of conditional moments of the form

$$\mu_j(h) = \mathbb{E}\big[\, g_j(X, A, Y, h(X)) \;\big|\; \mathcal{E}_j \,\big] \quad \text{for } j \in \mathcal{J},$$

where $g_j : \mathcal{X} \times \mathcal{A} \times \{0, 1\} \times \{0, 1\} \to [0, 1]$ and $\mathcal{E}_j$ is an event defined with respect to (X, A, Y). Crucially, g_j depends on h, while $\mathcal{E}_j$ cannot depend on h in any way.

Example 1 (DP)**.** *In a binary classification setting, demographic parity can be expressed as a set of $|\mathcal{A}|$ equality constraints, each of the form $\mathbb{E}[h(X) \mid A = a] = \mathbb{E}[h(X)]$. Letting $\mathcal{J} = \mathcal{A} \cup \{\star\}$, $g_j(X, A, Y, h(X)) = h(X)$ for all j,*

[1] See the Uniform Guidelines on Employment Selection Procedures, 29 C.F.R. §1607.4(D) (2015).

$\mathcal{E}_a = \{A = a\}$, and $\mathcal{E}_\star = \{True\}$, where $\{True\}$ refers to the event encompassing all points in the sample space, each equality constraint can be expressed as $\mu_a(h) = \mu_\star(h)$.[2] *Finally, because each such constraint can be equivalently expressed as a pair of inequality constraints of the form*

$$\mu_a(h) - \mu_\star(h) \leq 0$$
$$-\mu_a(h) + \mu_\star(h) \leq 0,$$

demographic parity can be expressed as equation (1), where $\mathcal{K} = \mathcal{A} \times \{+, -\}$, $M_{(a,+),a'} = \mathbf{1}\{a' = a\}$, $M_{(a,+),\star} = -1$, $M_{(a,-),a'} = -\mathbf{1}\{a' = a\}$, $M_{(a,-),\star} = 1$, and $\mathbf{c} = \mathbf{0}$. Expressing each equality constraint as a pair of inequality constraints allows us to control the extent to which each constraint is enforced by positing $c_k > 0$ for some (or all) k.

Example 2 (EO)**.** *In a binary classification setting, equalized odds can be expressed as a set of $2|\mathcal{A}|$ equality constraints, each of the form $\mathbb{E}[h(X) \mid A = a, Y = y] = \mathbb{E}[h(X) \mid Y = y]$. Letting $\mathcal{J} = (\mathcal{A} \cup \{\star\}) \times \{0, 1\}$, $g_j(X, A, Y, h(X)) = h(X)$ for all j, $\mathcal{E}_{(a,y)} = \{A = a, Y = y\}$, and $\mathcal{E}_{(\star,y)} = \{Y = y\}$, each equality constraint can be equivalently expressed as*

$$\mu_{(a,y)}(h) - \mu_{(\star,y)}(h) \leq 0$$
$$-\mu_{(a,y)}(h) + \mu_{(\star,y)}(h) \leq 0.$$

As a result, equalized odds can be expressed as equation (1), where $\mathcal{K} = \mathcal{A} \times \mathcal{Y} \times \{+, -\}$, $M_{(a,y,+),(a',y')} = \mathbf{1}\{a' = a,\ y' = y\}$, $M_{(a,y,+),(\star,y')} = -1$, $M_{(a,y,-),(a',y')} = -\mathbf{1}\{a' = a,\ y' = y\}$, $M_{(a,y,-),(\star,y')} = 1$, and $\mathbf{c} = \mathbf{0}$. Again, we can posit $c_k > 0$ for some (or all) k to allow small violations of some (or all) of the constraints.

Although we omit the details, we note that many other previously studied definitions of fairness can also be expressed as equation (1). For example, *equality of opportunity* (Hardt et al., 2016) (also known as *balance for the positive class*; Kleinberg et al., 2017), *balance for the negative class* (Kleinberg et al., 2017), *error-rate balance* (Chouldechova, 2017), *overall accuracy equality* (Berk et al., 2017), and *treatment equality* (Berk et al., 2017) can all be expressed as equation (1); in contrast, *calibration* (Kleinberg et al., 2017) and *predictive parity* (Chouldechova, 2017) cannot because to do so would require the event $\mathcal{E}_j$ to depend on h. We note that our approach can also be used to satisfy multiple definitions of fairness, though if these definitions are mutually contradictory, e.g., as described by Kleinberg et al. (2017), then our guarantees become vacuous.

2.2. Fair Classification

In a standard (binary) classification setting, the goal is to learn the classifier $h \in \mathcal{H}$ with the minimum classification error: $\mathrm{err}(h) := \mathbb{P}[h(X) \neq Y]$. However, because our goal is to learn the most accurate classifier while satisfying fairness constraints, as formalized above, we instead seek to find the solution to the constrained optimization problem[3]

$$\min_{h \in \mathcal{H}} \mathrm{err}(h) \quad \text{subject to} \quad \mathbf{M}\boldsymbol{\mu}(h) \leq \mathbf{c}. \tag{2}$$

Furthermore, rather than just considering classifiers in the set $\mathcal{H}$, we can enlarge the space of possible classifiers by considering *randomized classifiers* that can be obtained via a distribution over $\mathcal{H}$. By considering randomized classifiers, we can achieve better accuracy–fairness tradeoffs than would otherwise be possible. A randomized classifier Q makes a prediction by first sampling a classifier $h \in \mathcal{H}$ from Q and then using h to make the prediction. The resulting classification error is $\mathrm{err}(Q) = \sum_{h \in \mathcal{H}} Q(h)\,\mathrm{err}(h)$ and the conditional moments are $\boldsymbol{\mu}(Q) = \sum_{h \in \mathcal{H}} Q(h)\boldsymbol{\mu}(h)$ (see Appendix A for the derivation). Thus we seek to solve

$$\min_{Q \in \Delta} \mathrm{err}(Q) \quad \text{subject to} \quad \mathbf{M}\boldsymbol{\mu}(Q) \leq \mathbf{c}, \tag{3}$$

where Δ is the set of all distributions over $\mathcal{H}$.

In practice, we do not know the true distribution over (X, A, Y) and only have access to a data set of training examples $\{(X_i, A_i, Y_i)\}_{i=1}^n$. We therefore replace $\mathrm{err}(Q)$ and $\boldsymbol{\mu}(Q)$ in equation (3) with their empirical versions $\widehat{\mathrm{err}}(Q)$ and $\widehat{\boldsymbol{\mu}}(Q)$. Because of the sampling error in $\widehat{\boldsymbol{\mu}}(Q)$, we also allow errors in satisfying the constraints by setting $\widehat{c}_k = c_k + \varepsilon_k$ for all k, where $\varepsilon_k \geq 0$. After these modifications, we need to solve the empirical version of equation (3):

$$\min_{Q \in \Delta} \widehat{\mathrm{err}}(Q) \quad \text{subject to} \quad \mathbf{M}\widehat{\boldsymbol{\mu}}(Q) \leq \widehat{\mathbf{c}}. \tag{4}$$

3. Reductions Approach

We now show how the problem (4) can be reduced to a sequence of *cost-sensitive classification* problems. We further show that the solutions to our sequence of cost-sensitive classification problems yield a randomized classifier with the lowest (empirical) error subject to the desired constraints.

3.1. Cost-sensitive Classification

We assume access to a cost-sensitive classification algorithm for the set $\mathcal{H}$. The input to such an algorithm is a data set of training examples $\{(X_i, C_i^0, C_i^1)\}_{i=1}^n$, where C_i^0 and C_i^1 denote the losses—*costs* in this setting—for predicting the labels 0 or 1, respectively, for X_i. The algorithm outputs

$$\arg\min_{h \in \mathcal{H}} \sum_{i=1}^{n} h(X_i)\, C_i^1 + (1 - h(X_i))\, C_i^0. \tag{5}$$

[2] Note that $\mu_\star(h) = \mathbb{E}[h(X) \mid True] = \mathbb{E}[h(X)]$.

[3] We consider misclassification error for concreteness, but all the results in this paper apply to any error of the form $\mathrm{err}(h) = \mathbb{E}[g_{\mathrm{err}}(X, A, Y, h(X))]$, where $g_{\mathrm{err}}(\cdot, \cdot, \cdot, \cdot) \in [0, 1]$.

This abstraction allows us to specify different costs for different training examples, which is essential for incorporating fairness constraints. Moreover, efficient cost-sensitive classification algorithms are readily available for several common classifier representations (e.g., Beygelzimer et al., 2005; Langford & Beygelzimer, 2005; Fan et al., 1999). In particular, equation (5) is equivalent to a *weighted classification* problem, where the input consists of labeled examples $\{(X_i, Y_i, W_i)\}_{i=1}^n$ with $Y_i \in \{0,1\}$ and $W_i \geq 0$, and the goal is to minimize the weighted classification error $\sum_{i=1}^n W_i \, \mathbf{1}\{h(X_i) \neq Y_i\}$. This is equivalent to equation (5) if we set $W_i = |C_i^0 - C_i^1|$ and $Y_i = \mathbf{1}\{C_i^0 \geq C_i^1\}$.

3.2. Reduction

To derive our fair classification algorithm, we rewrite equation (4) as a saddle point problem. We begin by introducing a Lagrange multiplier $\lambda_k \geq 0$ for each of the $|\mathcal{K}|$ constraints, summarized as $\boldsymbol{\lambda} \in \mathbb{R}_+^{|\mathcal{K}|}$, and form the Lagrangian

$$L(Q, \boldsymbol{\lambda}) = \widehat{\mathrm{err}}(Q) + \boldsymbol{\lambda}^\top \left(\mathbf{M}\widehat{\boldsymbol{\mu}}(Q) - \widehat{\mathbf{c}}\right).$$

Thus, equation (4) is equivalent to

$$\min_{Q \in \Delta} \max_{\boldsymbol{\lambda} \in \mathbb{R}_+^{|\mathcal{K}|}} L(Q, \boldsymbol{\lambda}). \tag{6}$$

For computational and statistical reasons, we impose an additional constraint on the ℓ_1 norm of $\boldsymbol{\lambda}$ and seek to simultaneously find the solution to the constrained version of (6) as well as its dual, obtained by switching min and max:

$$\min_{Q \in \Delta} \max_{\boldsymbol{\lambda} \in \mathbb{R}_+^{|\mathcal{K}|},\, \|\boldsymbol{\lambda}\|_1 \leq B} L(Q, \boldsymbol{\lambda}), \tag{P}$$

$$\max_{\boldsymbol{\lambda} \in \mathbb{R}_+^{|\mathcal{K}|},\, \|\boldsymbol{\lambda}\|_1 \leq B} \min_{Q \in \Delta} L(Q, \boldsymbol{\lambda}). \tag{D}$$

Because L is linear in Q and $\boldsymbol{\lambda}$ and the domains of Q and $\boldsymbol{\lambda}$ are convex and compact, both problems have solutions (which we denote by $Q^\dagger$ and $\boldsymbol{\lambda}^\dagger$) and the minimum value of (P) and the maximum value of (D) are equal and coincide with $L(Q^\dagger, \boldsymbol{\lambda}^\dagger)$. Thus, $(Q^\dagger, \boldsymbol{\lambda}^\dagger)$ is the saddle point of L (Corollary 37.6.2 and Lemma 36.2 of Rockafellar, 1970).

We find the saddle point by using the standard scheme of Freund & Schapire (1996), developed for the equivalent problem of solving for an equilibrium in a zero-sum game. From game-theoretic perspective, the saddle point can be viewed as an equilibrium of a game between two players: the Q-player choosing Q and the $\boldsymbol{\lambda}$-player choosing $\boldsymbol{\lambda}$. The Lagrangian $L(Q, \boldsymbol{\lambda})$ specifies how much the Q-player has to pay to the $\boldsymbol{\lambda}$-player after they make their choices. At the saddle point, neither player wants to deviate from their choice.

Our algorithm finds an approximate equilibrium in which neither player can gain more than ν by changing their choice (where $\nu > 0$ is an input to the algorithm). Such an approximate equilibrium corresponds to a *ν-approximate saddle point* of the Lagrangian, which is a pair $(\widehat{Q}, \widehat{\boldsymbol{\lambda}})$, where

$$L(\widehat{Q}, \widehat{\boldsymbol{\lambda}}) \leq L(Q, \widehat{\boldsymbol{\lambda}}) + \nu \quad \text{for all } Q \in \Delta,$$
$$L(\widehat{Q}, \widehat{\boldsymbol{\lambda}}) \geq L(\widehat{Q}, \boldsymbol{\lambda}) - \nu \quad \text{for all } \boldsymbol{\lambda} \in \mathbb{R}_+^{|\mathcal{K}|},\ \|\boldsymbol{\lambda}\|_1 \leq B.$$

We proceed iteratively by running a no-regret algorithm for the $\boldsymbol{\lambda}$-player, while executing the best response of the Q-player. Following Freund & Schapire (1996), the average play of both players converges to the saddle point. We run the exponentiated gradient algorithm (Kivinen & Warmuth, 1997) for the $\boldsymbol{\lambda}$-player and terminate as soon as the suboptimality of the average play falls below the pre-specified accuracy ν. The best response of the Q-player can always be chosen to put all of the mass on one of the candidate classifiers $h \in \mathcal{H}$, and can be implemented by a single call to a cost-sensitive classification algorithm for the set $\mathcal{H}$.

Algorithm 1 fully implements this scheme, except for the functions $\textsc{Best}_{\boldsymbol{\lambda}}$ and $\textsc{Best}_h$, which correspond to the best-response algorithms of the two players. (We need the best response of the $\boldsymbol{\lambda}$-player to evaluate whether the suboptimality of the current average play has fallen below ν.) The two best response functions can be calculated as follows.

Algorithm 1 Exp. gradient reduction for fair classification

Input: training examples $\{(X_i, Y_i, A_i)\}_{i=1}^n$
fairness constraints specified by g_j, $\mathcal{E}_j$, $\mathbf{M}$, $\widehat{\mathbf{c}}$
bound B, accuracy ν, learning rate η
Set $\boldsymbol{\theta}_1 = \mathbf{0} \in \mathbb{R}^{|\mathcal{K}|}$
for $t = 1, 2, \ldots$ **do**
  Set $\lambda_{t,k} = B \, \frac{\exp\{\theta_k\}}{1 + \sum_{k' \in \mathcal{K}} \exp\{\theta_{k'}\}}$ for all $k \in \mathcal{K}$
  $h_t \leftarrow \textsc{Best}_h(\boldsymbol{\lambda}_t)$
  $\widehat{Q}_t \leftarrow \frac{1}{t}\sum_{t'=1}^t h_{t'}, \quad \overline{L} \leftarrow L\left(\widehat{Q}_t, \textsc{Best}_{\boldsymbol{\lambda}}(\widehat{Q}_t)\right)$
  $\widehat{\boldsymbol{\lambda}}_t \leftarrow \frac{1}{t}\sum_{t'=1}^t \boldsymbol{\lambda}_{t'}, \quad \underline{L} \leftarrow L\left(\textsc{Best}_h(\widehat{\boldsymbol{\lambda}}_t), \widehat{\boldsymbol{\lambda}}_t\right)$
  $\nu_t \leftarrow \max\left\{L(\widehat{Q}_t, \widehat{\boldsymbol{\lambda}}_t) - \underline{L}, \quad \overline{L} - L(\widehat{Q}_t, \widehat{\boldsymbol{\lambda}}_t)\right\}$
  if $\nu_t \leq \nu$ **then**
    Return $(\widehat{Q}_t, \widehat{\boldsymbol{\lambda}}_t)$
  end if
  Set $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \eta\left(\mathbf{M}\widehat{\boldsymbol{\mu}}(h_t) - \widehat{\mathbf{c}}\right)$
end for

$\textsc{Best}_{\boldsymbol{\lambda}}(Q)$**: the best response of the $\boldsymbol{\lambda}$-player.** The best response of the $\boldsymbol{\lambda}$-player for a given Q is any maximizer of $L(Q, \boldsymbol{\lambda})$ over all valid $\boldsymbol{\lambda}$s. In our setting, it can always be chosen to be either $\mathbf{0}$ or put all of the mass on the most violated constraint. Letting $\widehat{\boldsymbol{\gamma}}(Q) := \mathbf{M}\widehat{\boldsymbol{\mu}}(Q)$ and letting $\mathbf{e}_k$ denote the k^{th} vector of the standard basis, $\textsc{Best}_{\boldsymbol{\lambda}}(Q)$ returns

$$\begin{cases} \mathbf{0} & \text{if } \widehat{\boldsymbol{\gamma}}(Q) \leq \widehat{\mathbf{c}}, \\ B\mathbf{e}_{k^*} & \text{otherwise, where } k^* = \arg\max_k [\widehat{\gamma}_k(Q) - \widehat{c}_k]. \end{cases}$$

$\textsc{Best}_h(\boldsymbol{\lambda})$: the best response of the Q-player. Here, the best response minimizes $L(Q, \boldsymbol{\lambda})$ over all Qs in the simplex. Because L is linear in Q, the minimizer can always be chosen to put all of the mass on a single classifier h. We show how to obtain the classifier constituting the best response via a reduction to cost-sensitive classification. Letting $p_j := \widehat{\mathbb{P}}[\mathcal{E}_j]$ be the empirical event probabilities, the Lagrangian for Q which puts all of the mass on a single h is then

$$\begin{aligned}
&L(h, \boldsymbol{\lambda}) = \widehat{\mathrm{err}}(h) + \boldsymbol{\lambda}^\top \left(\mathbf{M}\widehat{\boldsymbol{\mu}}(h) - \widehat{\mathbf{c}}\right) \\
&= \widehat{\mathbb{E}}\left[\mathbf{1}\{h(X) \neq Y\}\right] - \boldsymbol{\lambda}^\top \widehat{\mathbf{c}} + \sum_{k,j} M_{k,j} \lambda_k \widehat{\mu}_j(h) \\
&= -\boldsymbol{\lambda}^\top \widehat{\mathbf{c}} + \widehat{\mathbb{E}}\left[\mathbf{1}\{h(X) \neq Y\}\right] \\
&\quad + \sum_{k,j} \frac{M_{k,j}\lambda_k}{p_j} \widehat{\mathbb{E}}\Big[g_j\big(X,A,Y,h(X)\big)\, \mathbf{1}\{(X,A,Y) \in \mathcal{E}_j\}\Big].
\end{aligned}$$

Assuming a data set of training examples $\{(X_i, A_i, Y_i)\}_{i=1}^n$, the minimization of $L(h, \boldsymbol{\lambda})$ over h then corresponds to cost-sensitive classification on $\{(X_i, C_i^0, C_i^1)\}_{i=1}^n$ with costs[4]

$$\begin{aligned}
C_i^0 &= \mathbf{1}\{Y_i \neq 0\} \\
&\quad + \sum_{k,j} \frac{M_{k,j}\lambda_k}{p_j} g_j(X_i,A_i,Y_i,0)\, \mathbf{1}\{(X_i,A_i,Y_i) \in \mathcal{E}_j\} \\
C_i^1 &= \mathbf{1}\{Y_i \neq 1\} \\
&\quad + \sum_{k,j} \frac{M_{k,j}\lambda_k}{p_j} g_j(X_i,A_i,Y_i,1)\, \mathbf{1}\{(X_i,A_i,Y_i) \in \mathcal{E}_j\}.
\end{aligned}$$

Theorem 1. *Letting $\rho := \max_h \|\mathbf{M}\widehat{\boldsymbol{\mu}}(h) - \widehat{\mathbf{c}}\|_\infty$, Algorithm 1 satisfies the inequality*

$$\nu_t \leq \frac{B \log(|\mathcal{K}| + 1)}{\eta t} + \eta \rho^2 B.$$

Thus, for $\eta = \frac{\nu}{2\rho^2 B}$, Algorithm 1 will return a ν-approximate saddle point of L in at most $\frac{4\rho^2 B^2 \log(|\mathcal{K}|+1)}{\nu^2}$ iterations.

This theorem, proved in Appendix B, bounds the suboptimality ν_t of the average play $(\widehat{Q}_t, \widehat{\boldsymbol{\lambda}}_t)$, which is equal to its suboptimality as a saddle point. The right-hand side of the bound is optimized by $\eta = \sqrt{\log(|\mathcal{K}| + 1)} \,/\, (\rho\sqrt{t})$, leading to the bound $\nu_t \leq 2\rho B \sqrt{\log(|\mathcal{K}| + 1) \,/\, t}$. This bound decreases with the number of iterations t and grows very slowly with the number of constraints $|\mathcal{K}|$. The quantity ρ is a problem-specific constant that bounds how much any single classifier $h \in \mathcal{H}$ can violate the desired set of fairness constraints. Finally, B is the bound on the ℓ_1-norm of $\boldsymbol{\lambda}$, which we introduced to enable this specific algorithmic scheme. In general, larger values of B will bring the problem (P) closer to (6), and thus also to (4), but at the cost of needing more iterations to reach any given suboptimality. In particular, as we derive in the theorem, achieving suboptimality ν may need up to $4\rho^2 B^2 \log(|\mathcal{K}| + 1) \,/\, \nu^2$ iterations.

[4] For general error, $\mathrm{err}(h) = \mathbb{E}[g_{\mathrm{err}}(X, A, Y, h(X))]$, the costs C_i^0 and C_i^1 contain, respectively, the terms $g_{\mathrm{err}}(X_i, A_i, Y_i, 0)$ and $g_{\mathrm{err}}(X_i, A_i, Y_i, 1)$ instead of $\mathbf{1}\{Y_i \neq 0\}$ and $\mathbf{1}\{Y_i \neq 1\}$.

Example 3 (DP). *Using the matrix $\mathbf{M}$ for demographic parity as described in Section 2, the cost-sensitive reduction for a vector of Lagrange multipliers $\boldsymbol{\lambda}$ uses costs*

$$C_i^0 = \mathbf{1}\{Y_i \neq 0\}, \quad C_i^1 = \mathbf{1}\{Y_i \neq 1\} + \frac{\lambda_{A_i}}{p_{A_i}} - \sum_{a \in \mathcal{A}} \lambda_a,$$

where $p_a := \widehat{\mathbb{P}}[A = a]$ and $\lambda_a := \lambda_{(a,+)} - \lambda_{(a,-)}$, effectively replacing two non-negative Lagrange multipliers by a single multiplier, which can be either positive or negative. Because $c_k = 0$ for all k, $\widehat{c}_k = \varepsilon_k$. Furthermore, because all empirical moments are bounded in $[0, 1]$, we can assume $\varepsilon_k \leq 1$, which yields the bound $\rho \leq 2$. Thus, Algorithm 1 terminates in at most $16B^2 \log(2\,|\mathcal{A}| + 1) \,/\, \nu^2$ iterations.

Example 4 (EO). *For equalized odds, the cost-sensitive reduction for a vector of Lagrange multipliers $\boldsymbol{\lambda}$ uses costs*

$$\begin{aligned}
C_i^0 &= \mathbf{1}\{Y_i \neq 0\}, \\
C_i^1 &= \mathbf{1}\{Y_i \neq 1\} + \frac{\lambda_{(A_i,Y_i)}}{p_{(A_i,Y_i)}} - \sum_{a \in \mathcal{A}} \frac{\lambda_{(a,Y_i)}}{p_{(\star,Y_i)}},
\end{aligned}$$

where $p_{(a,y)} := \widehat{\mathbb{P}}[A = a, Y = y]$, $p_{(\star,y)} := \widehat{\mathbb{P}}[Y = y]$, and $\lambda_{(a,y)} := \lambda_{(a,y,+)} - \lambda_{(a,y,-)}$. If we again assume $\varepsilon_k \leq 1$, then we obtain the bound $\rho \leq 2$. Thus, Algorithm 1 terminates in at most $16B^2 \log(4\,|\mathcal{A}| + 1) \,/\, \nu^2$ iterations.

3.3. Error Analysis

Our ultimate goal, as formalized in equation (3), is to minimize the classification error while satisfying fairness constraints under a true but unknown distribution over (X, A, Y). In the process of deriving Algorithm 1, we introduced three different sources of error. First, we replaced the true classification error and true moments with their empirical versions. Second, we introduced a bound B on the magnitude of $\boldsymbol{\lambda}$. Finally, we only run the optimization algorithm for a fixed number of iterations, until it reaches suboptimality level ν. The first source of error, due to the use of empirical rather than true quantities, is unavoidable and constitutes the underlying statistical error. The other two sources of error, the bound B and the suboptimality level ν, stem from the optimization algorithm and can be driven arbitrarily small at the cost of additional iterations. In this section, we show how the statistical error and the optimization error affect the true accuracy and the fairness of the randomized classifier returned by Algorithm 1—in other words, how well Algorithm 1 solves our original problem (3).

To bound the statistical error, we use the Rademacher complexity of the classifier family $\mathcal{H}$, which we denote by $R_n(\mathcal{H})$, where n is the number of training examples.

We assume that $R_n(\mathcal{H}) \leq Cn^{-\alpha}$ for some $C \geq 0$ and $\alpha \leq 1/2$. We note that $\alpha = 1/2$ in the vast majority of classifier families, including norm-bounded linear functions (see Theorem 1 of Kakade et al., 2009), neural networks (see Theorem 18 of Bartlett & Mendelson, 2002), and classifier families with bounded VC dimension (see Lemma 4 and Theorem 6 of Bartlett & Mendelson, 2002).

Recall that in our empirical optimization problem we assume that $\widehat{c}_k = c_k + \varepsilon_k$, where $\varepsilon_k \geq 0$ are error bounds that account for the discrepancy between $\boldsymbol{\mu}(Q)$ and $\widehat{\boldsymbol{\mu}}(Q)$. In our analysis, we assume that these error bounds have been set in accordance with the Rademacher complexity of $\mathcal{H}$.

Assumption 1. *There exists $C, C' \geq 0$ and $\alpha \leq 1/2$ such that $R_n(\mathcal{H}) \leq Cn^{-\alpha}$ and $\varepsilon_k = C' \sum_{j \in \mathcal{J}} |M_{k,j}| n_j^{-\alpha}$, where n_j is the number of data points that fall in $\mathcal{E}_j$,*

$$n_j \coloneqq \left|\left\{i : (X_i, A_i, Y_i) \in \mathcal{E}_j\right\}\right|.$$

The optimization error can be bounded via a careful analysis of the Lagrangian and the optimality conditions of (P) and (D). Combining the three different sources of error yields the following bound, which we prove in Appendix C.

Theorem 2. *Let Assumption 1 hold for $C' \geq 2C + 2 + \sqrt{\ln(4/\delta)\,/\,2}$, where $\delta > 0$. Let $(\widehat{Q}, \widehat{\boldsymbol{\lambda}})$ be any ν-approximate saddle point of L, let $Q^\star$ minimize $\mathrm{err}(Q)$ subject to $\mathbf{M}\boldsymbol{\mu}(Q) \leq \mathbf{c}$, and let $p_j^\star = \mathbb{P}[\mathcal{E}_j]$. Then, with probability at least $1 - (|\mathcal{J}| + 1)\delta$, the distribution $\widehat{Q}$ satisfies*

$$\mathrm{err}(\widehat{Q}) \leq \mathrm{err}(Q^\star) + 2\nu + \widetilde{O}(n^{-\alpha}),$$
$$\gamma_k(\widehat{Q}) \leq c_k + \frac{1+2\nu}{B} + \sum_{j \in \mathcal{J}} |M_{k,j}|\, \widetilde{O}(n_j^{-\alpha}) \quad \text{for all } k,$$

where $\widetilde{O}(\cdot)$ suppresses polynomial dependence on $\ln(1/\delta)$. If $np_j^\star \geq 8\log(2/\delta)$ for all j, then, for all k,

$$\gamma_k(\widehat{Q}) \leq c_k + \frac{1+2\nu}{B} + \sum_{j \in \mathcal{J}} |M_{k,j}|\, \widetilde{O}\Big((np_j^\star)^{-\alpha}\Big).$$

In other words, the solution returned by Algorithm 1 achieves the lowest feasible classification error on the true distribution up to the optimization error, which grows linearly with ν, and the statistical error, which grows as $n^{-\alpha}$. Therefore, if we want to guarantee that the optimization error does not dominate the statistical error, we should set $\nu \propto n^{-\alpha}$. The fairness constraints on the true distribution are satisfied up to the optimization error $(1 + 2\nu)\ /B$ and up to the statistical error. Because the statistical error depends on the moments, and the error in estimating the moments grows as $n_j^{-\alpha} \geq n^{-\alpha}$, we can set $B \propto n^\alpha$ to guarantee that the optimization error does not dominate the statistical error. Combining this reasoning with the learning rate setting of Theorem 1 yields the following theorem (proved in Appendix C).

Theorem 3. *Let $\rho \coloneqq \max_h \|\mathbf{M}\widehat{\boldsymbol{\mu}}(h) - \widehat{\mathbf{c}}\|_\infty$. Let Assumption 1 hold for $C' \geq 2C + 2 + \sqrt{\ln(4/\delta)\,/\,2}$, where $\delta > 0$. Let $Q^\star$ minimize $\mathrm{err}(Q)$ subject to $\mathbf{M}\boldsymbol{\mu}(Q) \leq \mathbf{c}$. Then Algorithm 1 with $\nu \propto n^{-\alpha}$, $B \propto n^\alpha$ and $\eta \propto \rho^{-2} n^{-2\alpha}$ terminates in $O(\rho^2 n^{4\alpha} \ln |\mathcal{K}|)$ iterations and returns $\widehat{Q}$, where*

$$\mathrm{err}(\widehat{Q}) \leq \mathrm{err}(Q^\star) + \widetilde{O}(n^{-\alpha}),$$
$$\gamma_k(\widehat{Q}) \leq c_k + \sum_{j \in \mathcal{J}} |M_{k,j}|\, \widetilde{O}(n_j^{-\alpha}) \qquad \text{for all } k.$$

Example 5 (DP)**.** *If n_a denotes the number of training examples with $A_i = a$, then Assumption 1 states that we should set $\varepsilon_{(a,+)} = \varepsilon_{(a,-)} = C'(n_a^{-\alpha} + n^{-\alpha})$ and Theorem 3 then shows that for a suitable setting of C', ν, B, and η, Algorithm 1 will return a randomized classifier $\widehat{Q}$ with the lowest feasible classification error up to $\widetilde{O}(n^{-\alpha})$ while also approximately satisfying the fairness constraints*

$$\Big|\mathbb{E}[h(X) \mid A = a] - \mathbb{E}[h(X)]\Big| \leq \widetilde{O}(n_a^{-\alpha}) \quad \text{for all } a,$$

where $\mathbb{E}$ is with respect to (X, A, Y) as well as $h \sim \widehat{Q}$.

Example 6 (EO)**.** *Similarly, if $n_{(a,y)}$ denotes the number of examples with $A_i = a$ and $Y_i = y$ and $n_{(\star,y)}$ denotes the number of examples with $Y_i = y$, then Assumption 1 states that we should set $\varepsilon_{(a,y,+)} = \varepsilon_{(a,y,-)} = C'(n_{(a,y)}^{-\alpha} + n_{(\star,y)}^{-\alpha})$ and Theorem 3 then shows that for a suitable setting of C', ν, B, and η, Algorithm 1 will return a randomized classifier $\widehat{Q}$ with the lowest feasible classification error up to $\widetilde{O}(n^{-\alpha})$ while also approximately satisfying the fairness constraints*

$$\Big|\mathbb{E}[h(X) \mid A = a, Y = y] - \mathbb{E}[h(X) \mid Y = y]\Big| \leq \widetilde{O}(n_{(a,y)}^{-\alpha})$$

for all a, y. Again, $\mathbb{E}$ includes randomness under the true distribution over (X, A, Y) as well as $h \sim \widehat{Q}$.

3.4. Grid Search

In some situations, it is preferable to select a deterministic classifier, even if that means a lower accuracy or a modest violation of the fairness constraints. A set of candidate classifiers can be obtained from the saddle point $(Q^\dagger, \boldsymbol{\lambda}^\dagger)$. Specifically, because $Q^\dagger$ is a minimizer of $L(Q, \boldsymbol{\lambda}^\dagger)$ and L is linear in Q, the distribution $Q^\dagger$ puts non-zero mass only on classifiers that are the Q-player's best responses to $\boldsymbol{\lambda}^\dagger$. If we knew $\boldsymbol{\lambda}^\dagger$, we could retrieve one such best response via the reduction to cost-sensitive learning introduced in Section 3.2.

We can compute $\boldsymbol{\lambda}^\dagger$ using Algorithm 1, but when the number of constraints is very small, as is the case for demographic parity or equalized odds with a binary protected attribute, it is also reasonable to consider a grid of values $\boldsymbol{\lambda}$, calculate the best response for each value, and then select the value with the desired tradeoff between accuracy and fairness.

Example 7 (DP)**.** *When the protected attribute is binary, e.g., $A \in \{a, a'\}$, then the grid search can in fact be conducted in a single dimension. The reduction formally takes*

two real-valued arguments λ_a and $\lambda_{a'}$, and then adjusts the costs for predicting $h(X_i) = 1$ by the amounts

$$\delta_a = \frac{\lambda_a}{p_a} - \lambda_a - \lambda_{a'} \quad \text{and} \quad \delta_{a'} = \frac{\lambda_{a'}}{p_{a'}} - \lambda_a - \lambda_{a'},$$

respectively, on the training examples with $A_i = a$ and $A_i = a'$. These adjustments satisfy $p_a\delta_a + p_{a'}\delta_{a'} = 0$, so instead of searching over λ_a and $\lambda_{a'}$, we can carry out the grid search over δ_a alone and apply the adjustment $\delta_{a'} = -p_a\delta_a/p_{a'}$ to the protected attribute value a'.

With three attribute values, e.g., $A \in \{a, a', a''\}$, we similarly have $p_a\delta_a + p_{a'}\delta_{a'} + p_{a''}\delta_{a''} = 0$, so it suffices to conduct grid search in two dimensions rather than three.

Example 8 (EO)**.** *If $A \in \{a, a'\}$, we obtain the adjustment*

$$\delta_{(a,y)} = \frac{\lambda_{(a,y)}}{p_{(a,y)}} - \frac{\lambda_{(a,y)} + \lambda_{(a',y)}}{p_{(\star,y)}}$$

for an example with protected attribute value a and label y, and similarly for protected attribute value a'. In this case, separately for each y, the adjustments satisfy

$$p_{(a,y)}\delta_{(a,y)} + p_{(a',y)}\delta_{(a',y)} = 0,$$

so it suffices to do the grid search over $\delta_{(a,0)}$ and $\delta_{(a,1)}$ and set the parameters for a' to $\delta_{(a',y)} = -p_{(a,y)}\delta_{(a,y)}/p_{(a',y)}$.

4. Experimental Results

We now examine how our exponentiated-gradient reduction[5] performs at the task of binary classification subject to either demographic parity or equalized odds. We provide an evaluation of our grid-search reduction in Appendix D.

We compared our reduction with the score-based post-processing algorithm of Hardt et al. (2016), which takes as its input any classifier, (i.e., a standard classifier without any fairness constraints) and derives a monotone transformation of the classifier's output to remove any disparity with respect to the training examples. This post-processing algorithm works with both demographic parity and equalized odds, as well as with binary and non-binary protected attributes.

For demographic parity, we also compared our reduction with the *reweighting* and *relabeling* approaches of Kamiran & Calders (2012). Reweighting can be applied to both binary and non-binary protected attributes and operates by changing importance weights on each example with the goal of removing any statistical dependence between the protected attribute and label.[6] Relabeling was developed for binary protected attributes. First, a classifier is trained on the original data (without considering fairness). The training examples close to the decision boundary are then relabeled to remove all disparity while minimally affecting accuracy. The final classifier is then trained on the relabeled data.

As the base classifiers for our reductions, we used the weighted classification implementations of logistic regression and gradient-boosted decision trees in scikit-learn (Pedregosa et al., 2011). In addition to the three baselines described above, we also compared our reductions to the "unconstrained" classifiers trained to optimize accuracy only.

We used four data sets, randomly splitting each one into training examples (75%) and test examples (25%):

- The adult income data set (Lichman, 2013) (48,842 examples). Here the task is to predict whether someone makes more than \$50k per year, with gender as the protected attribute. To examine the performance for non-binary protected attributes, we also conducted another experiment with the same data, using both gender and race (binarized into white and non-white) as the protected attribute. Relabeling, which requires binary protected attributes, was therefore not applicable here.
- ProPublica's COMPAS recidivism data (7,918 examples). The task is to predict recidivism from someone's criminal history, jail and prison time, demographics, and COMPAS risk scores, with race as the protected attribute (restricted to white and black defendants).
- Law School Admissions Council's National Longitudinal Bar Passage Study (Wightman, 1998) (20,649 examples). Here the task is to predict someone's eventual passage of the bar exam, with race (restricted to white and black only) as the protected attribute.
- The Dutch census data set (Dutch Central Bureau for Statistics, 2001) (60,420 examples). Here the task is to predict whether or not someone has a prestigious occupation, with gender as the protected attribute.

While all the evaluated algorithms require access to the protected attribute A at training time, only the post-processing algorithm requires access to A at test time. For a fair comparison, we included A in the feature vector X, so all algorithms had access to it at both the training time and test time.

We used the test examples to measure the classification error for each approach, as well as the violation of the desired fairness constraints, i.e., $\max_a \left|\mathbb{E}[h(X) \mid A = a] - \mathbb{E}[h(X)]\right|$ and $\max_{a,y} \left|\mathbb{E}[h(X) \mid A = a, Y = y] - \mathbb{E}[h(X) \mid Y = y]\right|$ for demographic parity and equalized odds, respectively.

We ran our reduction across a wide range of tradeoffs between the classification error and fairness constraints. We considered $\varepsilon \in \{0.001, \ldots, 0.1\}$ and for each value ran Algorithm 1 with $\widehat{c}_k = \varepsilon$ across all k. As expected, the returned randomized classifiers tracked the training Pareto

[5] https://github.com/Microsoft/fairlearn

[6] Although reweighting was developed for demographic parity, the weights that it induces are achievable by our grid search, albeit the grid search for equalized odds rather than demographic parity.

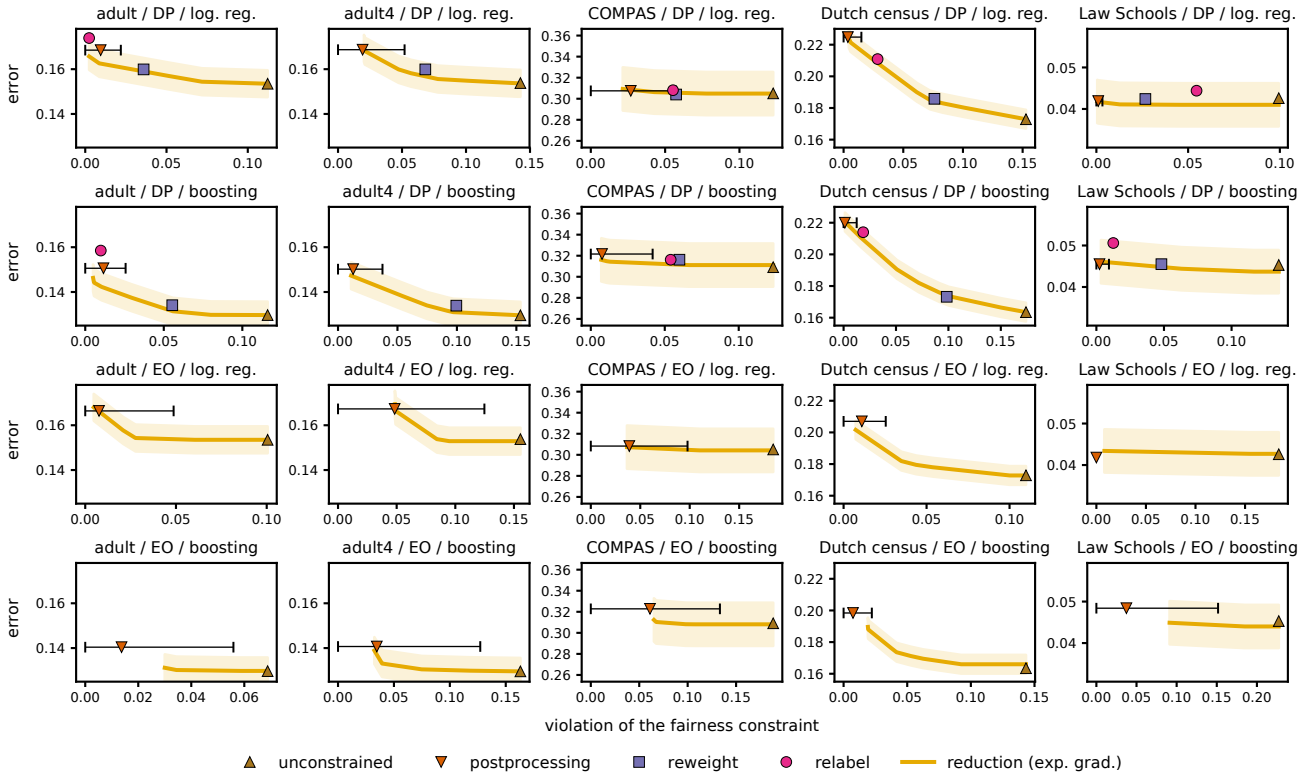

Figure 1. Test classification error versus constraint violation with respect to DP (top two rows) and EO (bottom two rows). All data sets have binary protected attributes except for *adult4*, which has four protected attribute values, so relabeling is not applicable there. For our reduction approach we plot the convex envelope of the classifiers obtained on training data at various accuracy–fairness tradeoffs. We show 95% confidence bands for the classification error of our reduction approach and 95% confidence intervals for the constraint violation of post-processing. Our reduction approach dominates or matches the performance of the other approaches up to statistical uncertainty.

frontier (see Figure 2 in Appendix D). In Figure 1, we evaluate these classifiers alongside the baselines on the *test* data.

For all the data sets, the range of classification errors is much smaller than the range of constraint violations. Almost all the approaches were able to substantially reduce or remove disparity without much impact on classifier accuracy. One exception was the Dutch census data set, where the classification error increased the most in relative terms.

Our reduction generally dominated or matched the baselines. The relabeling approach frequently yielded solutions that were not Pareto optimal. Reweighting yielded solutions on the Pareto frontier, but often with substantial disparity. As expected, post-processing yielded disparities that were statistically indistinguishable from zero, but the resulting classification error was sometimes higher than achieved by our reduction under a statistically indistinguishable disparity. In addition, and unlike the post-processing algorithm, our reduction can achieve any desired accuracy–fairness tradeoff, allows a wider range of fairness definitions, and does not require access to the protected attribute at test time.

Our grid-search reduction, evaluated in Appendix D, sometimes failed to achieve the lowest disparities on the training data, but its performance on the test data very closely matched that of our exponentiated-gradient reduction. However, if the protected attribute is non-binary, then grid search is not feasible. For instance, for the version of the adult income data set where the protected attribute takes on four values, the grid search would need to span three dimensions for demographic parity and six dimensions for equalized odds, both of which are prohibitively costly.

5. Conclusion

We presented two reductions for achieving fairness in a binary classification setting. Our reductions work for any classifier representation, encompass many definitions of fairness, satisfy provable guarantees, and work well in practice.

Our reductions optimize the tradeoff between accuracy and any (single) definition of fairness given training-time access to protected attributes. Achieving fairness when training-time access to protected attributes is unavailable remains an open problem for future research, as does the navigation of tradeoffs between accuracy and multiple fairness definitions.

Acknowledgements

We would like to thank Aaron Roth, Sam Corbett-Davies, and Emma Pierson for helpful discussions.

References

Agarwal, A., Beygelzimer, A., Dudík, M., and Langford, J. A reductions approach to fair classication. In *Fairness, Accountability, and Transparency in Machine Learning (FATML)*, 2017.

Alabi, D., Immorlica, N., and Kalai, A. T. Unleashing linear optimizers for group-fair learning and optimization. In *Proceedings of the 31st Annual Conference on Learning Theory (COLT)*, 2018.

Barocas, S. and Selbst, A. D. Big data's disparate impact. *California Law Review*, 104:671–732, 2016.

Bartlett, P. L. and Mendelson, S. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

Berk, R., Heidari, H., Jabbari, S., Kearns, M., and Roth, A. Fairness in criminal justice risk assessments: The state of the art. arXiv:1703.09207, 2017.

Beygelzimer, A., Dani, V., Hayes, T. P., Langford, J., and Zadrozny, B. Error limiting reductions between classification tasks. In *Proceedings of the Twenty-Second International Conference on Machine Learning (ICML)*, pp. 49–56, 2005.

Boucheron, S., Bousquet, O., and Lugosi, G. Theory of classification: a survey of some recent advances. *ESAIM: Probability and Statistics*, 9:323–375, 2005.

Calmon, F., Wei, D., Vinzamuri, B., Ramamurthy, K. N., and Varshney, K. R. Optimized pre-processing for discrimination prevention. In *Advances in Neural Information Processing Systems 30*, 2017.

Chouldechova, A. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data*, Special Issue on Social and Technical Trade-Offs, 2017.

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

Donini, M., Oneto, L., Ben-David, S., Shawe-Taylor, J., and Pontil, M. Empirical risk minimization under fairness constraints. 2018. arXiv:1802.08626.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pp. 214–226, 2012.

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

Fan, W., Stolfo, S. J., Zhang, J., and Chan, P. K. Adacost: Misclassification cost-sensitive boosting. In *Proceedings of the Sixteenth International Conference on Machine Learning (ICML)*, pp. 97–105, 1999.

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

Freund, Y. and Schapire, R. E. Game theory, on-line prediction and boosting. In *Proceedings of the Ninth Annual Conference on Computational Learning Theory (COLT)*, pp. 325–332, 1996.

Freund, Y. and Schapire, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. In *Neural Information Processing Systems (NIPS)*, 2016.

Johnson, K. D., Foster, D. P., and Stine, R. A. Impartial predictive modeling: Ensuring fairness in arbitrary models. arXiv:1608.00528, 2016.

Kakade, S. M., Sridharan, K., and Tewari, A. On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. In *Advances in neural information processing systems*, pp. 793–800, 2009.

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

Kamishima, T., Akaho, S., and Sakuma, J. Fairness-aware learning through regularization approach. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pp. 643–650, 2011.

Kearns, M., Neel, S., Roth, A., and Wu, Z. S. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2018.

Kivinen, J. and Warmuth, M. K. Exponentiated gradient versus gradient descent for linear predictors. *Information and Computation*, 132(1):1–63, 1997.

Kleinberg, J., Mullainathan, S., and Raghavan, M. Inherent trade-offs in the fair determination of risk scores. In *Proceedings of the 8th Innovations in Theoretical Computer Science Conference*, 2017.

Langford, J. and Beygelzimer, A. Sensitive error correcting output codes. In *Proceedings of the 18th Annual Conference on Learning Theory (COLT)*, pp. 158–172, 2005.

Ledoux, M. and Talagrand, M. *Probability in Banach Spaces: Isoperimetry and Processes*. Springer, 1991.

Lichman, M. UCI machine learning repository, 2013. URL http://archive.ics.uci.edu/ml.

Menon, A. K. and Williamson, R. C. The cost of fairness in binary classification. In *Proceedings of the Conference on Fiarness, Accountability, and Transparency*, 2018.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Rockafellar, R. T. *Convex analysis*. Princeton University Press, 1970.

Shalev-Shwartz, S. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2012.

Wightman, L. LSAC National Longitudinal Bar Passage Study, 1998.

Woodworth, B. E., Gunasekar, S., Ohannessian, M. I., and Srebro, N. Learning non-discriminatory predictors. In *Proceedings of the 30th Conference on Learning Theory (COLT)*, pp. 1920–1953, 2017.

Zafar, M. B., Valera, I., Rodriguez, M. G., and Gummadi, K. P. Fairness constraints: Mechanisms for fair classification. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 962–970, 2017.